University of Sheffield

Modelling and Simulation of Natural Systems Assignment

Using an agent-based model to investigate the relationship between population density and the rate of spread of a virus similar to the flu

Sean Heneghan

Demetra C Neacsu

Matthew Butt

Peter Bull

Department of Computer Science

May 2020

Abstract

In this paper we will examine the relationship between population density and the rate of spread of a virus similar to the flu. We have created a model which simulates how a virus, with characteristics similar to that of the flu, spreads across an environment from human to human. There will be 2 types of agents: a healthy human and an infected human. A stochastic method has been used in order to determine how the virus will pass along agents, whereby proximity of agents is the key factor which determines infection. The results show that the virus is able to spread faster, reaching its peak number of infections more quickly, the more densely populated the environment is in which agents interact.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Background

Influenza is transmitted either through human-human contact or human-environment. While the virus is mainly spread via aerosol infection such as coughing, sneezing etc., the virus can survive on surfaces humans come into contact with and touch. It can even survive in water if the temperature is relatively low Blut et al. (2009). The virus is generally quite vulnerable on surfaces and is dependent on factors such as temperature and humidity to spread effectively Lowen et al. (2007) which is why certain seasons tend to bring on more cases of Influenza (particularly in winter). The virus' mortality rate changes depending on access to healthcare and immunisation but humans with respiratory, autoimmune or cardiovascular problems tend to be more at risk Zarocostas (2009).

1.1 Literature Review

Agent-based modelling has been used in a variety of ways to represent the spread of Influenza and similar contagious viruses among the population. Many models seek to use this representation as a way of testing methods to combat the spread among the population. In Kumar et al. (2013), an agent-based model is constructed in order to show the effectiveness of paid sick days in the spread of Influenza. Allowing workers to have one or two paid sick days with Influenza showed to drastically decrease the transmission among the population due to reduced contact. Mao (2011) saw something very similar too, as the control strategy was to extend the weekend by a few more days, seeing a decrease in Influenza outbreaks for extensions more than two days. This is just one example of a control strategy that may cause a slowdown in the rate of spread among the population. In Khaled M. Khalil & Salem (2011), many control strategies were run on the same model of 1000 agents in Egypt, looking at how the spread of Influenza changed when agents were either socially distanced, increasingly visited by doctors and the infected, quarantined or vaccinated. It was found that simply awareness of Influenza in the population offered the greatest impact on the virus spreading, followed by vaccination of 50 percent of the population. Social distancing and quarantining of the agents during the peak of spread proved inefficient with little to no impact. This is interesting as what we saw in Kumar et al. (2013) and Mao (2011) was that agents with sick days or extended weekends, which is more or less minor quarantining/isolation, caused the outbreak severity to lessen. This may be down to a factor of compliance among agents with the rules implemented though, as in a more realistic model some agents will ignore guidelines such as to take sick days or isolate.

Models may be made for more than one type of agent, so as to test the effectiveness of strategies on a non-homogeneous situation. Marek Laskowski (2011) shows that patient-oriented infection control policies tend to have a larger effect than those targeting healthcare workers. Non-homogeneous models come with their drawbacks though, as they very much rely on computational power and efficiency. In Aleman et al. (2011) there were 256 CPU's used to execute their model in a computationally reasonable time.

As our model will be run on single CPU architecture on MATLAB, even with the most efficient coding, a non-homogeneous model would not be viable. Aleman et al. (2011) did however show a much less severe outbreak of Influenza when infected individuals stayed at home. Assumptions must therefore be made in regards to the model in order to save computational efficiency. Such assumptions are like those made in Guo et al. (2015) where the frequency of contact between agents differs depending on the environment they are currently placed in.

A human who contracts Influenza typically does not show symptoms straight away. There is an incubation period between when the person first contracts Influenza and when they show symptoms of having it. The symptom period occurs after the incubation period and is when you are likely to see signs that someone is affected by Influenza, however the affected sometimes do not show symptoms; this is called being asymptomatic. As such, humans that are asymptomatic continue as if nothing is wrong therefore keeping the Influenza spread ongoing. The percentage of afflicted humans that show no symptoms does vary depending on the environment, however Nancy H. L. Leung & Cowling (2015) found that it ranges from 4% to 28%.

For an infectious disease such as Influenza, which can rapidly spread across a population, population density also needs to be taken into account as a key factor which can influence the spread. Humans that congregate around large cities tend to be the ones who pass on the disease the fastest, as there is a larger rate of contact among individuals. In an agent-based model, population density can be varied across the given environment to symbolise towns, cities etc, where people live in close proximity. As Hunter et al. (2018) shows, varying the population density in the model allowed them to mimic the population spread of the town Schull in Ireland and show how an outbreak would affect that town. They later go on to mention that by varying just a few parameters in their model, they would be able to simulate an outbreak in any town in Ireland. In this study they investigated the outbreak of Measles on the population, however note that they could replicate the study to show an outbreak of any disease that spreads in a similar SEIR (Susceptible, Exposed, Infectious, Recovered) manner, such as Influenza.

Chapter 2

Methodology

2.1 Overview

The approach that we have decided on is based on the simulation models that describe agent-based models written in the paper by Volker Grimm (2006).

This project aims to simulate and observe the spread of a virus in an environment with different levels of population density. Analysis will be conducted on the time it takes for a virus to reach peak infections, as well as how long it takes once the peak has been reached for the population to be entirely healthy again, the former being the focus however as re-infections will not take place (as will be explained in the following section).

2.2 Design Concept

The system presents 2 agents: a healthy and an infected agent.

According to Nancy H. L. Leung & Cowling (2015) infected agents present a 16% probability of being asymptomatic. In the model presented, if the subject is not asymptomatic, they will move randomly until the end of their incubation period. The non-asymptomatic infected agents will then stop moving for the duration of the symptom period. On the other hand, if the patients are asymptomatic, they will move randomly for the equivalent of the incubation and symptom period as they have no catalyst for their own isolation. The probability for a healthy agent to become infected is modelled as p = 0.2/d, where d represents the distance in kilometers to an infected agent and 0.2 is the base probability of infection. Thus the probability of infection is based on the proximity of agents, however there is also a random element (the requirement that p>rand) to help simulate the randomness of whether an encounter in the real world would result in an infection. Though it should be noted that if the two agents are in extremely close proximity then it will still be a guaranteed infected, shown mathematically by the fact that if d<0.2, p>rand no matter what (where rand is equal to a random number between 0 and 1).

Once the infected agents have reached the end of the symptom period, they recover. In our model, the infected subject will despawn and respawn a healthy agent, which has immunity, in its place. We are operating on the assumption that you are guaranteed to gain immunity after recovering from the virus.

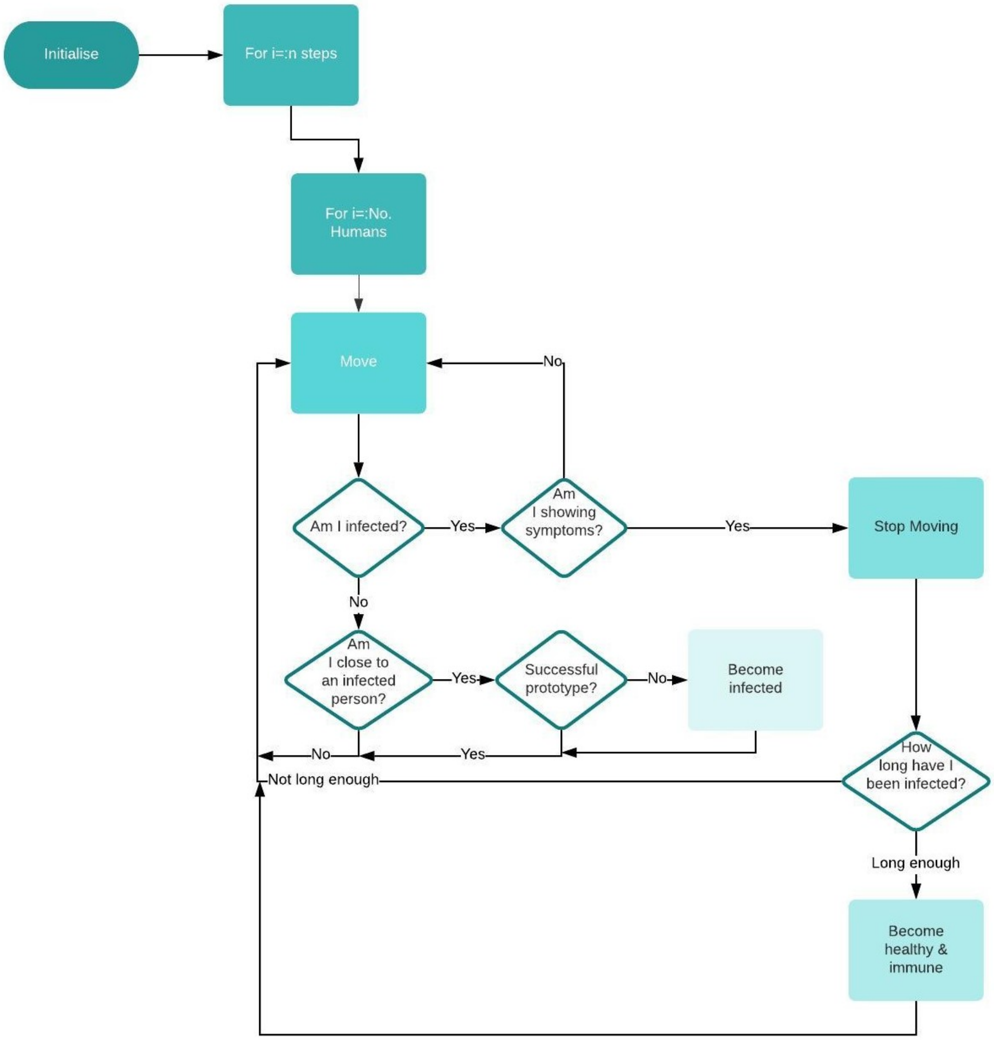


Figure 2.1: *Flowchart Illustrating Model Logic*

2.3 Details

Parameter values have been chosen with reference to the data available to us regarding Influenza. Despite this, it should be made clear that our aim is to generate a model which allows us to answer our research question, as opposed to attempting to perfectly replicate the behaviour of Influenza. The area in which our simulation is run has been chosen as an arbitrary value of 50km^2 and it stays constant as the population value increases in order to simulate different population densities.

The value for the probability of infection has been chosen after researching several online sources, including research conducted by the World-Health-Organization (2010) on limiting the spread of pandemic and seasonal epidemic influenza. With this in mind, and in consideration of the fact that our model is not simulating any virus strain in particular, we felt that choosing a value of 20% for the probability of infection was suitable for the purposes of our experiments.

In order to choose the values for duration of symptoms and incubation period we conducted further research, taking into consideration several high-credibility sources such as the World-Health-Organization (2010) and for Disease Control & Prevention (2019). For the duration of symptoms we are using a value of 5 days, which is represented in our code as 120 iterations, where 1 iteration is 1 hour. The incubation period has been set as 2 days, or 48 iterations in the code. Finally, the speed of agents has been set to a conservative constant speed of 2km/h.

Parameters	**Value**
Area (km^2)	50
Base Infection Rate	20%
Symptom Period	5 days (120 iterations)
Incubation Period	2 days (48 iterations)
Speed of agents (km/h)	2

Table 2.1: *Parameters Table*

Chapter 3

Results

3.1 Averaged Results for Each Simulation Criterion

No. of Agents	Avg. Peak No. Infected	Avg. Peak Infection Iteration	Iterations Until All Healthy	Avg. Infection Spread Rate	Avg. Recovery Rate
200	200	129.20	300.20	1.55	1.17
400	400	99.60	268.60	4.02	2.37
600	600	77.60	246.60	7.73	3.55
800	800	71.80	240.80	11.14	4.73
1000	1000	59.20	228.00	16.89	5.92

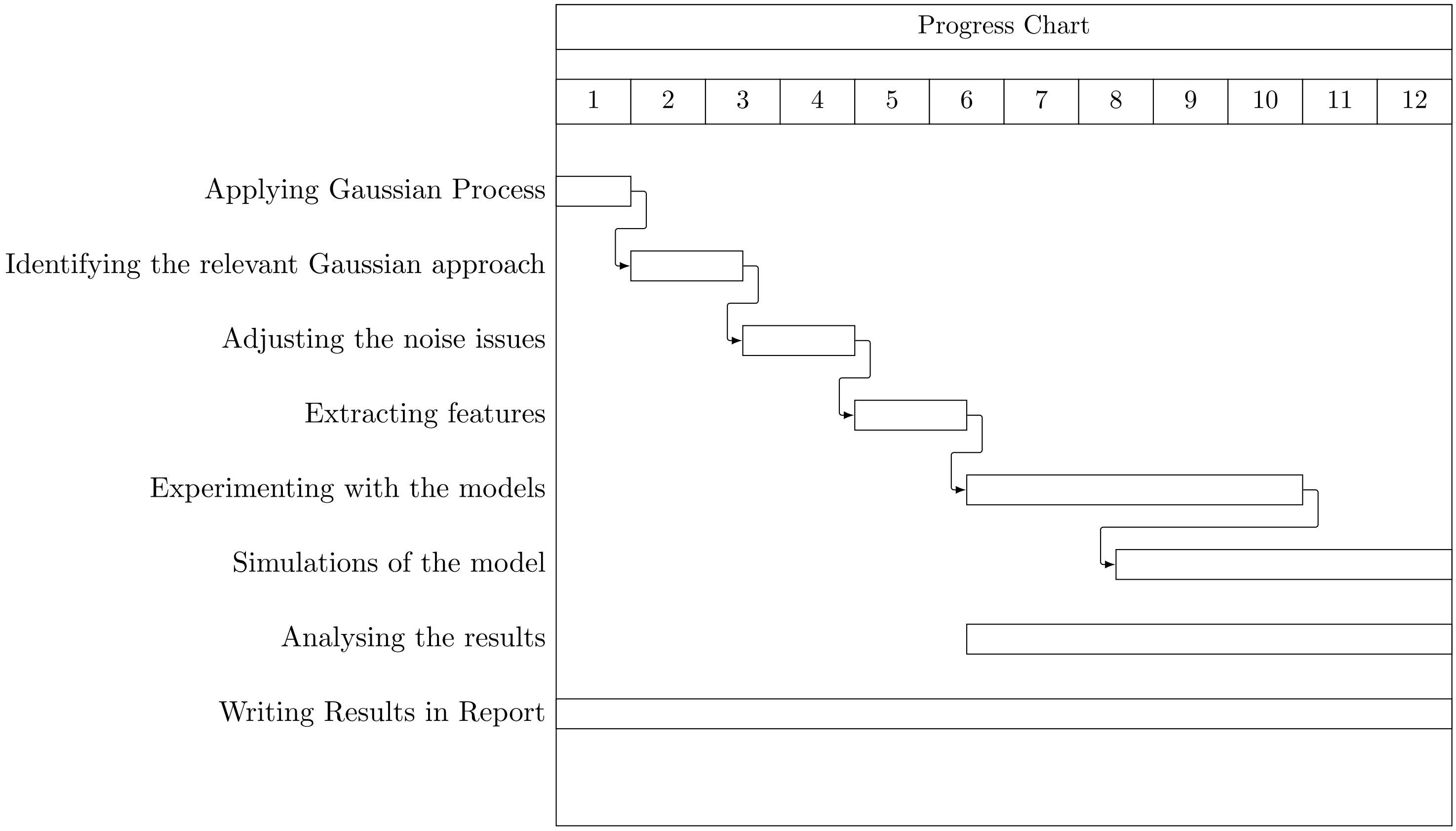
Table 3.1: *Averaged Results for Each Simulation Criterion*

3.2 Model Predictions

In order to test the model, it was decided that a fixed (arbitrary) area of 50km^2 would be used for the agents to interact within, as mentioned previously. Next it was decided that 1000 agents should not be exceeded in any given test in order to reduce unnecessary computational strain. With this in mind, simulations containing 200, 400, 600, 800 and 1000 agents were pursued. It was ensured that the simulations had enough iterations of runtime to transition to peak infection and back down to 100% healthy agents via some preliminary testing to scope upper bounds for the required number of iterations. The composition of each simulation followed the rule that there is 1 infected agent to begin with and the rest of the agents are healthy. Each test was repeated 5 times in recognition of the stochastic behaviour which fuels the migration system of the model.

As shown in Table 3.1, for each population density the peak number of infected agents was in fact the total number of agents, meaning that in regards to comparison of results between different population densities, the infection spread rate and recovery rate can be focused on. The average infection spread rate (infections per iteration) increased from 1.55 (200 agents) to 4.02 (400 agents) to 7.73 (600 agents) to 11.14 (800 agents) to 16.89 (1000 agents). The average recovery rate (recoveries per iteration) increased from 1.17 (200 agents) to 2.37 (400 agents) to 3.55 (600 agents) to 4.73 (800 agents) to 5.92 (1000 agents). This indicates a fairly linear increase in both the average infection rate and average recovery rate as the population density is increased. Located in the appendix are tables containing the individual results from which these averages are formed, as well as graphical representations of each individual simulation illustrating the associated infection and recovery curves.

3.3 Computational Efficiency



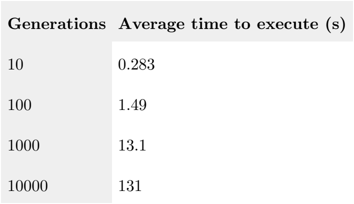
3.3.1 Method

The results were gathered using the MATLAB function 'tic' and were obtained on a computer only running MATLAB to try and avoid any external functions that may affect the outcome. All plotting functions and command window outputs were disabled during these tests and results are averages of five complete program runs displayed to three significant figures. Unless stated otherwise, the input to the program is ecolab(20,100,10,100).

It would be expected for program runtime to increase as each of the variables is increased (world size, healthy agents, infected agents and generations), but in an ideal program none of these increases should result in more than a linear growth, as the program should not have any parameters that would affect the runtime disproportionately.

3.3.2 Generations

It is expected that increasing the number of generations would result in a linear increase in execution time. The input for this set of tests was ecolab(20,100,10,x), where x represents the number of generations being tested. Performing the tests on our model gave the following results:



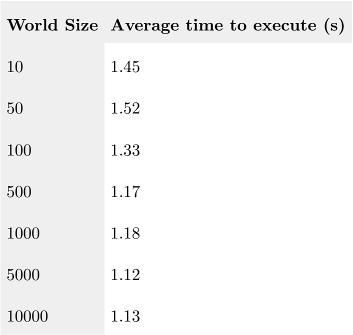
Generations	Average time to execute (s)
10	0.283
100	1.49
1000	13.1
10000	131

Table 3.2: *Computational Efficiency Table - Generations*

These results support the theory that the increase would be linear.

3.3.3 World Size

Other than initialisation the world size should not impact the runtime at all, so it is our prediction that the results will be consistent with each other, other than a slight increase as world size is increased to allow for a slightly longer initialisation phase. The input for this set of tests was ecolab(x,100,10,100), where x represents the world size being tested. The results we gathered are as follows:



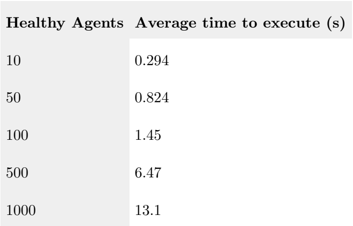
World Size	Average time to execute (s)
10	1.45
50	1.52
100	1.33
500	1.17
1000	1.18
5000	1.12
10000	1.13

Table 3.3: *Computational Efficiency Table - World size*

The data we collected was actually the opposite of what we were expecting, but having examined the code we are confident we have found the explanation. Whilst the larger world sizes would cause a slight increase in initialisation, this is more than outweighed by the fact that as the world size increases past a certain point the chances of two agents interacting is increasingly small, thereby causing a faster runtime as the program is no longer required to create and remove agents (a relatively computationally expensive action). This results in having slightly longer runtimes for the smaller world sizes but once a certain threshold is reached then the results remain relatively consistent.

3.3.4 Healthy Agents

We expect that increasing the number of healthy agents will have a significant impact on the runtime, as each healthy agent calculates the distance to all infected agents. Therefore increasing healthy agents will not only result in more calculations as a consequence of their own movement, but also due to the calculation of distances between increasingly more and more agents. The input for this set of tests was ecolab(20,x,10,100), where x represents the number of healthy agents being tested.



Healthy Agents	Average time to execute (s)
10	0.294
50	0.824
100	1.45
500	6.47
1000	13.1

Table 3.4: *Computational Efficiency Table - Healthy agents*

The results we gathered provide evidence that the relationship between healthy agents and runtime is linear. This makes sense as the number of healthy agents is never used within the code for a healthy agent and while the impact on runtime is significant, it is not so high as to cause issues when dealing with smaller numbers of agents.

3.3.5 Infected Agents

We expect infected agents to have an almost identical impact as healthy agents did. The input for this set of tests was ecolab(20,100,x,100), where x represents the number of infected agents being tested.

Infected Agents	Average time to execute
10	1.57
50	2.11
100	2.58
500	7.23
1000	13.1

Table 3.5: *Computational Efficiency Table - Infected Agents*

The results we collected supported this idea, and the discrepancy in the sets of values can be attributed to the differing number of total agents in the two simulations. For example, for the 10 agents test for healthy agents there are 20 agents total (10 + the 10 infected), whereas in the 10 agents test for infected agents there are 110 agents (10 + the 100 healthy). It is worth noting that the impact each type of agent has on runtime are very close, with healthy agents only causing marginally more slowdown (due to having to calculate the distance to all infected agents as described above).

Chapter 4

Discussion

4.1 Overview

The simulations we have run have been successful in allowing us to achieve the initial aim of the project: simulating and observing the spread of a virus across a population, in different densities. Our tests show that the density of a population has a direct effect on the rate of the spread of a virus, as a higher density allows for a faster spread and thus a sharper peak in infections.

Our results appear to show a linear scaling of the average infection spread rate as a function of population density. However it should be noted that if we were able to run simulations much greater than 1000 agents, it is possible that this scaling could reveal different characteristics due to an increase in sample size.

We couldn't find another agent-based model which investigates our research question in particular and could provide like-for-like results, however we can compare our findings to another study, Li et al. (2018) which showed that higher density populations resulted in a higher rate of deaths when looking at pandemics across America. Not only were there more deaths in denser populated states but they reached the peak of their deaths much earlier than less densely populated states. The study investigated influenza with pneumonia in addition, something which would raise the mortality rate, across a few states in America. The rate of deaths from a virus is typically at least linearly proportional to rate of spread and thus if we were able to scale up our results to that of a population the size of Philadelphia, we feel they would reinforce one another well in the sense that a higher rate of spread is shown in the denser populated areas.

4.2 Computational Efficiency - Conclusions and Possible Future Improvements

From our testing we can see that none of the parameters scale more than linearly which is close to ideal for a model of this nature. The two that caused the greatest impact on runtime (the two types of agents) is an issue but given the type of system and the environment we are modelling, this type of increase in runtime is to be expected.

Of course, in building the system we aimed to maximise computational efficiency as best we could but there are potentially some other methods we could try in an effort to maximise efficiency, such as reworking how the infection spreads. This could possibly be achieved by removing the *infected* function from the healthy agents and instead using an *infect* function in the infected agents, thereby reducing the number of agents (at least initially) that need to perform these calculations.

Another potential improvement that could be made is limiting the number of chances an agent has to complete a successful move. Currently if an agent can't complete a move it tries again until it can. Potentially by limiting this we could save on computation at the cost of reducing the model's accuracy, although if the threshold was high enough this may not have a meaningful impact on the simulation as a whole.

4.3 Graphs of the System

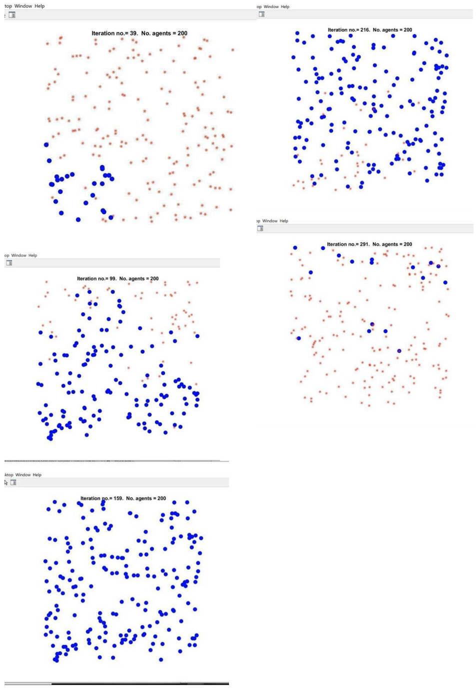

Iteration no.= 39. No. agents = 200
Iteration no.= 216. No. agents = 200
Iteration no.= 99. No. agents = 200
Iteration no.= 291. No. agents = 200
Iteration no.= 159. No. agents = 200

Figure 4.1: *Infection spread until point of peak infections followed by recovery of population*

4.4 Findings and Future Solutions

As described in the overview, our simulations are showing that the rate of spread of a virus is directly influenced by population density. The time it takes for a population to reach peak infection rate diminishes as the size of the population rises, assuming a constant area.

Furthermore, as expected, the average recovery rate also rises with a rise in population density.

When creating our agent-based model we decided to only use two agents, a healthy agent and an infected one, whilst opting to not include a carrier agent which was initially in our plans. Rather than using a carrier agent, we included its functionality in the infected agent in an effort to increase the computational efficiency of our model. Furthermore and as mentioned previously, we have made the assumption that once an infected agent reaches the end of its symptom period, it will gain immunity to the virus. In the code, this is represented by the infected agent de-spawning, and a healthy agent with immunity spawning in its place. This assumption has a direct impact on the results we have obtained. Creating the model with only a fraction of infected agents gaining immunity, or none gaining immunity at all, would provide different results particularly in the form of a slowdown in recovery rate due to re-infections.

With regards to future work, the model could use different values for the base probability of infection, which is currently modelled as 20%. Currently the value selected represents an average value from researching several different sources, such as the World-Health-Organization (2010) on the spread of different seasonal epidemics or pandemics. However, in the future, the current value could be replaced by a more specific value that perhaps represents a specific virus. This could be useful in an investigation where the aim is centered around gaining a more thorough understanding of how a particular virus spreads and perhaps any differences in emergent behaviour that different viruses may elicit.

Chapter 5

Conclusion

Our investigation was to look at the relationship between population density and the rate of spread of a flu-like virus. From our results we can discern that there is a positive correlation between the density of the population and the rate of spread of the infectious disease. From this relationship, we should see that the peak of the disease should occur much earlier in the model's cycle the denser the population, which is supported by our results.

Through the implementation of some simple rules, our model has displayed the ability to simulate the spread of a flu-like virus to a suitable degree for the type of investigation that we're undertaking, as is evident from the way the virus can be seen to spread in the simulations. This implies that the agent-based model approach is advantageous in representing certain aspects, in their entirety, in the spread of such a virus. Given that this model is only plausible up to a certain population density threshold due to computational constraints, our model will be unable to scale up to large cities so as to represent an outbreak in London or Birmingham. Our model is also quite homogeneous in how the agents are defined. The extent to which our model can be applied to real-life scenarios with any degree of accuracy is limited by default due to the inherent difficulties of modelling the intricacies of human interaction. However the stochastic approach which we have taken still grants us insight into the emergent behaviour of the population as a whole, namely that of the infection spread. With more time and computational power, we might have been able to show the effects of 'overcrowding' in our model. This is a more realistic effect of high population densities during a virus outbreak whereby the standard of living conditions decline due to overpopulation in a given area, such as what Korean hospitals faced in November 2009, Yoon et al. (2011). Finally, as was alluded to previously, given that our code was intended to be a fairly general model for the spread of a virus amongst a population, it could be used to model other viruses transmitted from human-human simply through parameter modifications.

Bibliography

Aleman, D. M., Wibisono, T. G. & Schwartz, B. (2011), ‘A nonhomogeneous agent-based simulation approach to modeling the spread of disease in a pandemic outbreak’, *Interfaces* **41**(3), 301–315.

Blut, A. et al. (2009), ‘Influenza virus’, *Transfusion Medicine and Hemotherapy* **36**(1), 32.

for Disease Control, C. & Prevention (2019), ‘Clinical signs and symptoms of influenza’, *Signs and symptoms* .
URL: *https://www.cdc.gov/flu/professionals/acip/clinical.htm?fbclid=IwAR2igeK6bKkMkS7QkzTrX4KD6J*
QAFv5MGx0ZmIQSSBo

Guo, D., Li, K. C., Peters, T. R., Snively, B. M., Poehling, K. A. & Zhou, X. (2015), ‘Multiscale modeling for the transmission of influenza and the evaluation of interventions toward it’.
URL: *https://www.nature.com/articles/srep08980*

Hunter, E., Mac Namee, B. & Kelleher, J. (2018), ‘An open-data-driven agent-based model to simulate infectious disease outbreaks’, *PloS one* **13**(12).

Khaled M. Khalil, M. Abdel-Aziz, T. T. N. & Salem, A.-B. M. (2011), ‘An agent-based modeling for pandemic influenza in egypt’, (2).
URL: *https://arxiv.org/ftp/arxiv/papers/1001/1001.5275.pdf*

Kumar, S., Grefenstette, J. J., Galloway, D., Albert, S. M. & Burke, D. S. (2013), ‘Policies to reduce influenza in the workplace: impact assessments using an agent-based model’, *American journal of public health* **103**(8), 1406–1411.

Li, R., Richmond, P. & Roehner, B. M. (2018), ‘Effect of population density on epidemics’, *Physica A: Statistical Mechanics and its Applications* **510**, 713–724.

Lowen, A. C., Mubareka, S., Steel, J. & Palese, P. (2007), ‘Influenza virus transmission is dependent on relative humidity and temperature’, *PLoS Pathog* **3**(10), e151.

Mao, L. (2011), ‘Agent-based simulation for weekend-extension strategies to mitigate influenza outbreaks’, *BMC public health* **11**(1), 522.

Marek Laskowski, Bryan C P Demianyk, J. W. (2011), ‘Agent-based modeling of the spread of influenza-like illness in an emergency department: A simulation study’, (2), 4477–4479.
URL: $https://www.researchgate.net/publication/51544257_Agent -$
$Based_Modeling_of_the_Spread_of_Influenza-Like_Illness_in_an_Emergency_Department_{AS}imulation_Study$

Nancy H. L. Leung, Cuiling Xu, D. K. M. I. & Cowling, B. J. (2015), 'The fraction of influenza virus infections that are asymptomatic: a systematic review and meta-analysis', *Epidemiology* **338**, 862–872.

Volker Grimm, Uta Berger, F. B. S. E. V. G. J. G. J. G.-C. T. G. S. K. H. G. H. A. H. J. U. J. C. J. W. M. M. B. M. G. P. C. P. S. F. R. A. M. R. M. M. R. E. R. N. R. E. S. S. S. R. A. S. R. V. U. V. D. L. D. (2006), 'A standard protocol for describing individual-based and agent-based models', *Ecological Modelling* **198**, 115–126.

World-Health-Organization (2010), 'Public health research agenda for influenza', *Public Health Research Agenda for Influenza* .
URL: *https://www.who.int/influenza/resources/research/*$research_{a}genda_{i}nfluenza_{s}tream_{2l}imiting_{s}pread.p$
$IwAR3OJ3gs_{R}mqEx22Rkcr0SaOTffH5lUcZh6ASu2XW42NAz9d - QBsm6vDTNE$

Yoon, Y. H., Choi, S. H., Cho, H. J., Kim, J. Y., Han, C., Lee, K. H. & Shin, S. D. (2011), 'H1n1 influenza: hospital countermeasures and the need for planning', *Journal of the Korean Society of Emergency Medicine* **22**(1), 72–78.

Zarocostas, J. (2009), 'World health organization declares a (h1n1) influenza pandemic', *BMJ: British Medical Journal (Online)* **338**.

Appendices

Appendix A

Appendix

A.1 Results Tables

Run Statement	No. Agents	Test	Peak No. Infected	Peak Infection Iteration	Iterations Until All healthy	Averages
ecolab(50,199,1,350)	200	1	200	151	320	Avg. peak number infected is 200
ecolab(50,199,1,350)	200	2	200	123	292	Avg. peak infection iteration is 129.20
ecolab(50,199,1,350)	200	3	200	135	304	Avg. iterations until all healthy is 300.20
ecolab(50,199,1,350)	200	4	200	126	305	Avg. infection spread rate is 1.55
ecolab(50,199,1,350)	200	5	200	111	280	Avg. recovery rate is 1.17

Table A.1: *Results table for 200-agent simulations*

Run Statement	**No. Agents**	**Test**	**Peak No. Infected**	**Peak Infection Iteration**	**Iterations Until All healthy**	**Averages**
ecolab(50,399,1,300)	400	1	400	91	260	Avg. peak number infected is 400
ecolab(50,399,1,300)	400	2	400	100	269	Avg. peak infection iteration is 99.60
ecolab(50,399,1,300)	400	3	400	121	290	Avg. iterations until all healthy is 268.60
ecolab(50,399,1,300)	400	4	400	86	255	Avg. infection spread rate is 4.02
ecolab(50,399,1,300)	400	5	400	100	269	Avg. recovery rate is 2.37

Table A.2: *Results table for 400-agent simulations*

Run Statement	**No. Agents**	**Test**	**Peak No. Infected**	**Peak Infection Iteration**	**Iterations Until All healthy**	**Averages**
ecolab(50,599,1,270)	600	1	600	81	250	Avg. peak number infected is 600
ecolab(50,599,1,270)	600	2	600	71	240	Avg. peak infection iteration is 77.60
ecolab(50,599,1,270)	600	3	600	78	247	Avg. iterations until all healthy is 246.60
ecolab(50,599,1,270)	600	4	600	75	244	Avg. infection spread rate is 7.73
ecolab(50,599,1,270)	600	5	600	83	252	Avg. recovery rate is 3.55

Table A.3: *Results table for 600-agent simulations*

Run Statement	No. Agents	Test	Peak No. Infected	Peak Infection Iteration	Iterations Until All healthy	Averages
ecolab(50,799,1,250)	800	1	800	77	246	Avg. peak number infected is 800
ecolab(50,799,1,250)	800	2	800	66	235	Avg. peak infection iteration is 71.80
ecolab(50,799,1,250)	800	3	800	79	248	Avg. iterations until all healthy is 240.80
ecolab(50,799,1,250)	800	4	800	71	240	Avg. infection spread rate is 11.14
ecolab(50,799,1,250)	800	5	800	66	235	Avg. recovery rate is 4.73

Table A.4: *Results table for 800-agent simulations*

Run Statement	No. Agents	Test	Peak No. Infected	Peak Infection Iteration	Iterations Until All healthy	Averages
ecolab(50,999,1,250)	1000	1	1000	52	224	Avg. peak number infected is 1000
ecolab(50,999,1,250)	1000	2	1000	67	230	Avg. peak infection iteration is 59.20
ecolab(50,999,1,250)	1000	3	1000	60	229	Avg. iterations until all healthy is 228.00
ecolab(50,999,1,250)	1000	4	1000	62	230	Avg. infection spread rate is 16.89
ecolab(50,999,1,250)	1000	5	1000	55	224	Avg. recovery rate is 5.92

Table A.5: *Results table for 1000-agent simulations*

Appendix B

Appendix

B.1 Results Graphs

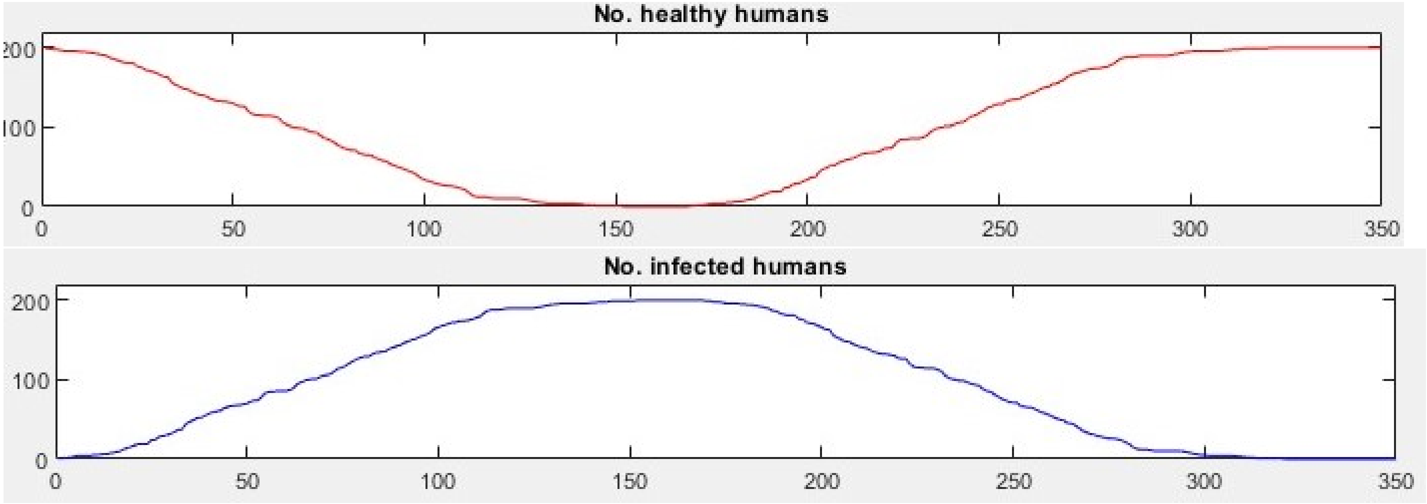


Figure B.1: *Graphs showing test 1 results for 200-agent simulation*

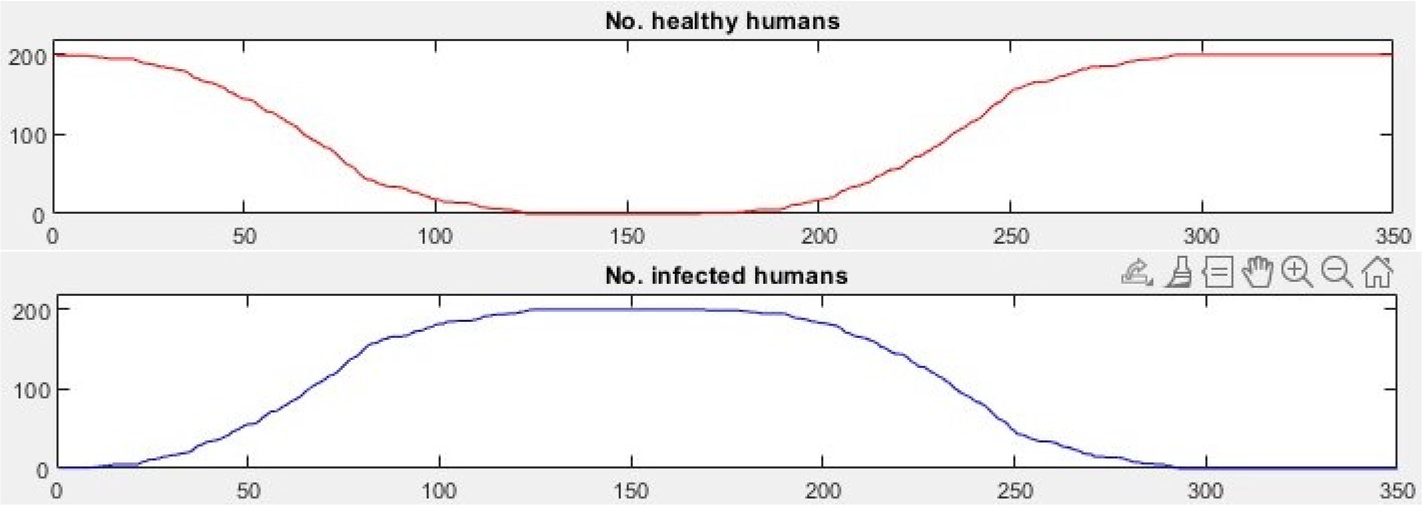


Figure B.2: *Graphs showing test 2 results for 200-agent simulation*

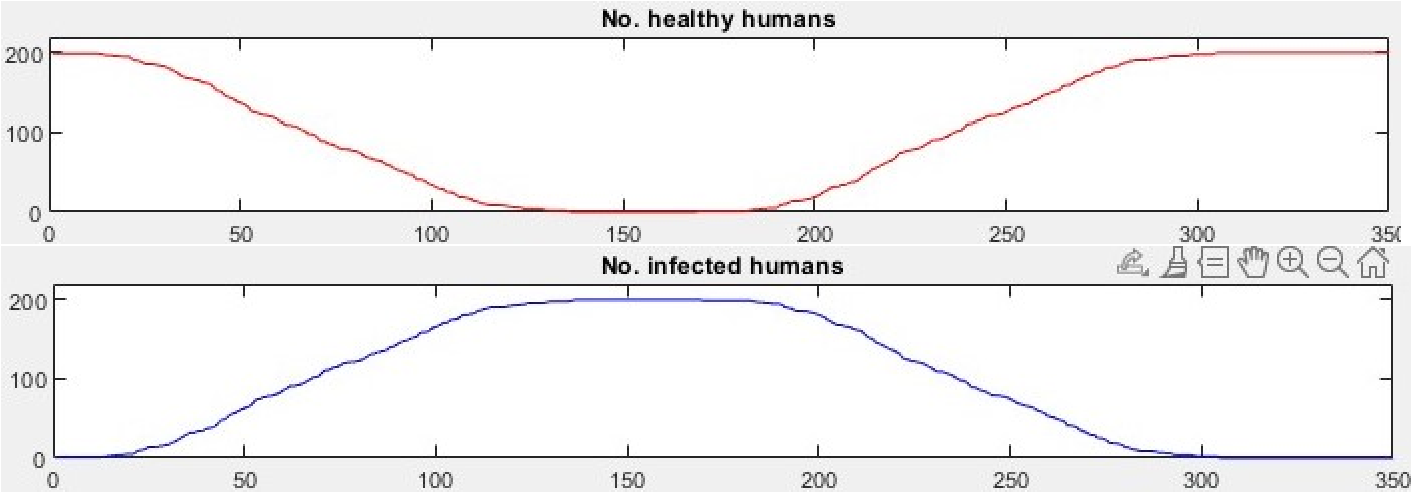


Figure B.3: *Graphs showing test 3 results for 200-agent simulation*

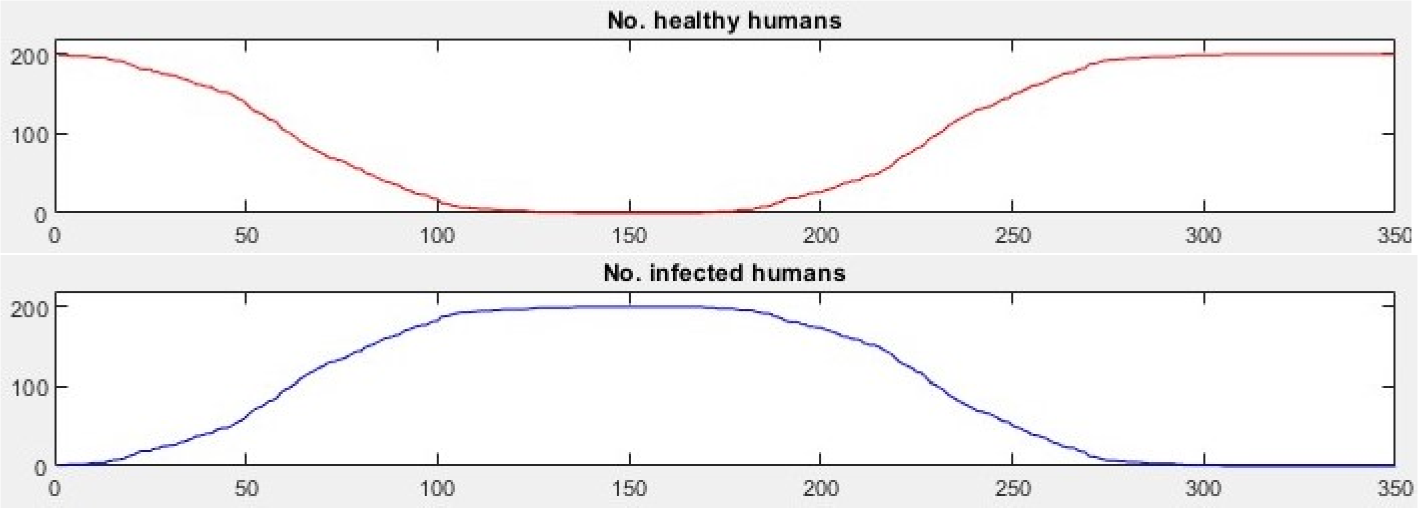


Figure B.4: *Graphs showing test 4 results for 200-agent simulation*

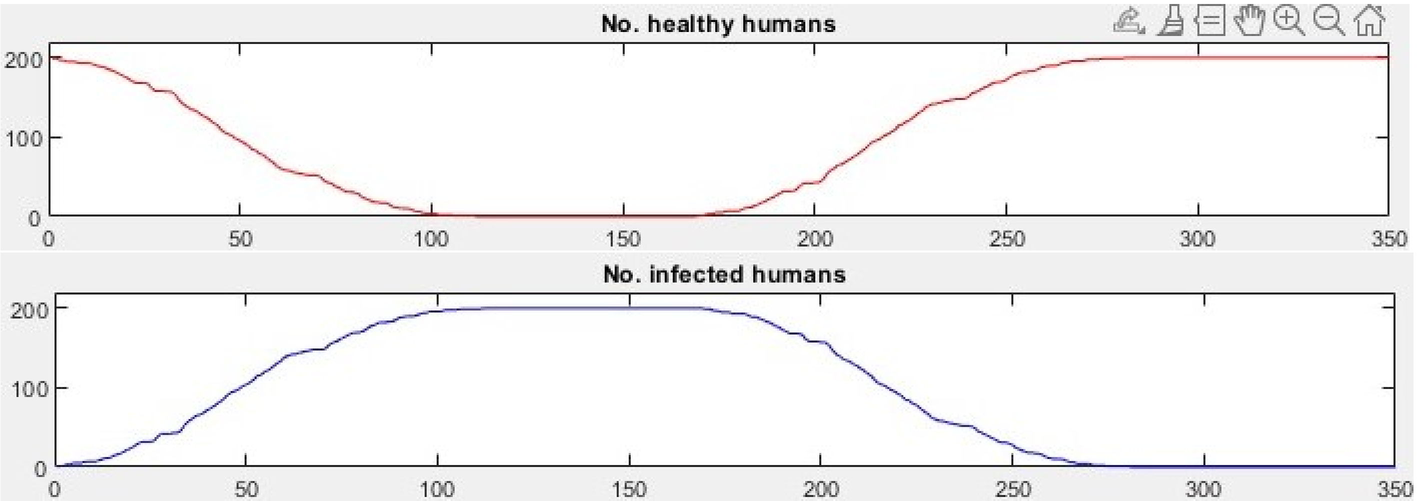


Figure B.5: *Graphs showing test 5 results for 200-agent simulation*

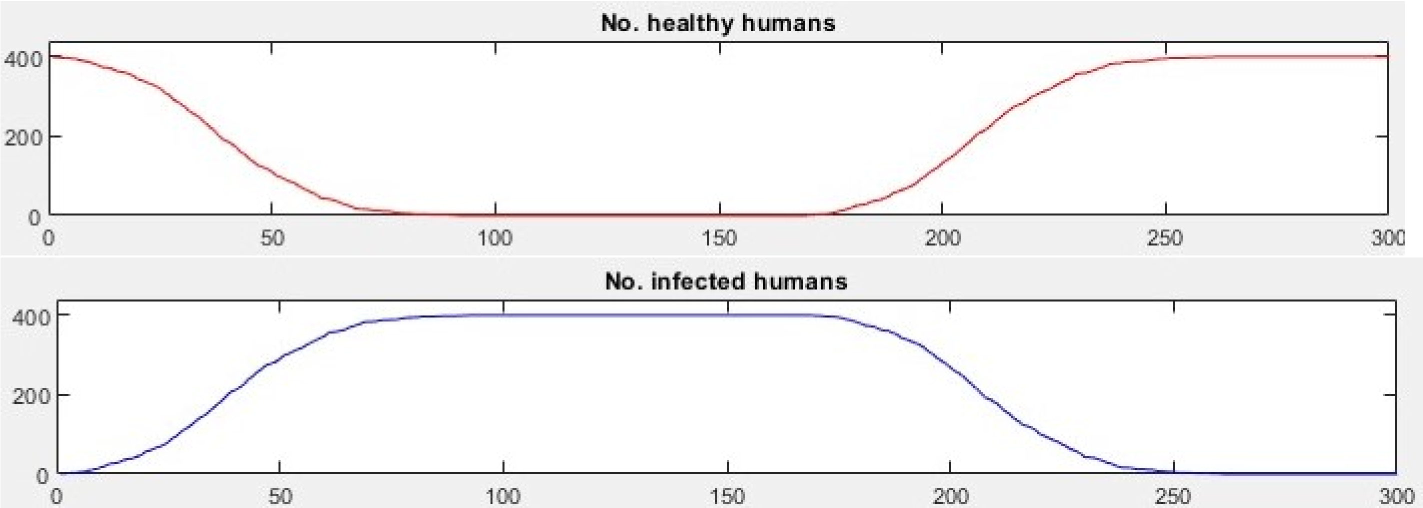


Figure B.6: *Graphs showing test 1 results for 400-agent simulation*

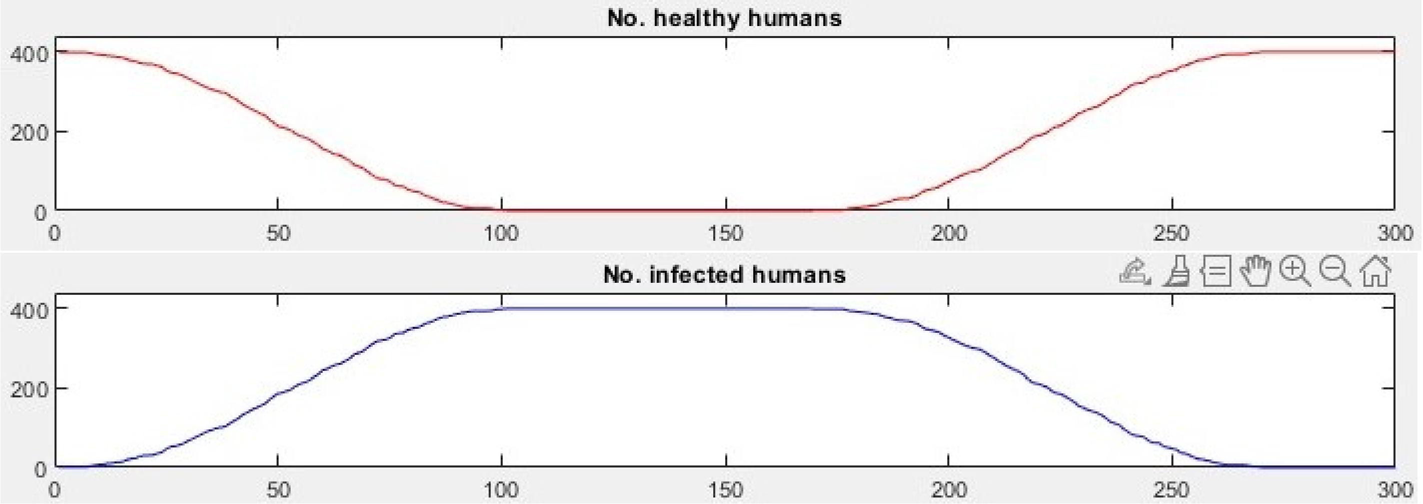


Figure B.7: *Graphs showing test 2 results for 400-agent simulation*

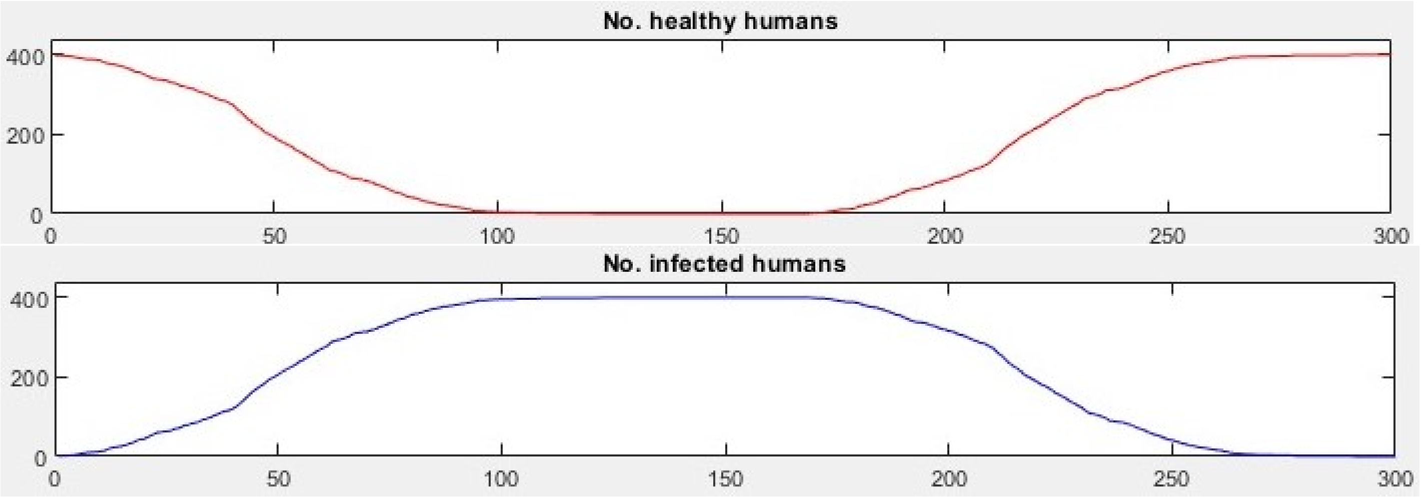


Figure B.8: *Graphs showing test 3 results for 400-agent simulation*

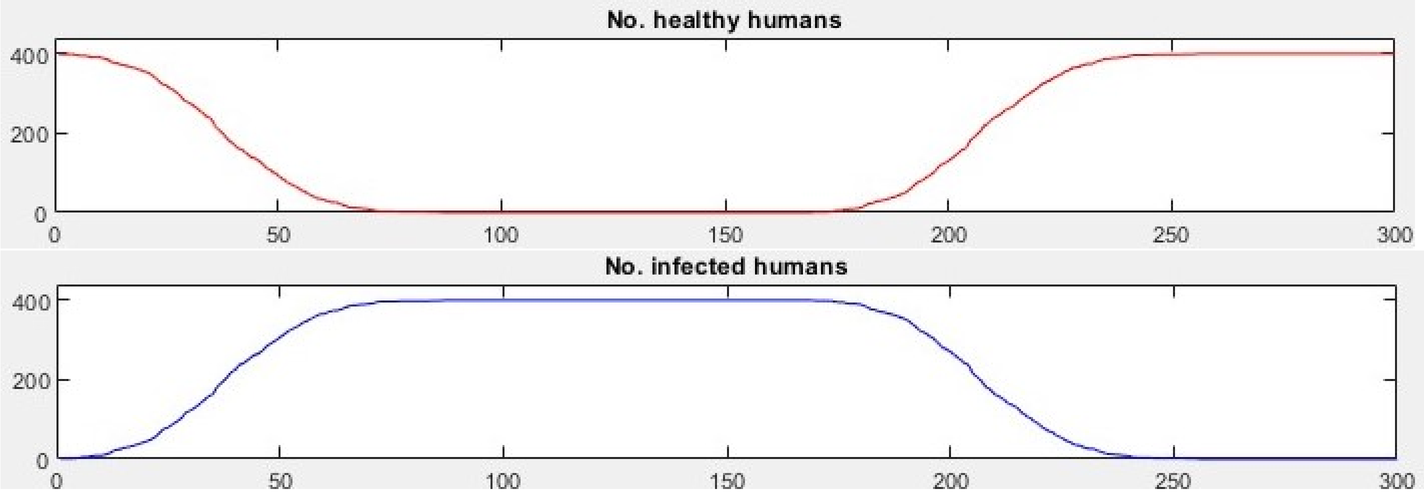


Figure B.9: *Graphs showing test 4 results for 400-agent simulation*

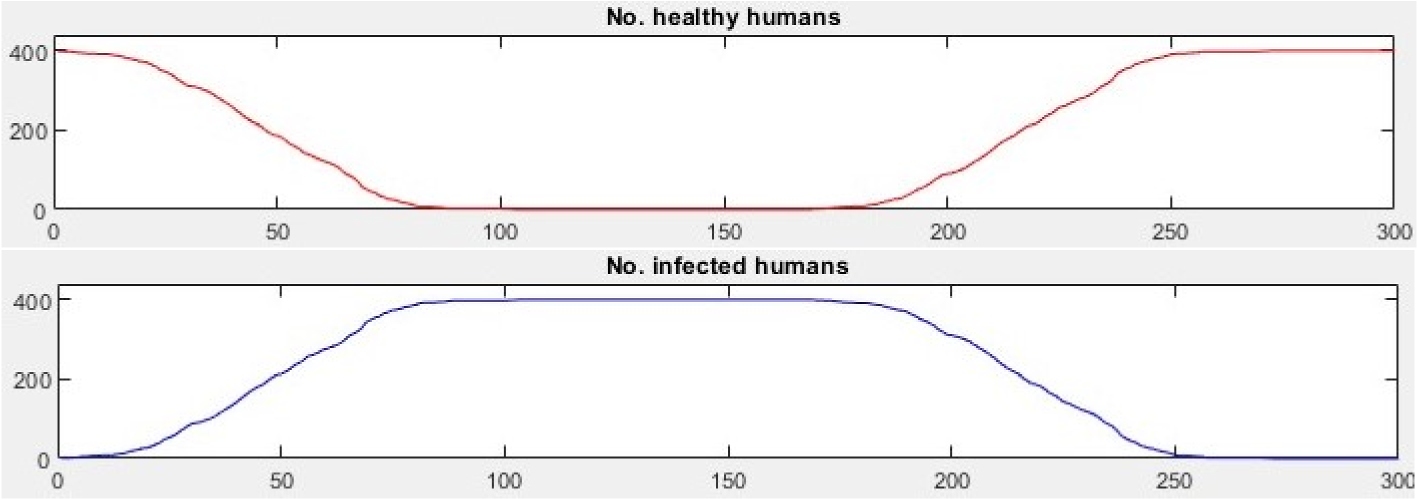


Figure B.10: *Graphs showing test 5 results for 400-agent simulation*

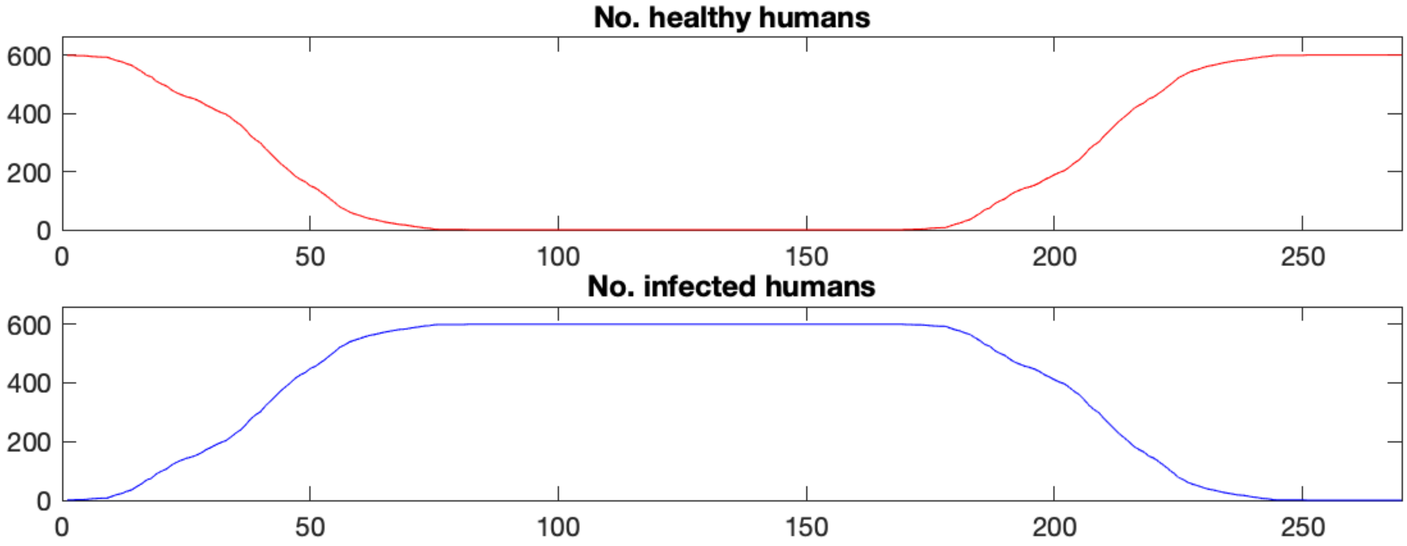


Figure B.11: *Graphs showing test 1 results for 600-agent simulation*

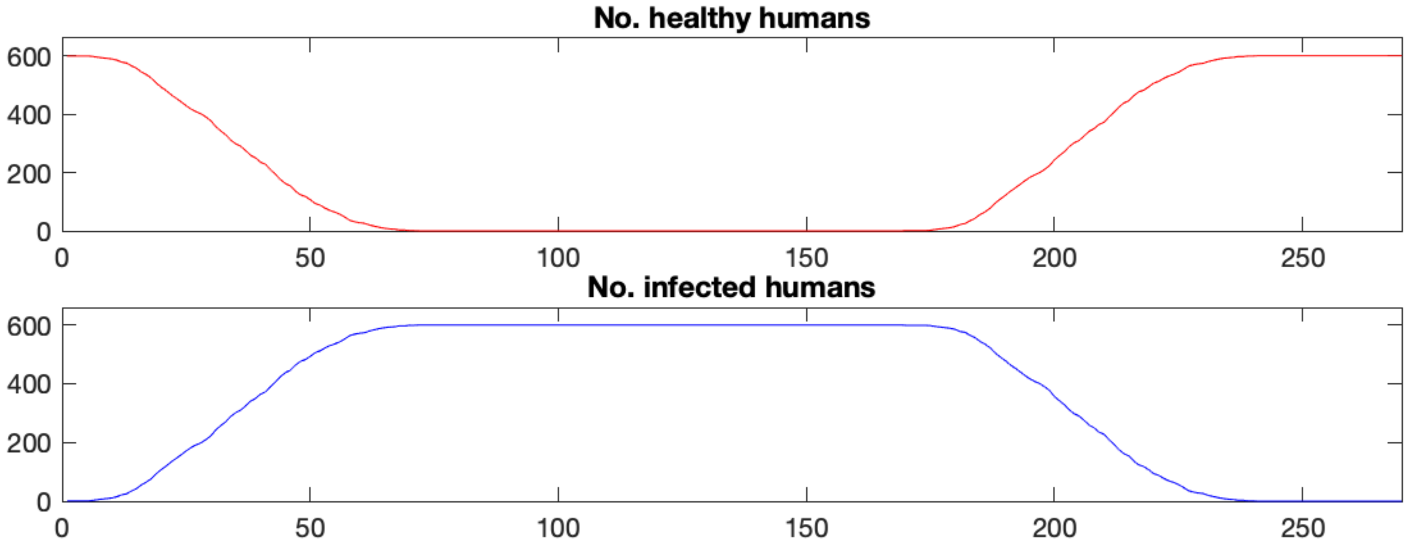


Figure B.12: *Graphs showing test 2 results for 600-agent simulation*

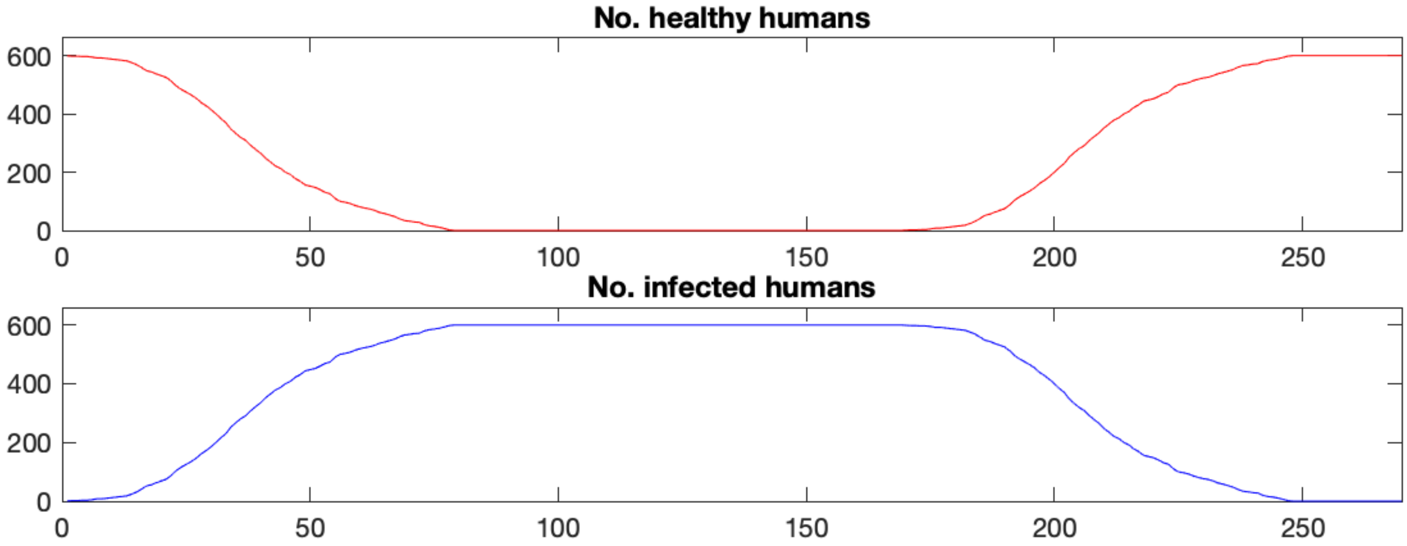


Figure B.13: *Graphs showing test 3 results for 600-agent simulation*

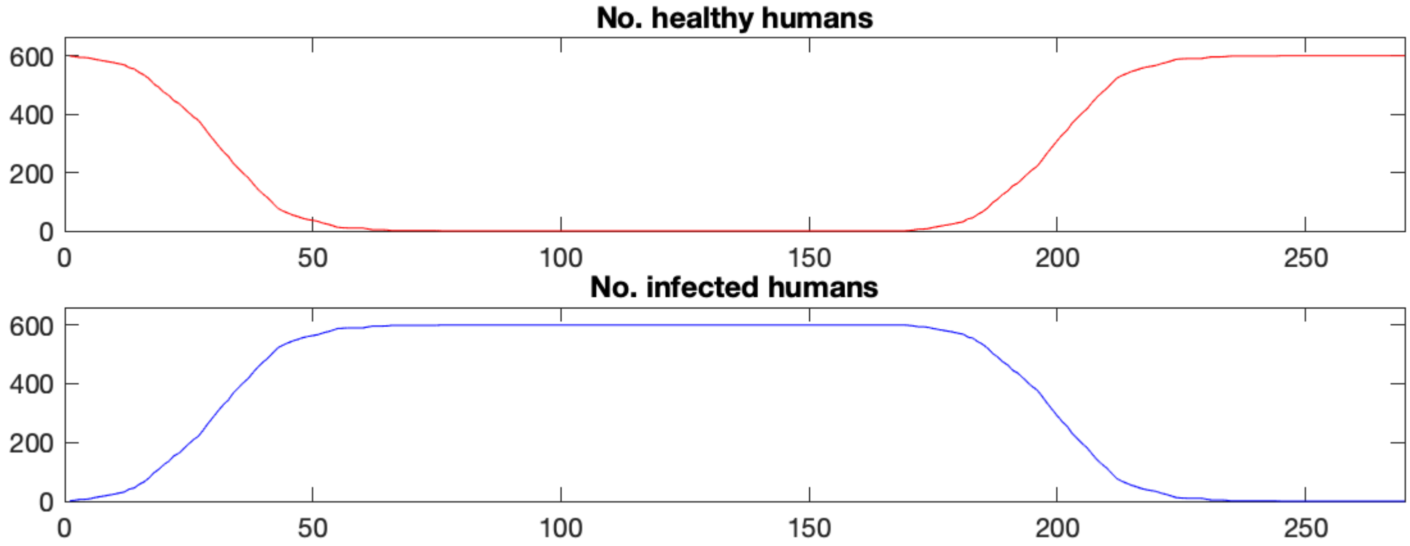


Figure B.14: *Graphs showing test 4 results for 600-agent simulation*

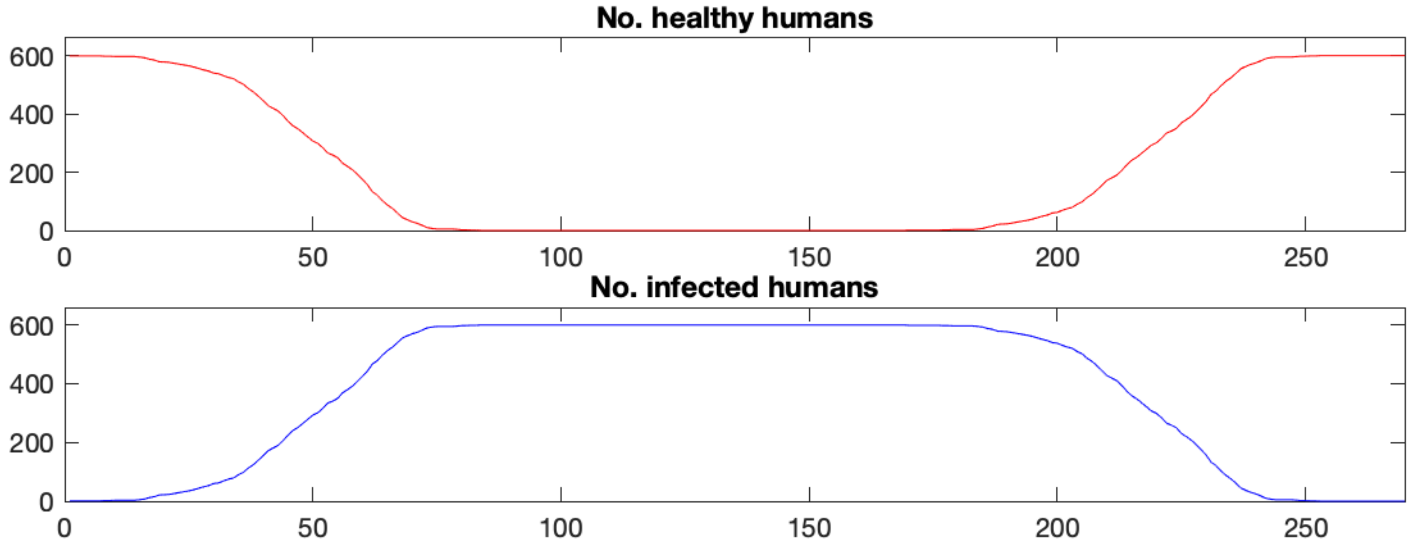


Figure B.15: *Graphs showing test 5 results for 600-agent simulation*

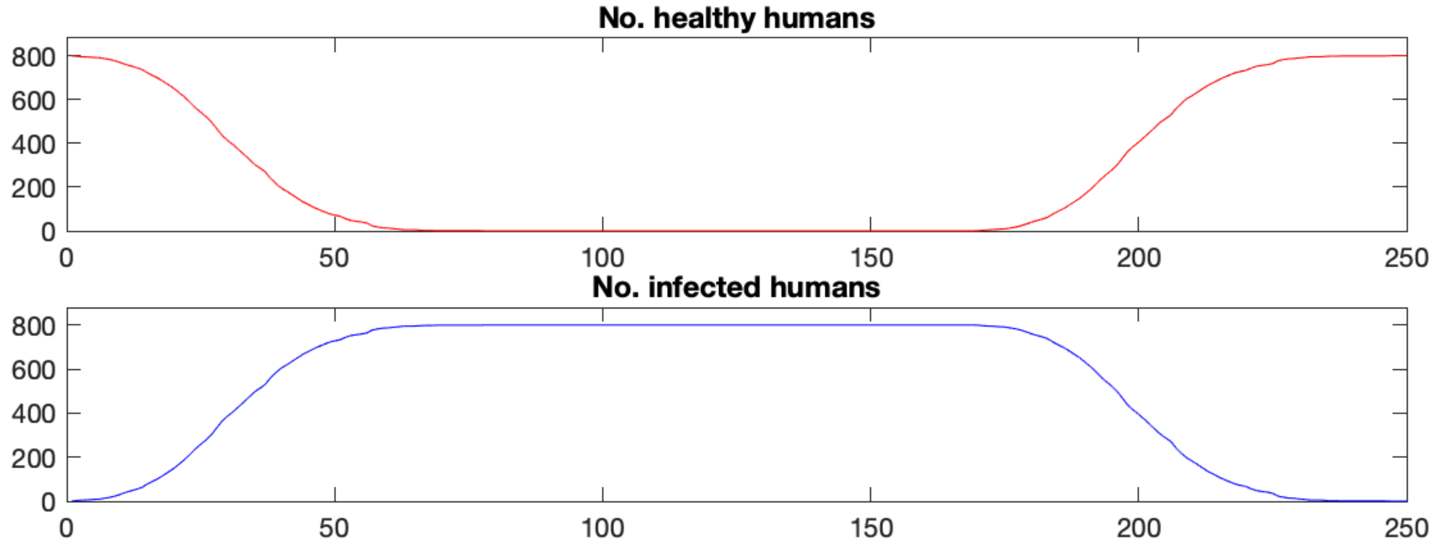


Figure B.16: *Graphs showing test 1 results for 800-agent simulation*

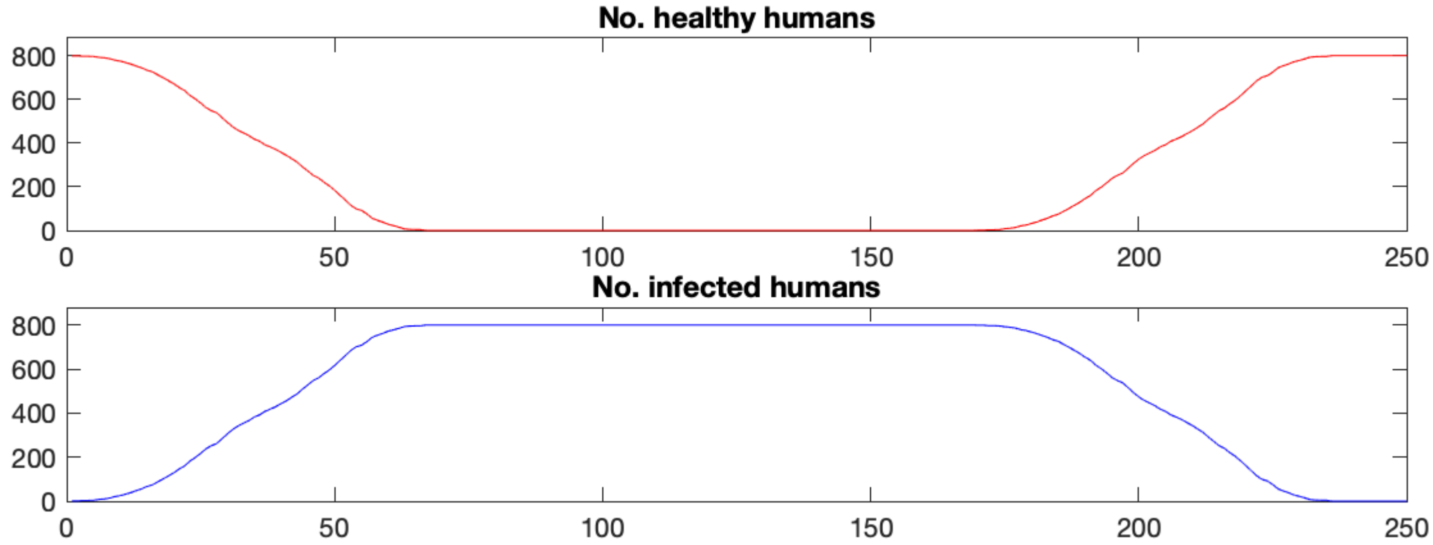


Figure B.17: *Graphs showing test 2 results for 800-agent simulation*

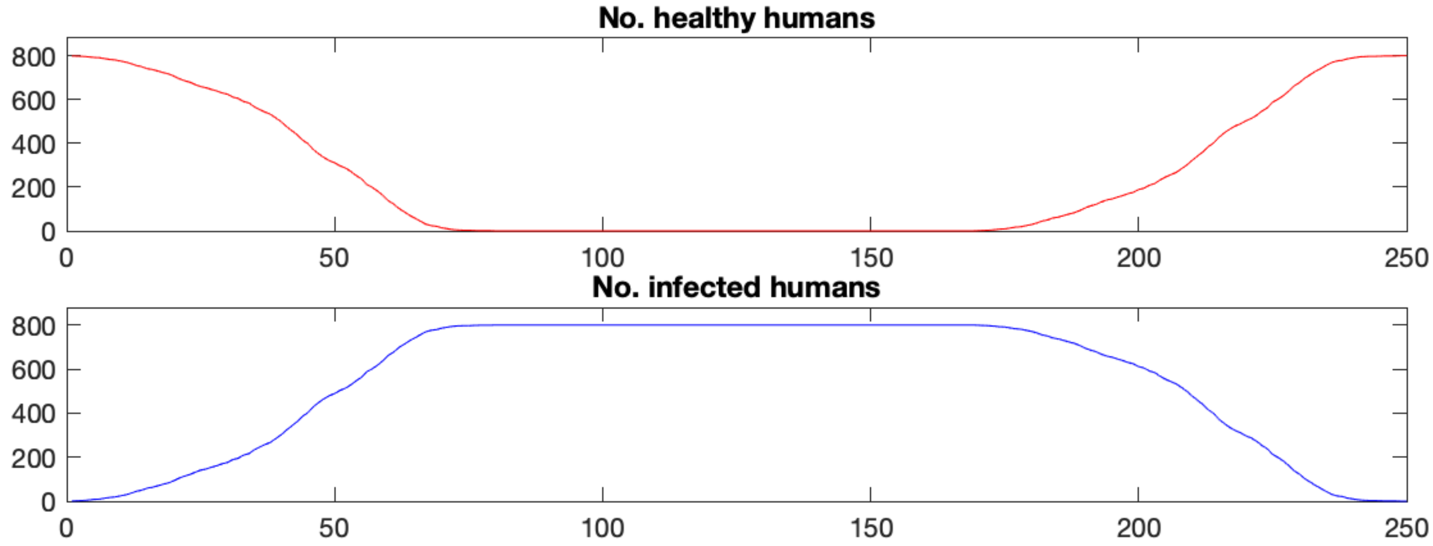


Figure B.18: *Graphs showing test 3 results for 800-agent simulation*

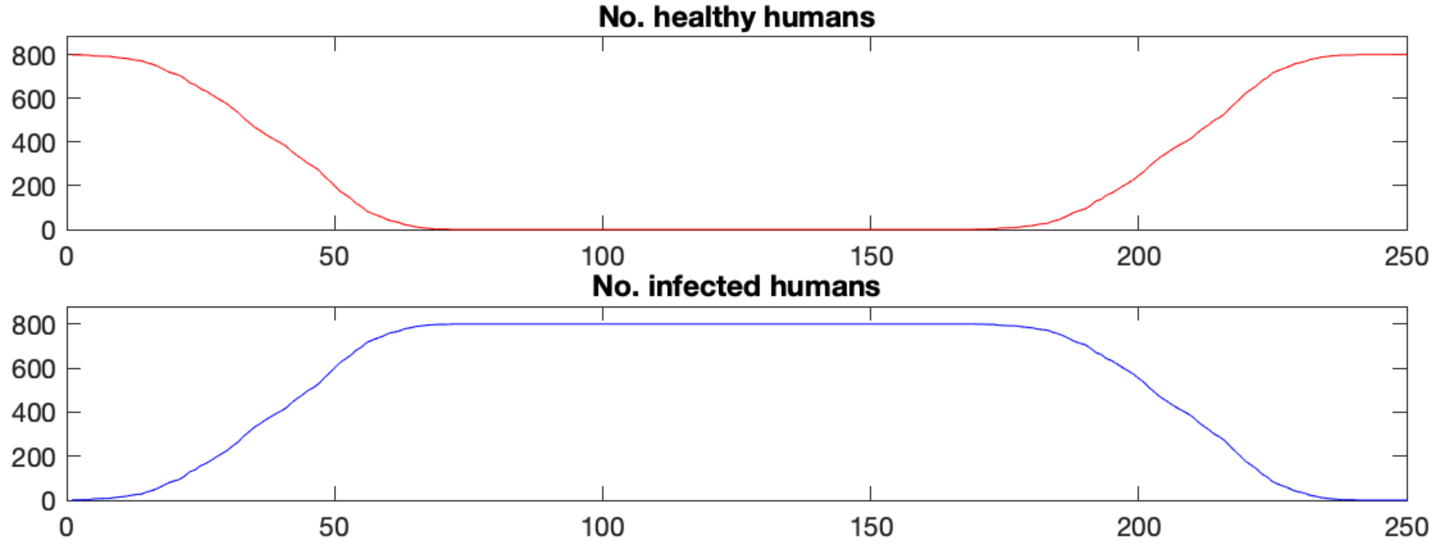


Figure B.19: *Graphs showing test 4 results for 800-agent simulation*

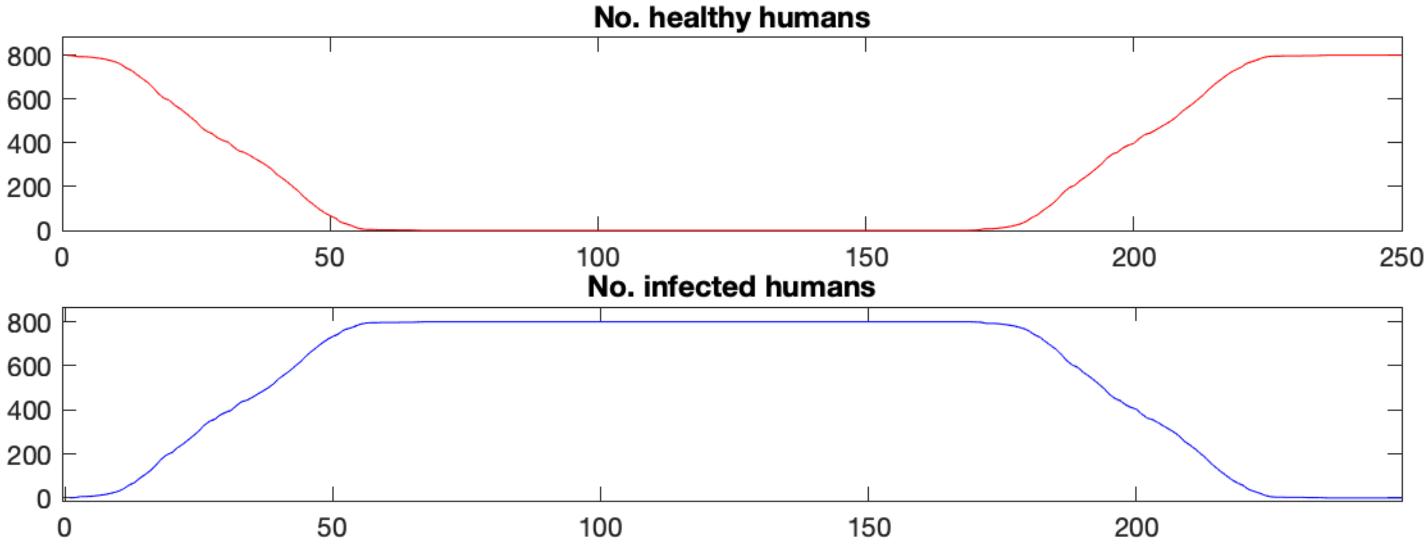


Figure B.20: *Graphs showing test 5 results for 800-agent simulation*

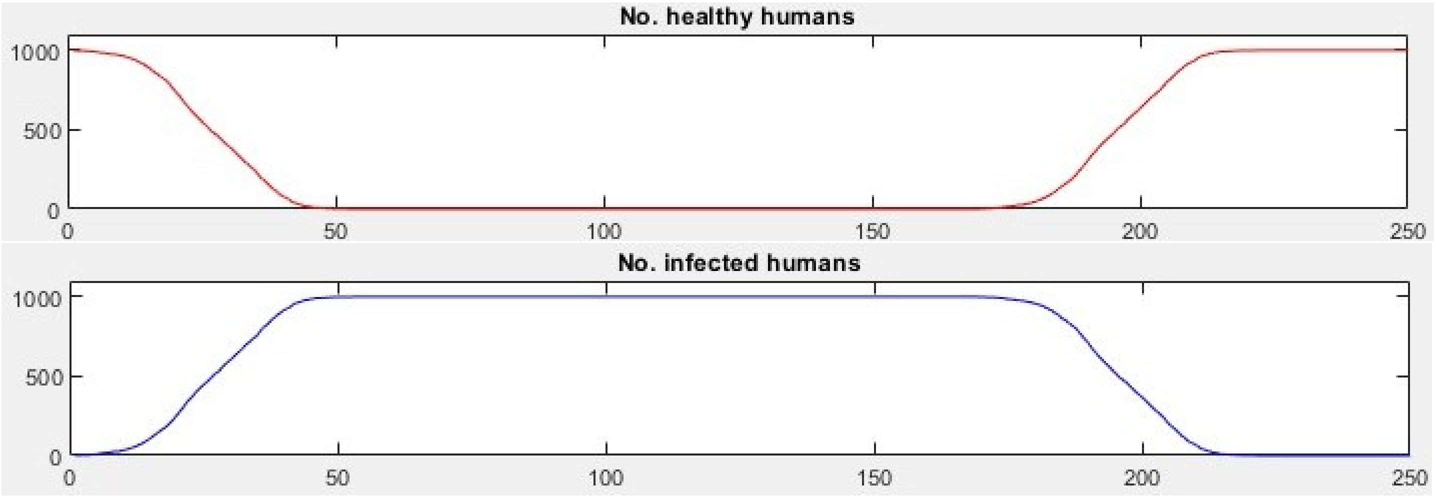


Figure B.21: *Graphs showing test 1 results for 1000-agent simulation*

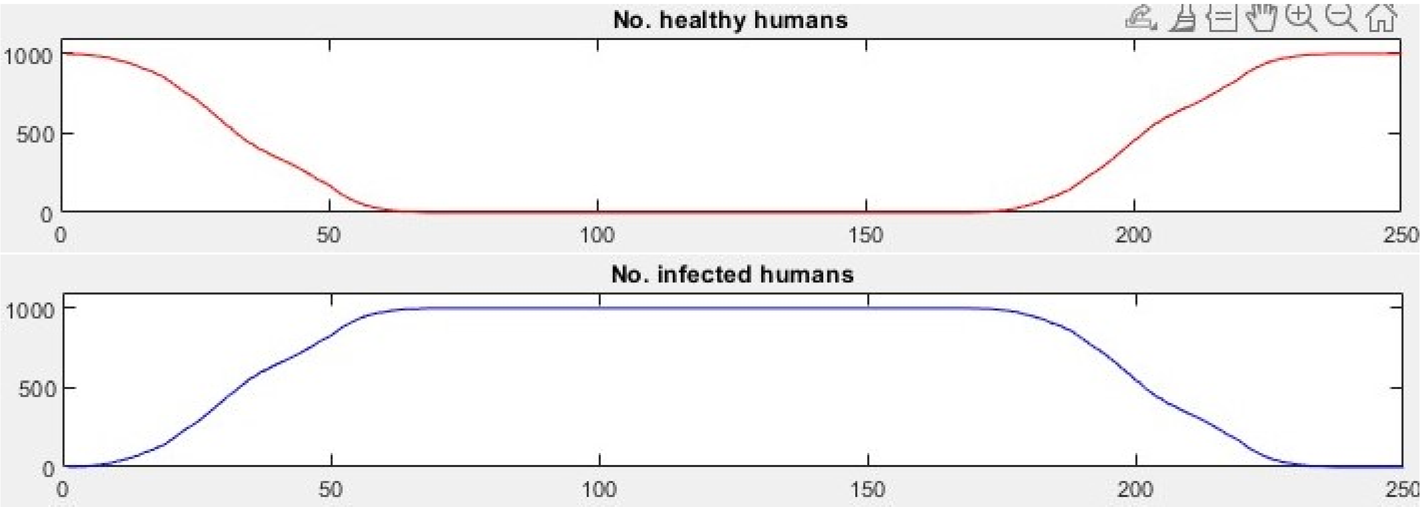


Figure B.22: *Graphs showing test 2 results for 1000-agent simulation*

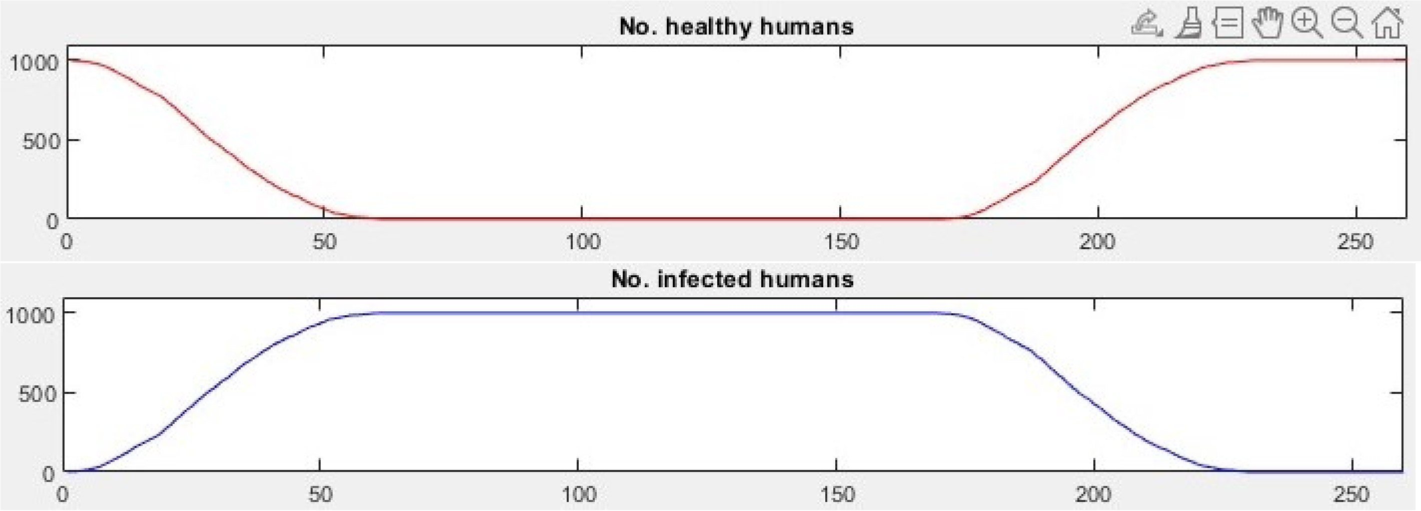


Figure B.23: *Graphs showing test 3 results for 1000-agent simulation*

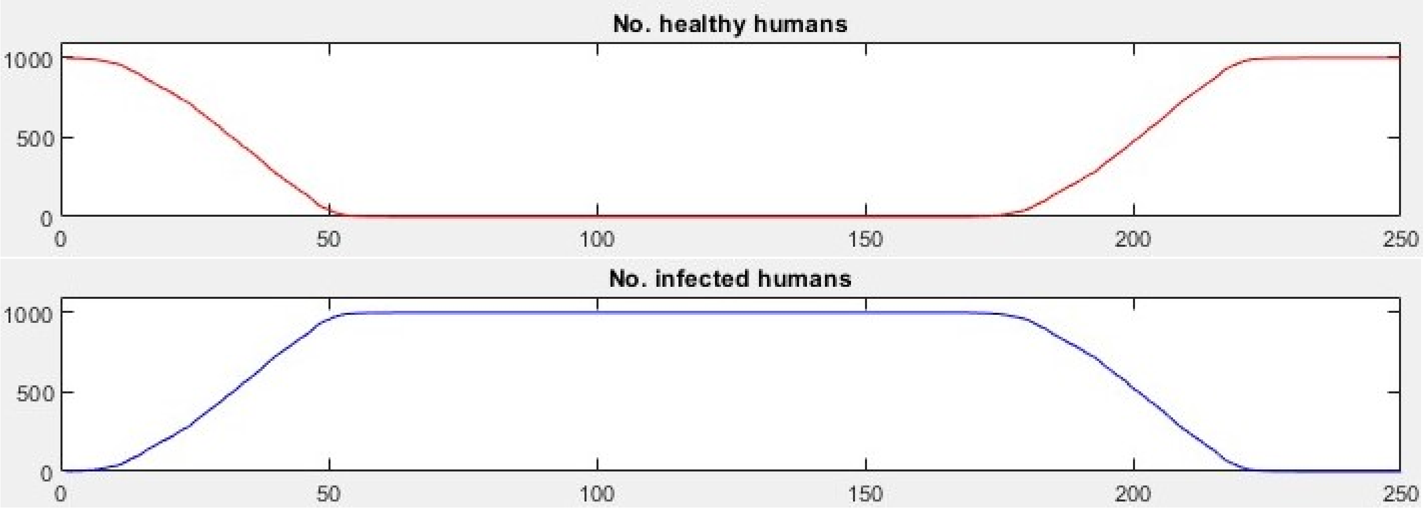


Figure B.24: *Graphs showing test 4 results for 1000-agent simulation*

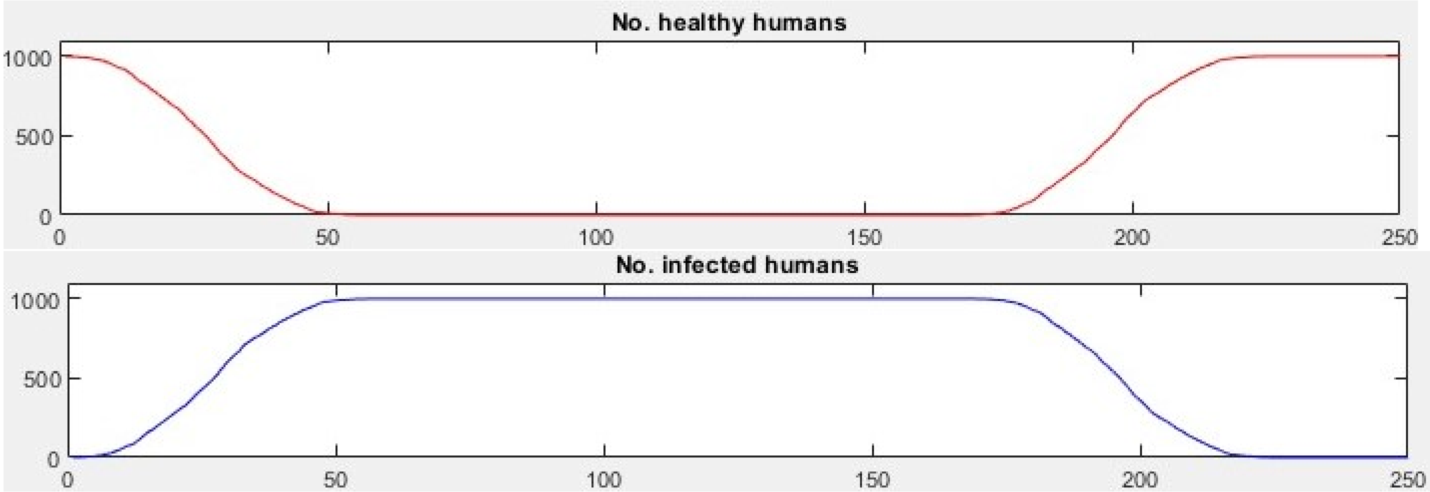


Figure B.25: *Graphs showing test 5 results for 1000-agent simulation*

Appendix C

Appendix

C.1 Computational Efficiency Full Results

Generations	1	2	3	4	5	Average
10	0.247322	0.291940	0.245259	0.255713	0.374782	0.2830032
100	1.487387	1.624970	1.449210	1.487880	1.410288	1.491947
1000	12.901532	13.066232	13.132690	13.278208	13.104579	13.0966482
10000	132.982569	130.760734	131.653043	130.469256	130.887811	131.3506826

Table C.1: *Results table for changing generations*

World Size	1	2	3	4	5	Average
10	1.408523	1.434864	1.487546	1.440227	1.461205	1.446473
50	1.528226	1.541222	1.486595	1.510810	1.566672	1.526705
100	1.441604	1.317518	1.315216	1.262738	1.316930	1.330801
500	1.131024	1.216711	1.228061	1.129568	1.154775	1.172028
1000	1.121564	1.272851	1.176770	1.166193	1.177519	1.182979
5000	1.104081	1.161773	1.115970	1.129447	1.109845	1.124223
10000	1.120321	1.184854	1.109625	1.126511	1.130825	1.134427

Table C.2: *Results table for world size*

Healthy Agents	1	2	3	4	5	Average
10	0.351019	0.262993	0.298993	0.264481	0.294605	0.2944182
50	0.819786	0.834075	0.796239	0.824778	0.847601	0.8244958
100	1.414506	1.476006	1.432185	1.480907	1.438207	1.4483622
500	6.578176	6.398603	6.468991	6.372733	6.515408	6.4667822
1000	13.280826	13.229499	12.896326	13.018739	12.999854	13.0850488

Table C.3: *Results table for changing the number of healthy agents*

Infected Agents	1	2	3	4	5	Average
10	1.440781	1.733948	1.522385	1.544336	1.609108	1.5701116
50	2.282771	2.062567	2.012185	2.196913	2.008409	2.112569
100	2.568244	2.586883	2.624446	2.584329	2.583496	2.5894796
500	7.440147	7.229176	7.192003	7..129472	7.173788	7.2329172
1000	13.910258	12.920455	12.876910	12.720854	12.932684	13.0722322

Table C.4: *Results table for changing the number of infected agents*