CSE 6010 Final project report

Name: Zongyi Li, Zhiquan Zhang

GtID: zli732, zzhang749

Virus Spread and Disease Control Simulation

1. Project Overview

According to the reports on WHO official website, a deadly disease Ebola virus are still widely spreaded in some poor areas with little medical help[1]. Ebola Virus Disease (EVD) is a rare and deadly disease in people and nonhuman primates. The viruses that cause EVD are located mainly in sub-Saharan Africa. People can get EVD through direct contact with an infected animal (bat or nonhuman primate) or a sick or dead person infected with Ebola virus[2].

Our project is to build models to simulate the spread Ebola virus disease in cities in Africa, and assign a rescue team to traverse and give medical help to those cities. We firstly used SIRS model[3] to simulate Ebola virus disease spread in each city simultaneously. Then, we designed two routes for the rescue team can compared the effects. One route is based on the idea that we always assign the nearest city as the destination for the rescue team. And another is designed based on the idea that we always assign the city with the maximum number of infected people among all the cities at the moment the rescue team is about to leave, as the destination. And then we use Dijkstra algorithm[5] to find the shortest path and assign it as the route. While the rescue team reached and stays in a city, the process of virus spread will change in the period of time.

We calculated the total number of infected people in each city at each time under the influences of rescue team, and compared the change of number of infected people in two routes of rescue team, and drew some conclusions in part5. Also we did some experiments on tuning some parameters in virus spread model to see the changes of curves of number of infected people in one city, to see which parameters can mostly influence virus spread.

2. Models

2.1 SIRS virus spread model

In the simulation of virus spread in each city, we used SIRS virus spread model[4]. In this model, there exists three kinds of people: Susceptible, Infectious and Recovered. Susceptible is people that is easy to get infected but not confirmed infected. Infectious is people that is confirmed as infected, and Recovered is people that recovers from infectious. A person in the network should be one of these three types of people, and has specific possibility to be transformed into another. After being recovered from infectious, a person still have some probabilities to be infected again. The relationship between each kind of people is as below.

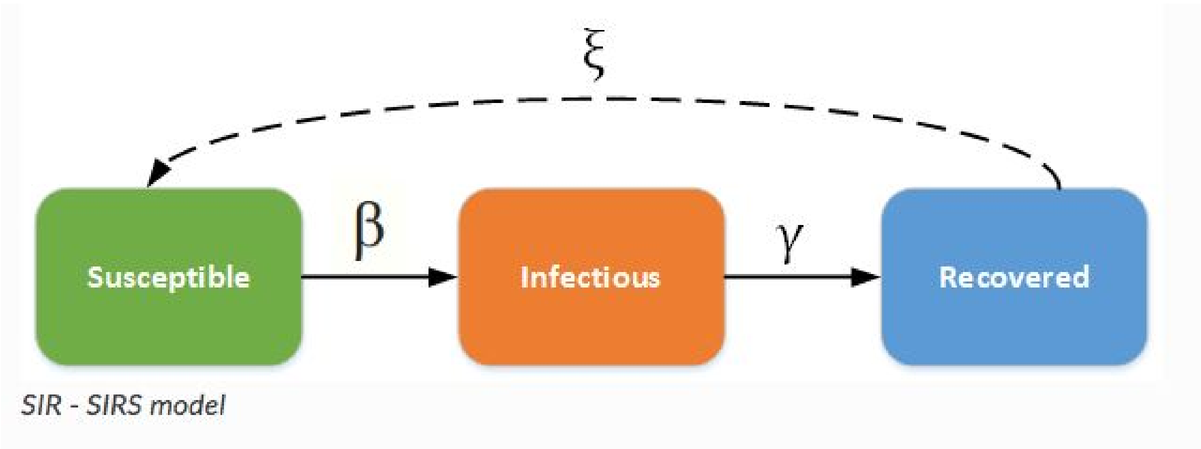


Figure1. SIRS model principle

For the possibilities above, β is Infection rate. γ is Recovery rate and ξ is the rate of being infected again.

We made an assumption that each city is a closed system, which means that the population in each city remains unchanged during the process of virus diffusion. Thus, we considered there's a social network in each city. Each person in a city can be described as a node in graph structure. If a person can connect with another, there's an edge between them. So the social network in a city can be described as an undirected graph.

So in a given time, we can model the spread of Ebola virus in all the cities simultaneously using the rules of transformation between these three types of people, and calculate the total number of each type of people at a specific time.

2.2 Disease control and rescue team design

We assumed that there's a rescue team to give medical help in each city. The rescue team will traverse between all the cities. While the city reaches a city and stays for a period of time, the infection rate and recovery rate will change in that time. The function of these two probabilities also depend on the times of having been visited by the rescue team. Firstly we randomly initialize the starting city, and designed routes for the rescue team. There are two routes for the rescue team.

1. Distance Mode

In Distance Mode, as the rescue team finishes giving medical help in one city and is about to leave for another, we assign the nearest city as the destination of the rescue team.

2. Emergency Mode

In Emergency Mode, as the rescue team finishes giving medical help and is about to leave, we assign the city with maximum number of Infectious as the next destination, and use Dijkstra algorithm to find the shortest path for the rescue team.

3. Implementation

The workflow of our project can be described as below. Firstly we read the configuration file which contains the number of cities, each city's population and location. Then we initialized all the structs and parameters in our project according to the files we read. Then we can simulate the virus spread process using SIRS model in two modes of routes of rescue team. Finally we output the statistics into files and do some experiments on tuning parameters of our models, and analyze the results.

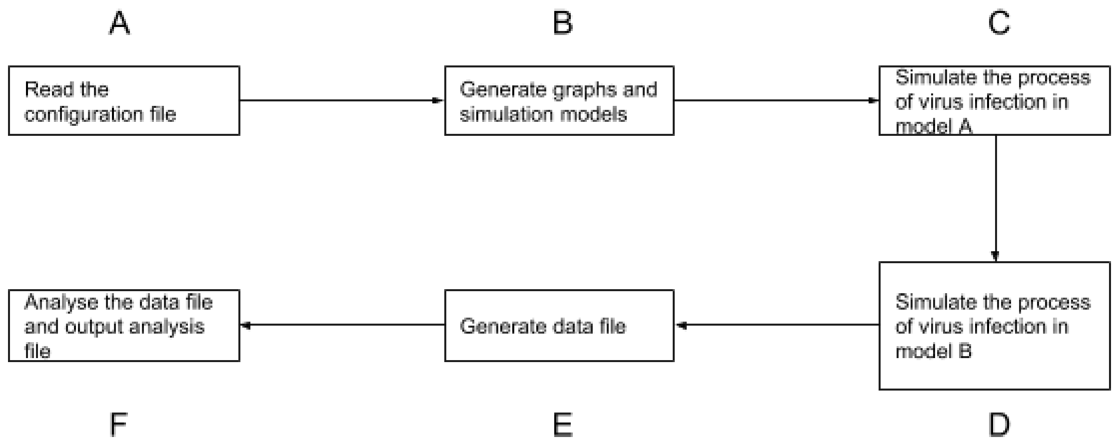


Figure2. Workflow of our project

3.1 Read configuration file

All source information is stored in the two configuration files. One file contains the very basic information of a city like the name of a city, the population of a city, the location of a city. There is a struct ***city,***which is used to restore the data read and used to simulate the virus circulation process.

As for another configuration file about path, we are supposed to read it and save the data into a double dimension matrix adjacency matrix. When we calculate the shortest path between cities, we should use this matrix.

Reading files basically is not a key part in this simulation part. And the configuration files' format is strict.

3.2 Intialization

For the initialization, we firstly defined two structs: City and Assistance.

For City struct, We firstly read the total number of cities, the population and location information of each city from existing files. Specifically, the location of a city includes latitude and longitude. In City struct, there are attribute parameters, which include population, is_rescued, start_time, **AdjacencyMatrix, *vertex and visited_time. Also there are some variables and arrays to record the information in SIRS diffusion. For the Adjacency Matrix in City, we initialized it according to the average contact rate and the population of that city. It's an undirected graph so the adjacency matrix is symmetric. Then we define an array for City struct to record all the cities information. Also, we calculated the distances between two pairs of cities using the latitude and longitude of them.

For Assistance struct, it contains parameters location, depart_time, arrival_time, speed and **VisitedCities. the location is the city with index 1 in the City array. The arrive_time is 1, and depart_time is 10, since the rescue time is 10. The speed of rescue team is 20. And we also initialized a HashTable to record all the visited cities and their visited times. The key is the index of city and value is the visited times.

3.3 Virus spread model

The steps of SIRS virus spread model can be described as followed.

1. For each city, randomly select a person as Infectious, and others as Susceptible. the infected rate for a city β is 0.6*(1-0.1*city[i].visited_time), and recovery rate γ is 1-β.
2. In the given diffusion time:
    a. For each Susceptible (node) in a city, traverse its neighbors and calculate the total number of infected neighbors as k. Then transform it to Infectious with the probability 1-(1-β)^k
    b. In the next step, for each person in the graph, set the Infectious as Recovered with probability γ, and Susceptible with probability ξ.
    c. Calculate the total number of Infectious for each city.

3.4 Disease control and route design

During the virus spread time, while the city is being rescued, which means that the current time is larger than the Assistance's arrival time and less than the Assistance's departure time, and the location of Assistance is just the index of that city the Infected rate β will become 0.4*(1-0.1*visited_time), and recovery rate γ is 1-β. And the visited_times of that city will add 1.

For Emergency mode, if the current time equals to Assistance's departure time, then record the index of city with maximum number of Infectious as the destination, and use Dijkstra algorithm to find the shortest path. Then we can calculate the total traveling time for the Assistance, and obtain the arrival time and departure time at the next city.

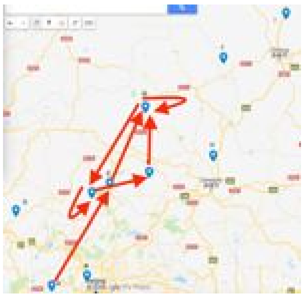


Figure3. Emergency mode

For Distance mode, if the current time equals to Assistance's departure time, then record the index of city that is nearest to the current city as destination, and calculate the traveling time according to the distance between them, and obtain the arrival time and departure time at the next city.

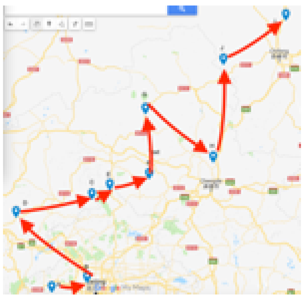


Figure4. Distance Mode

3.5 HashTable

Considering the fact that a city can not be visited twice before all cities are visited once, we need a Hash Table to implement this function. A Hash Table can implement basic functions like:

Initialization	Create and initialize the Hash Table
Size	Return the number of the elements
Clear	Delete all the elements in the Hash Table
Insert	Insert elements in the table
Delete	Delete special elements in the table
Search	Search elements in the table

The Hash Table is based on array data structure. The hash function is the value of key module the parameter size, which is an initialized parameter when creating a hash table. This hash table's key is integer type and the data it corresponds to can be of any type.

In the program, the Hash Table can be utilized in the distance mode. It means that every city can have a member Hash Table, which records all the cities that have been visited. Once the number of the visited cities reach ten, we can use clear function to help remove all the elements.

3.6 Dijkstra algorithm

This algorithm here is based on the adjacency matrix. It can calculate the shortest path between the source city and destination city. Note that there at least exists one path between any two cities in the map. The concrete implementation of this algorithm can be referred in many websites so it will be omitted here.

4. Data Rescources

4.1. Ebola virus disease data

We downloaded the Ebola virus disease report from WHO official website, and found the number of total population, confirmed infected people, and recovered people in cities with high infected rate in Africa.

2. City location

We used Google Earth to find the latitude and longitude of each city.

3. City population

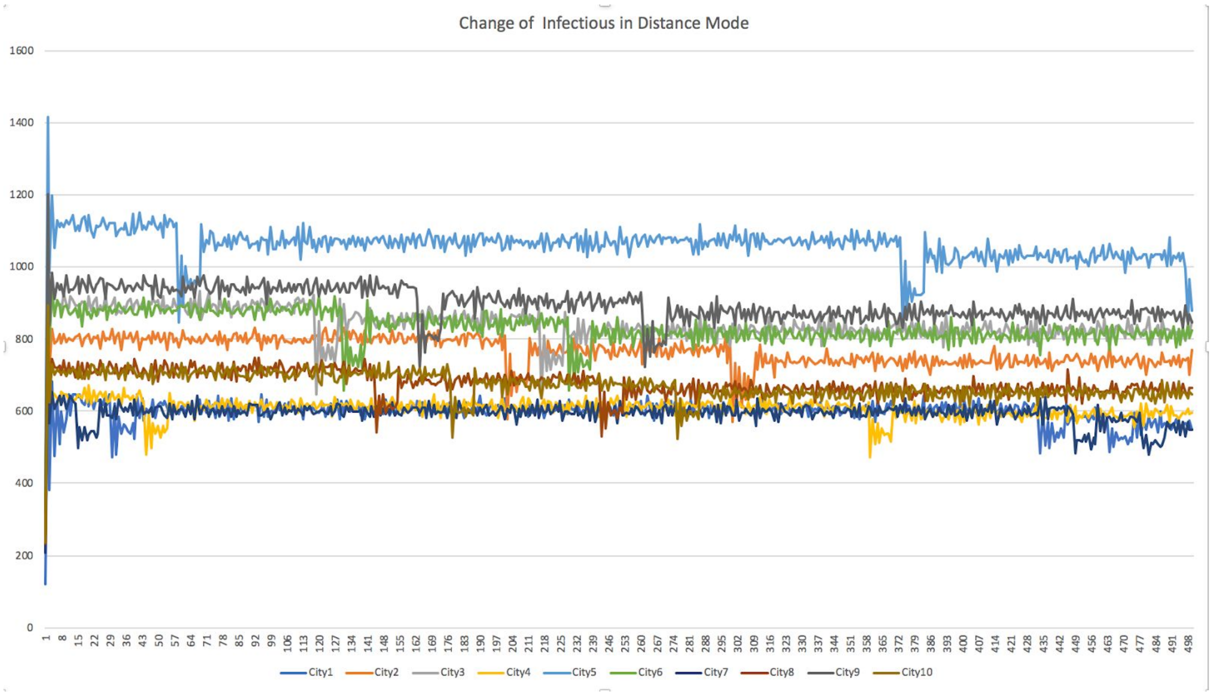
The cities' population is not the actual number of the total. It's based on some special group like woman, children or old man over 70. Because virus only spread in a special group.

5. Experiments and Analysis

5.1 Results of two different modes of route design

We input 10 cities' population and location information, and set the average contact rate as 0.4, total virus diffusion time as 500, and rescue times as 10, the speed of rescue team as 20. While not being rescued, infection rate β is 0.6*(1-0.1*visited_time), and recovery probability γ is 1-β. While the city is being rescued, the infection rate β of that city is 0.4*(1-0.1*visited_time), and Recovery probability γ is 1-β.

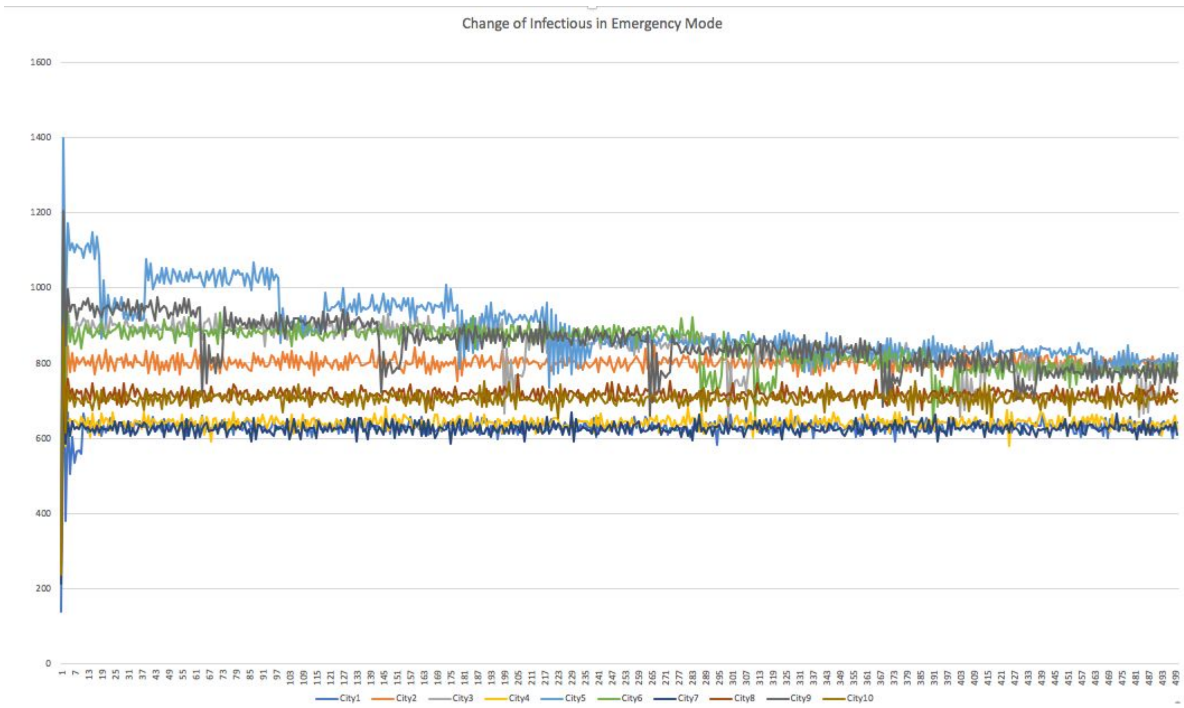
The curves of change of total Infectious with time of all the cities in Distance mode are as below.



Figrue5. Results of curves of total number of Infectious in Distance mode

We can see that with with medical help, the number of infected people of a city during the rescue time will obviously decrease. But after the rescue team leaves the city, the number of infected people will increase to some high level and change slightly.

The curves of change of total Infectious with time in Emergency mode are as below.



Figrue6. Results of curves of total number of Infectious in Emergency mode

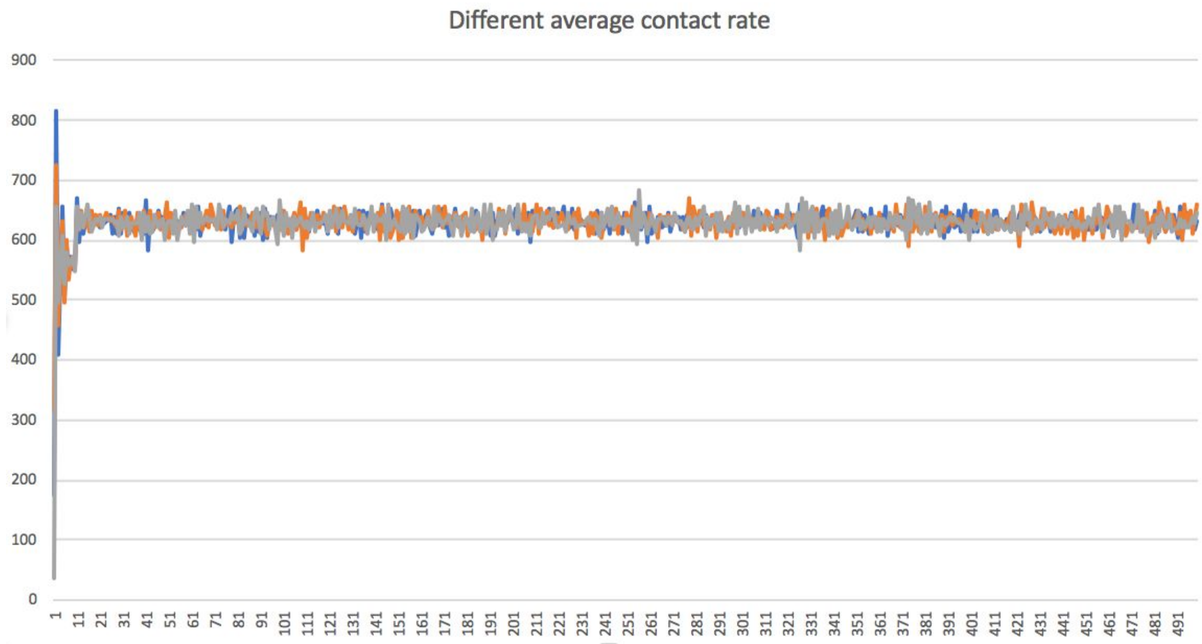
We can see from the results that in Emergency mode, the total number of infected people will decrease more rapidly compared with Distance mode (it's obvious according to the result of city1). Thus, we can draw a conclusion that in the same condition, Emergency mode appears to have better disease control than Distance mode.

5.2 Experiments on different rate parameters of one city

Then we did some researched on the influences of different parameters in our models. Specifically, we studied the virus spread principles on one city, to see which parameters can have influential effects.

5.2.1 Different average contact rate α

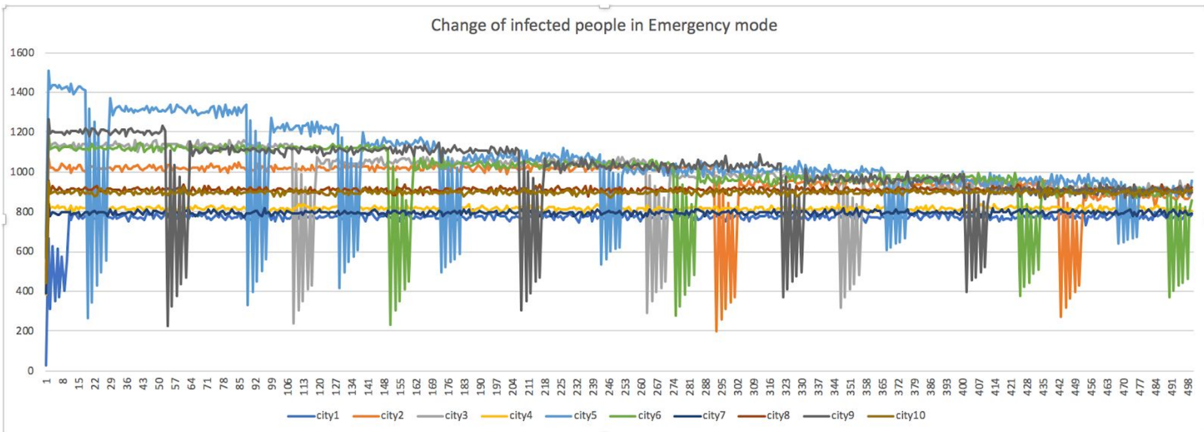
We set the average contact rate α as 0.1, 0.5 and 0.9. And other parameters remain the same in 5.1. The results of curves of change of number of Infectious in city1 in Distance mode is as below. We can see that average contact rate α will not have much influence on the total number of infected people in a long period time, since while the process of virus spread tends to be stable, the number of infected people will remains at a certain level. But initially at the early stage of diffusion, if the average contact rate α is large, the number of infected people will increase rapidly to a higher level, then decrease(as the blue line showed as below, which is the results of α=0.9).



Figrue7. Results total number of Infectious in Emergency mode with different α

5.2.2 Different Infection rate β

Then we set β as 0.1*(1-0.1*visited_time) while the city is being rescued, and 0.9*(1-0.1*visited_time) while it's not being rescued. The curve results in Emergency mode is as below.

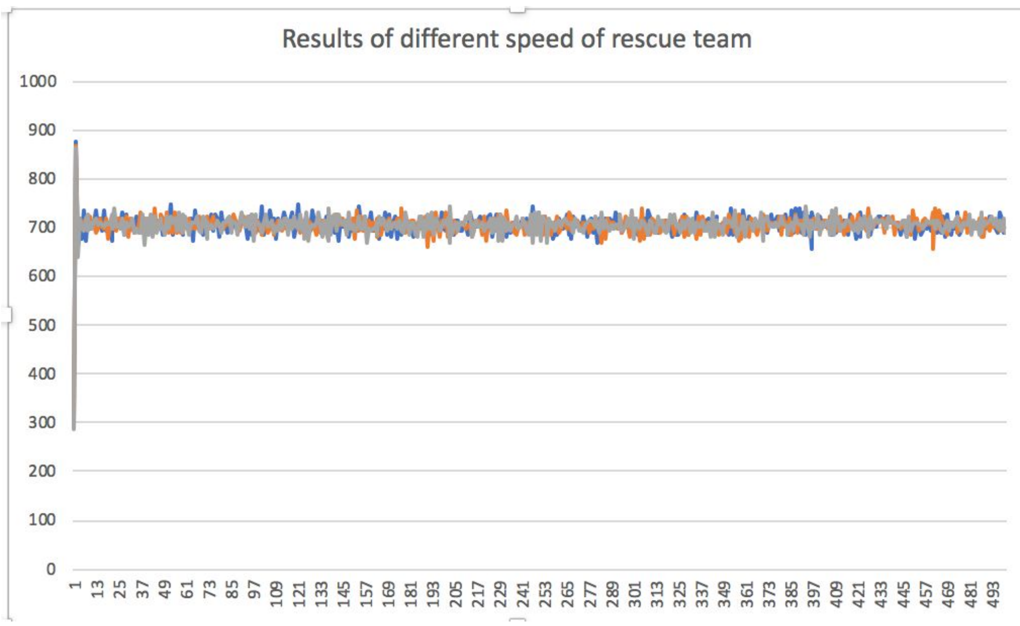


Figrue8. Results of total number of Infectious in Emergency mode with different β

We can see from the results that infection rate β can obviously influence the results. While being rescued, the city with lower infection rate will tremendously decrease the number of infected people, but it will soon increase to a high level after the rescue team leaves. And city with high β while not being rescued will have larger number of infected people in the diffusion process.

5.2.3 Different traveling speeds of rescue team

Then we set different traveling speed of the rescue team to see the changes. We set as 10,20 and 30. The The curve results in Emergency mode of city4 is as below.



Figrue9. Results of total number of Infectious with different speed

We can see from the results that the speed of traveling will not have much influence on the curves of number of infected people in a long period of diffusion process.

References:

[1] World Health Organization. "WHO: Ebola response roadmap situation report 15 October 2014." (2014).

[2] WHO Ebola Response Team. "Ebola virus disease in West Africa—the first 9 months of the epidemic and forward projections." New England Journal of Medicine 371.16 (2014): 1481-1495.

[3] Li, Michael Y., et al. "Global dynamics of a SEIR model with varying total population size." Mathematical biosciences 160.2 (1999): 191-213.

[4] Hethcote, Herbert W., and P. Van den Driessche. "Some epidemiological models with nonlinear incidence." Journal of Mathematical Biology 29.3 (1991): 271-287.
[5] Brandes, Ulrik. "A faster algorithm for betweenness centrality." Journal of mathematical sociology 25.2 (2001): 163-177.