Disease modeling with the SIRS model: comparison of a coupled differential equation and Monte Carlo based approach.

Johannes Sørby Heines

December 18, 2020

Abstract

1 Introduction

When determining how best to respond to an outbreak of an infectious disease, being able to predict its spread throughout a given population is crucial. Compartmental models classify the total population into different compartments – for example S: "susceptible", I: "infected" and R: "recovered" – and models the average transitions between these compartments.

Such compartmental models have the advantage of being easily expandable both in terms of transitions and compartments, allowing for models tailored to a specific situation.

In this project we study the SIRS model, in which the immunity gained by recovery is eventually lost, leading to a cyclic system.

After introducing the SIRS model, we explain its implementation both as a system of coupled ordinary differential equations, and as a Monte Carlo system. We outline various improvements on the basic model and compare the two approaches in each case.

2 Theory

All code can be found at `https://github.com/johashei/ComputationalPhysics/tree/master/doc/Projects/2020/Project5/DiseaseModeling/project/code`.

2.1 The SIRS model

The basic SIRS model is a simple cyclic model of the evolution of a contagious disease in a population. It distinguishes three groups: susceptible (S), infected (I), and recovered (R), and models the transitions between these groups. These transitions are given by three quantities, namely the rates of transmission β, recovery γ, and immunity loss λ.

The parameter β is defined as the average number of contacts per person per time, multiplied by the probability of infection in a contact between S and I. The parameters γ and λ are given by the inverse of the average recovery and immunity loss times, respectively.

The transitions are calculated assuming the tree groups mix homogeneously. That is, the proportions of S, I and R in a subset of the population will be the same as in the population as a whole.

The SIRS model can then be described by the following coupled differential equations 1, where each of the terms $\beta SI/N, \gamma I, \lambda R$ is the average number of individuals making the respective transitions $(S \to I), (I \to R), (R \to S)$ in one time unit.

$$\frac{\mathrm{d}S}{\mathrm{d}t} = \lambda R - \beta\frac{SI}{N}, \tag{1a}$$
$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta\frac{SI}{N} - \gamma I, \tag{1b}$$
$$\frac{\mathrm{d}R}{\mathrm{d}t} = \gamma I - \lambda R. \tag{1c}$$

2.2 Improvements to the SIRS model

One of the advantages of the SIRS model is that it can easily be modified to include more effects, by changing or adding new terms to the differential equations 1. In this report, we will treat vital dynamics, seasonal variations and vaccination.

Vital dynamics: The basic SIRS model described above assumes the population remains constant throughout the simulation. If the time scale of the epidemic is comparable to or longer than the average life span, the birth and death rates need to be accounted for. If all newborns are susceptible to the disease, then the equilibrium will change even if the total population remains constant, as dying individuals in I and R are replaced with newborns in S. In addition, the disease may increase the death rate for the infected population.

The corresponding differential equations are

$$\frac{\mathrm{d}S}{\mathrm{d}t} = \lambda R - \beta\frac{SI}{N} - \mu S + \nu N \tag{2a}$$
$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta\frac{SI}{N} - \gamma I - \mu I - \mu_I I \tag{2b}$$
$$\frac{\mathrm{d}R}{\mathrm{d}t} = \gamma I - \lambda R - \mu R \tag{2c}$$

where ν is the birth rate, μ is the death rate not related to the disease, and μ_I is the rate of death caused by the disease.

Seasonal variation: For many common diseases, such as influenza or the common cold, the rate of transmission depends largely on the time of year. This can be caused by a multitude of factors which are not the subject of this report. On the simplest level, this variation can be modeled by adding a harmonic oscillation to the transmission parameter β:

$$\beta(t) = A\cos(\omega t) + \beta_0, \tag{3}$$

where A in the maximum deviation from β_0 and ω is the angular frequency of the seasonal variation.

Vaccination: Vaccination trains the immune system to respond to the infecting agent, creating a transition directly from S to R. As opposed to the previously discussed effects, the rate of vaccination typically depends little on the number of individuals in each compartment, and more on external factors like medical research and vaccination campaigns. We therefore model the vaccination rate as a simple function of time.

$$\frac{\mathrm{d}S}{\mathrm{d}t} = \lambda R - \beta\frac{SI}{N} - f(t) \tag{4a}$$
$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta\frac{SI}{N} - \gamma I \tag{4b}$$
$$\frac{\mathrm{d}R}{\mathrm{d}t} = \gamma I - \lambda R + f(t) \tag{4c}$$

All these improvements, along with many others, can be combined to create a more realistic model for a given scenario.

The basic reproduction number All the improvements outlined above have analytical expressions for the equilibrium, which can be found by setting all three derivatives to zero. Particularly, $\mathrm{d}I/\mathrm{d}t = 0$ gives the equilibrium s^* for S. The number $R_0 = N/s^*$ is then the basic reproduction number, and if it is larger than one, the disease will remain in the population.

3 Methods

The goal of this project was threefold: to make an easily expandable code for the SIRS model, to test the model's behaviour when including the effects described above, and to compare a Monte Carlo implementation to one based on coupled differential equations.

Since every state of the system depends on the previous state, neither algorithm is suited for vectorization nor parallelization. As we chose to focus on expandability rather than speed, this will not be discussed further in this report.

3.1 The coupled ODE approach

The differential equations were solved using the Runge-Kutta 4 algorithm. This method is not symplectic, but has a global error $\mathcal{O}(h^4)$, where h is the chosen step length. The algorithm is

for $i = 0 : n - 1$ **do**
$k_1 = h \cdot f(t_i, y_i)$
$k_2 = h \cdot f(t_{i+1/2}, y_i + \frac{k_1}{2})$
$k_3 = h \cdot f(t_{i+1/2}, y_i + \frac{k_2}{2})$
$k_4 = h \cdot f(t_{i+1}, y_i + k_3)$
$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)$
end for

When simulating a system of coupled differential equations, each line in calculated for every equation before moving to the next.

The Runge-Kutta 4 method for the coupled differential equations was implemented with the python class `ODESolver` written by H.P.Langtangen, and used in the course IN1900. More information on this class can be found in ref [2].

3.2 The Monte Carlo approach

In order to create a Monte Carlo algorithm, transition probabilities must be determined for each transition between two compartments. From the differential equations 1, the number of individuals moving from S to I in a short time interval Δt is given by

$$n_{S\to I} = a\frac{SI}{N}\Delta t.$$

Note that this number is largest when $S = I = N/2$. If we then chose Δt such that

$$\max\left\{a\frac{SI}{N}\Delta t\right\} = a\frac{N}{2}\Delta t = 1,$$

we get a number between 0 and 1, which can be interpreted as a transition probability [1].

By doing this for all transitions, and picking the smallest value for Δt, we get the transition probabilities of the system. At each time step, the Monte Carlo algorithm performs each transition with the corresponding probability.

4 Results

The basic SIRS model was simulated for four populations, each starting at $[S, I, R] = [300, 100, 0]$, and with the parameters shown in table 1. The Results are shown in figure 1.

Table 1

Rate	**A**	**B**	**C**	**D**
β	4	4	4	4
γ	1	2	3	4
λ	0.5	0.5	0.5	0.5

Figure 2 shows the same calculation for population C, but with a different seed for the random number generator.

Figure 3 shows the evolution of population B with an added birth rate and death rate of $\nu = \mu = 0.1$, and no death from the disease.

In figure 4, the size of population C was increased tenfold in order to reduce the effects of random fluctuations. Birth and death rates of $\mu = \nu = 0.9$ were added to the model. In one case, the disease also had a death rate $\mu_I = 0.1$.

Figure 5 shows the effect of an oscillating transmission parameter given by 3 on the population B for two different frequencies.

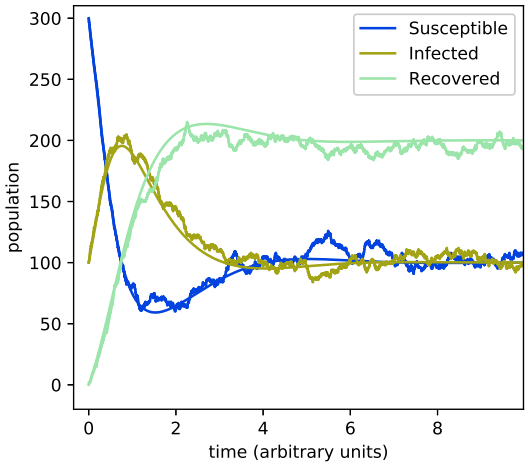
Figure 6a shows the effect of a simple vaccination campaign on population B. The vaccination rate is given by the function

$$f(t) = \begin{cases} 70 & t \in [20, 40], \\ 0 & \text{else.} \end{cases} \tag{5}$$

Figure 6b shows instead an increasing vaccina-

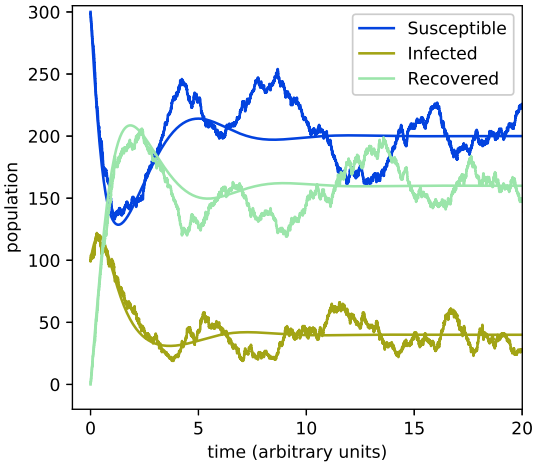
tion rate given by the function

$$f(t) = \begin{cases} t - 20 & t > 20, \\ 0 & \text{else.} \end{cases} \tag{6}$$



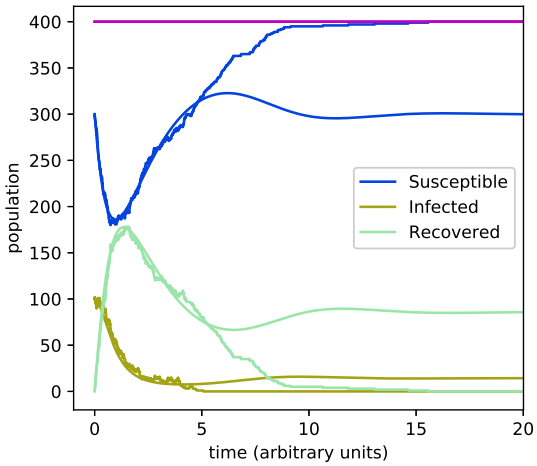
	Monte Carlo	ODE
S	101.1 ± 11.2	100.0
I	100.6 ± 10.1	100.0
R	198.3 ± 9.7	200.0

(a) Population A, equilibrium reached at $t_{\text{eq}} = 10$.



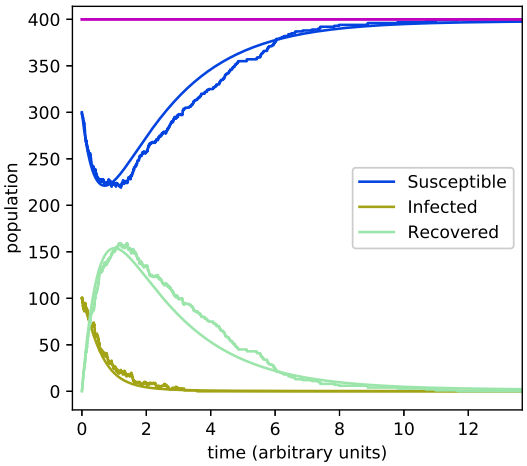
	Monte Carlo	ODE
S	202.8 ± 23.1	200.0
I	39.8 ± 10.7	40.0
R	157.4 ± 18.3	160.0

(b) Population B, equilibrium reached at $t_{\text{eq}} = 20$.



	Monte Carlo	ODE
S	400.0 ± 0.0	300.0
I	0.0 ± 0.0	14.3
R	0.0 ± 0.0	85.7

(c) Population C, equilibrium reached at $t_{\text{eq}} = 20$.



	Monte Carlo	ODE
S	400.0 ± 0.0	399.3
I	0.0 ± 0.0	0.1
R	0.0 ± 0.0	0.6

(d) Population C, equilibrium reached at $t_{\text{eq}} = 20$.

Figure 1: Evolution of the model for the four populations in table 1. The graphs show the evolution before equilibration, and the tables show the equilibrium state. The values for the Monte Carlo are means over $t \in [t_{\text{eq}}, 100]$.

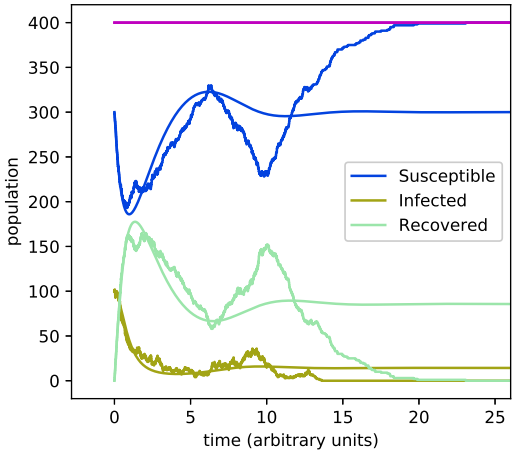
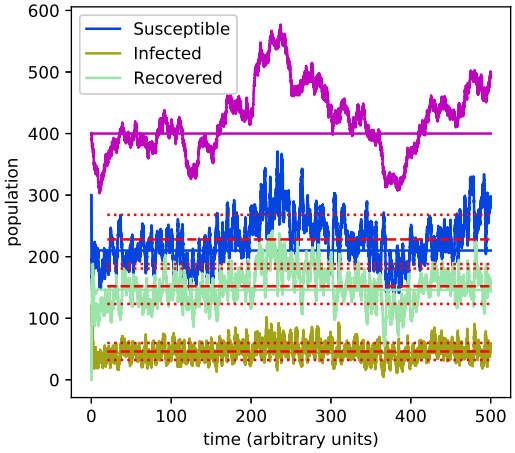
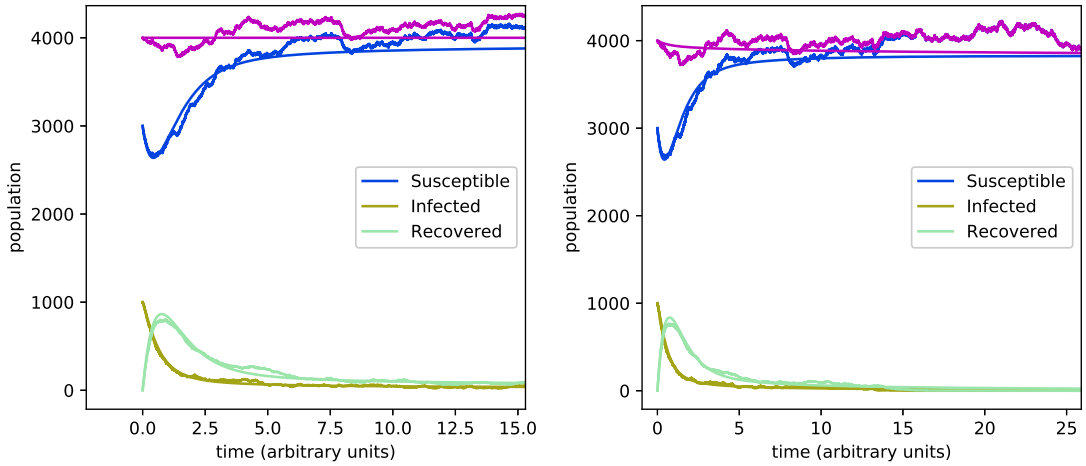


Figure 2: Evolution of the model for population C before equilibrium. The model is identical to the one shown in figure 1c, but uses a different seed when generating random numbers.



	Monte Carlo	ODE
S	228.0 ± 40.0	210.0
I	46.0 ± 13.8	43.8
R	151.9 ± 28.7	146.2

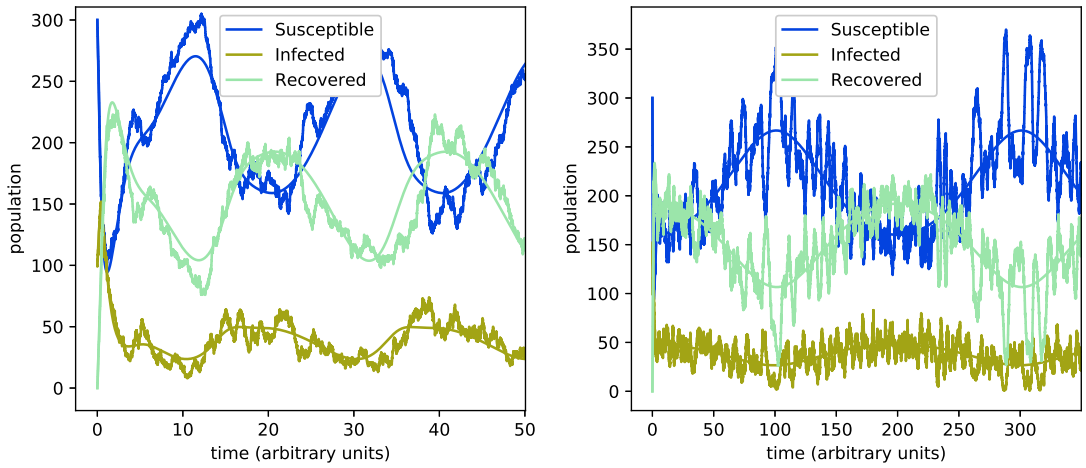
Figure 3: Evolution of population B with added parameters $\nu = \mu = 0.1$. The magenta curves show the total population, and the red lines indicate the mean and standard deviation of the Monte Carlo simulation after $t = 20$.



(a) Vital parameters $\mu = \nu = 0.9$, thus $R_0 = 4/3.9 \approx 1.026$

(b) Vital parameters $\mu = \nu = 0.9$ and $\mu_I = 0$, thus $R_0 = 1$

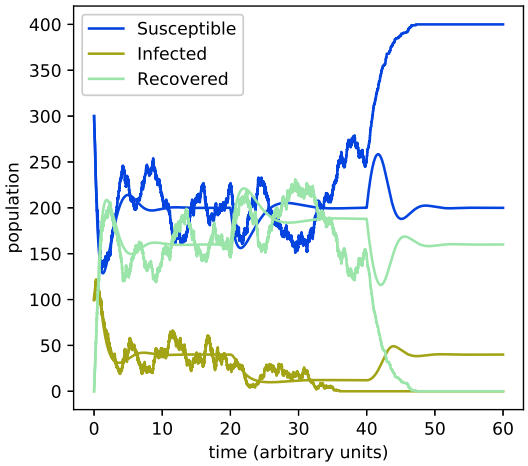
Figure 4: Evolution of the system for a population equivalent to population C but ten times as large, with basic reproduction numbers slightly larger and equal to 1.



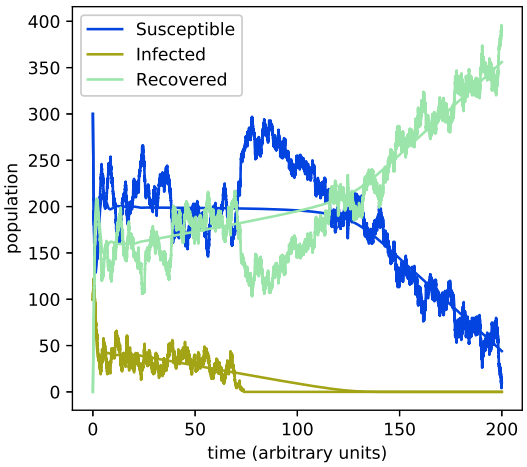
(a) $\omega = \pi/10$: period comparable to t_{eq}.

(b) $\omega = \pi/100$: period much longer than t_{eq}.

Figure 5: Evolution of the model for population B with time dependent transmission parameter $\beta(t)$ given by equation 3, using $A = 1$ and different oscillation frequencies.



(a) Constant vaccination rate of 70 in the time interval $[20, 40]$.



(b) Linearly increasing vaccination rate from $t = 20$.

Figure 6: Evolution of the model for population B when adding vaccination for two different vaccination rates $f(t)$.

5 Discussion

5.1 Impact of the studied improvements

In accordance with expectation, we observe a change in the equilibrium values in figure 3 when compared to figure 1b. The proportions of S, I and R are similar for the two algorithms:

	ODE	Monte Carlo
S	52.50%	53.53%
I	10.95%	10.80%
R	36.55%	35.66%

The large fluctuations in the total population will be discussed in section 5.2.

For the model with vital dynamics, the basic reproduction number is found from equation **??** to be

$$R_0 = \frac{\beta}{\gamma + \mu + \mu_I}.$$

The impact of this threshold value is seen in figure 4. Indeed, in this scenario the lethal disease was eradicated while the non-lethal one remained in the population.

Adding seasonal variation creates a varying equilibrium with the same period. As seem in figure 5, the maximum transmission rate $\beta(t)$ corresponds to the maximum I and R and the minimum S. For periods much longer than the system's equilibration time, the variations are close to sinusoidal; the change is slow enough that the compartments can equilibrate almost instantaneously. This situation is shown in figure 5b. For periods comparable to the equilibration time, as in figure 5a, the other rates also contribute to the shape of the oscillations. For instance, we see a slight delay between the maxima of I and R, due to the recovery rate γ.

In a simple vaccination campaign like the one modeled in figure 6a, where we assume a constant vaccination rate, induces a new equilibration of the system. Moving individuals from S to R also decreases the probability of infection $(aSI/N)\Delta t$, resulting in the equilibrium for S stying the same, as observed. If the number of infected drops low enough, fluctuations in the Monte Carlo simulation can cause it to become zero, eradicating the disease. The ODE algorithm will always return to the original steady state once the campaign ends, because I can

approach zero asymptotically.

For the increasing vaccination rate modeled in figure 6b, the entire population will eventually be vaccinated. While there are still infected, S stays at equilibrium for the same reason as above, and the changes in I and R are symmetrical. Once the disease has been eradicated, the size of S starts to decline linearly and symmetrically with R.

5.2 Notable cases of deviation between the two approches

As seen in figures 2 6 and 1c, the two implementations differ greatly once the number of infected individuals approaches zero. If the I is close to zero for a long time compared to Δt, then the Monte Carlo algorithm is likely to reach zero due to random fluctuations, ending the pandemic. This also means the seed of the random number generator will have a large impact on the resulting behaviour, as illustrated by the difference in figures 2 and 1c.

The vital dynamics are extremely sensitive to the random fluctuations of the Monte Carlo algorithm. While the basic SIRS model has one steady state for a given set of parameters, there is no single steady state for the total population. If $\nu = \mu$, then the population is expected to remain constant, but has no preference for a particular value. In keeping with this, figure 3 shows roughly constant proportions of S, I and R, with large fluctuations in the total population.

When the change is smaller than one individual per time step, the Monte Carlo algorithm changes faster than the ODEs. This is illustrated in fig **??**. The reason for this behaviour is that the Monte Carlo will make its minimum change of one individual before the ODE reach that change, effectively jumping ahead.

For such small numbers of individuals, treating the system statistically is no longer justifiable, and the model breaks down. One can however use the different possibilities as starting points for new simulations.

5.3 Shortcomings of the model

This SIRS model assumes the recovery follows an exponential curve as seen in figure **??**, with expectation value $1/D_r$ where D_r is the average recovery time. Recovery from a disease is not a time independent stochastic process; it is the result of many complex processes within the body, as the immune system reacts to the disease. In other words, the probability of recovery depends on the time spent in the infected category.

In many of the cases described above, the fluctuations in the Monte Carlo algorithm had a large impact on the evolution of the disease once the number of individuals in a compartment became small. This illustrates a fundamental problem with this type of model: small groups cannot be described accurately by statistics. The dependence on fluctuations reflects how such a situation can vary heavily based on individual behaviour. Thus, the SIRS model and similar compartmental models can only predict the evolution of a system while each compartment is large.

6 Conclusion

References

[1] Hjort-Jensen, Morten, (2020) *Project 5, Disease Modeling*, URL: `http://compphysics.github.io/ComputationalPhysics/doc/Projects/2020/Project5/DiseaseModeling/html/DiseaseModeling.html`.

[2] Langtangen, Hans Petter & Sundnes, Joakim, (2017) *App.E: Systems of differential equations*, URL: `https://www.uio.no/studier/emner/matnat/ifi/IN1900/h17/ressurser/slides/ode_systems_short.pdf`

[3] Amaro, J.E., Dudouet, J., Orce, J.N., (2020) Global analysis of the COVID-19 pandemic using simple epidemiological models, *Applied Mathematical Modelling*, 90, p.995 - 1008.