

Final Project Report

Introduction

For the final project, we are looking into how a disease may spread based on various factors. The disease propagation modeling algorithm is developed in C++ and is divided into two major component - creating a general disease propagation model (base case model) and additional customizations to observe how the disease spreads depending on three different factors:

1. Investigate when some members of the population are more susceptible to the virus than others.
2. Investigate how the disease propagates when a percentage of the population isolates after being exposed to the virus.
3. Investigate how asymptomatic cases contribute to the spread of the disease.

Implementation

The simulation of disease propagation model is developed based on two major classes: Population and Person. Additionally, various system variables are used to construct the simulation for the base case model. Such system variables for the **base case model** are set as the following:

System Variables

Field Name	Description	Value
pop_size	Population size used for the simulation	40,000
interaction	Number of interaction a person have per day	40
max_sick	Counter for the maximum number of sick people for a given simulation	Calculated in integer
d	Counter for the number of days	Calculated in integer

Person Class

Field Name	Description	Value
status	Current status of the person	String: "healthy", "asymptomatic", "infected", "vaccinated"
sick_days	The number of days someone will be sick when infected with the virus	5

Population Class

Field Name	Description	Value
<code>vector<Person></code> <code>People</code>	Vector of person objects with a size of "population_size"	<vector>
<code>population_size</code>	Number of people in the community	40,000
<code>day</code>	Current day of simulation	0 up to when the day when no one in the population is sick

To observe how the disease propagates when various factors are taken into considerations, we have created the following additional customizations:

Customization 1 - Asymptomatic cases

The first customization to the code was the possibility of asymptomatic cases. We defined these persons as people who were unaware of their positive infection status and would not be considered for isolating. For this simulation, we declared 2% of the population as asymptomatic. The following variables and methods were introduced:

Person Class	Population Class	
Methods	Variables	Methods
<code>void prob_sick()</code>	<code>vector<Person> Sick</code>	<code>int asym_count()</code>
This method was adjusted to give a person a 2% chance of being asymptomatic	This vector would now include people with not only an infection status of "infected" but also "asymptomatic"	Returns the number of currently asymptomatic people

Customization 2 - Susceptible members of the population

The second customization added to the code were the immunocompromised or "susceptible" members of the population. We estimated that about 25% of the population fits this description as ~3% are immunosuppressed, and ~22% are under the age of 12. This was implemented within the code by creating new functions and variables within the Person and Population objects as follows:

Person Class			Population Class	
Variables	Methods		Variables	Methods
<code>bool susceptible</code>	<code>bool get_susceptible()</code>	<code>bool set_susceptible()</code>	<code>vector<int> susceptible_People</code>	<code>int sus_count()</code>
Initialized to false	Returns susceptible status	Sets susceptible status to true	25% of population's probability of infection is 50%	Returns the number of currently susceptible people

Customization 3 - Isolation during infection

The final customization added the probability that some people are likely to isolate themselves when they are infected. We implemented the probability that about half of the infected people will isolate themselves. The following additions to the code were made:

Person Class			Population Class	
Variables	Methods		Variables	Methods
<code>bool isolate</code>	<code>bool get_isolate()</code>	<code>bool set_isolate()</code>	<code>vector<int> isolating_People</code>	<code>int iso_count()</code>
Initialized to false	Returns isolate status	Sets isolate status to true	Converts 50% of population isolate statuses to true	Returns the number of currently isolating people

Output file

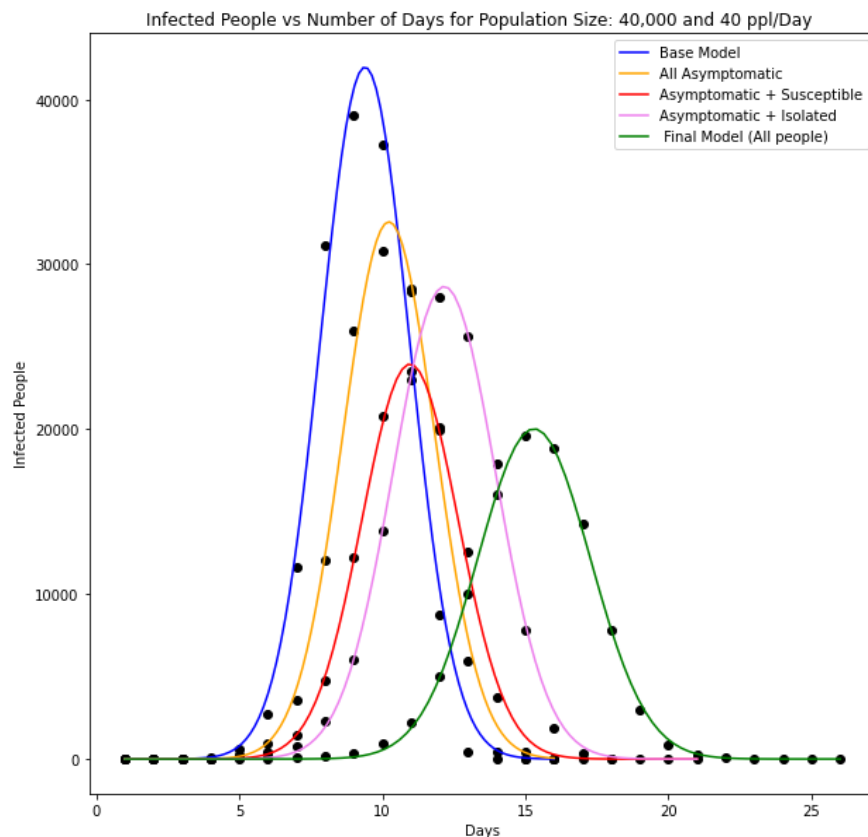
Name	Format
SIR_model_case1.csv	day sick count
asymptomatic.csv	day sick count
asymptomatic+susceptible.csv	day sick count
asymptomatic+isolate.csv	day sick count
all_simulations.csv	day sick count

Results

With all our simulations, a real world example would differ from our initial simulation, as there would be a larger number of additions that would need to be considered to model the true environment. However, by configuring the code to create different combinations of our code customizations, we can analyze how the number of days the simulation took to run until no one in the population is sick changes as well as the herd immunity percentages.

When running the each simulation for a population of 40,000 people with a daily interactions of 40 people per person, we get the following results:

Simulation	Days Until No One is Sick	Peak Number of Infections
Base case (No customizations added)	16	39,005
Adding Asymptomatic cases	16	30,794
Asymptomatic & Susceptible cases	18	22,989
Asymptomatic & Isolation cases	21	27,268
All cases combined	26	19,614



Conclusions

It is interesting to see that the number of days remained the same and the peak infection count decreased while interactions continued for asymptomatic people, but it may be fair since the asymptomatic population is only 2%. As for the Asymptomatic and Susceptible simulation, the minor increase in days can be attributed to the increase in probability of getting infected for this portion of the population. The increase in days allows for the simulation to have a wider spread, causing a flattening of the peak number of infected individuals.

The Asymptomatic and Isolation trial gave interesting numbers as well with an even longer number of days till the simulation ended but with a higher peak than the trial for Asymptomatic and Susceptible. The peak, however, is still lower than the peak for the base case. The number of days can be attributed to the fact that isolating people does not infect other people, so it takes the simulation longer to infect and recover the whole population.

The combination of all customizations is where we see the curve flatten the most. The number of days is over 50% longer than the base case, and the peak is about half. With the asymptomatic cases so low, the susceptible population's probability high, and the 50% likelihood of isolating when sick, the curve became flatter and wider.

In order to draw a conclusion that resembles real-life cases more accurately, further customizations and simulations will need to be added, so that a more realistic analysis of the flattening curve could be achieved.