Virus Spread and Disease Control Simulation

Team member: Zongyi Li, Zhiquan Zhang

Overview

Our project is to build models to simulate the process of Ebola virus disease spread in cities in Africa. And assign rescue team for different cities.

1. Used SIRS model to simulate virus spread in different cities.
2. Assigned a rescue team to traverse these cities. Virus spread will be influenced in the city while it's being helped. Designed two different routes for the rescue team.
3. Calculated the total infected people in given time through these two routes of rescue team.

Models

- SIRS virus spread model

Susceptible: Easy to get infected
Infectious: Confirmed infected people
Recovered: Recovered people
$\boldsymbol{\beta}$: Infection rate
$\boldsymbol{\gamma}$: Recovery rate
$\boldsymbol{\xi}$: Rate of being infected again
$\boldsymbol{\alpha}$: Average contact rate
visited_times: times of having been rescued
While the city is being rescued, β will become β*(1-0.1*visited_times), γ will become 1- β.

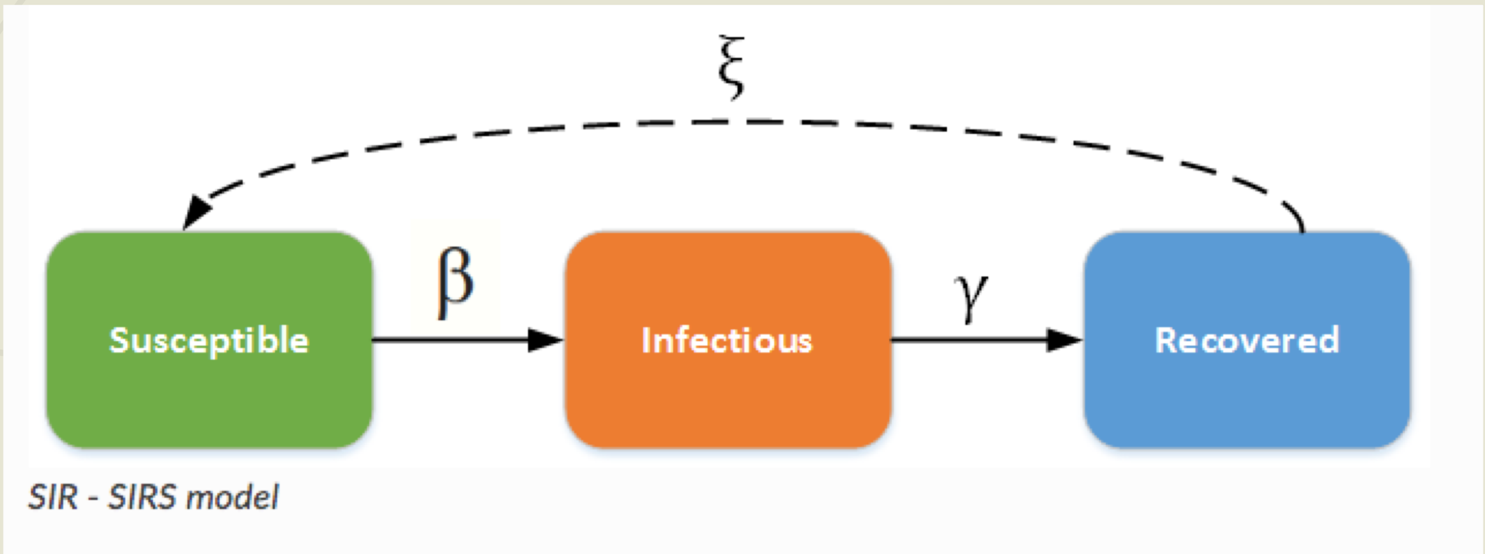


Figure 1. SIRS virus spread model

Spread Process:

1. Initialize the Adjacency Matrix according to population and α.
2. At start time, randomly select a person as Infectious, others as Susceptible.
3. In given times:

a. for each person, calculate the number of infected neighbors. Set it as infectious with probability $1-(1-\beta)$^**k.**

b. In the next step, set the person as Recovered with probability γ, and Susceptible with probability ξ.

4. Calculate numbers of different types as each time in the loop.

- Rescue team route design

1. Distance Mode

Always assign the closest city as the next destination

2. Emergency Mode

Assign the city with largest number of infected people at the time rescue team is about to leave as the next destination. Used Dijkstra algorithm to find the shortest path.

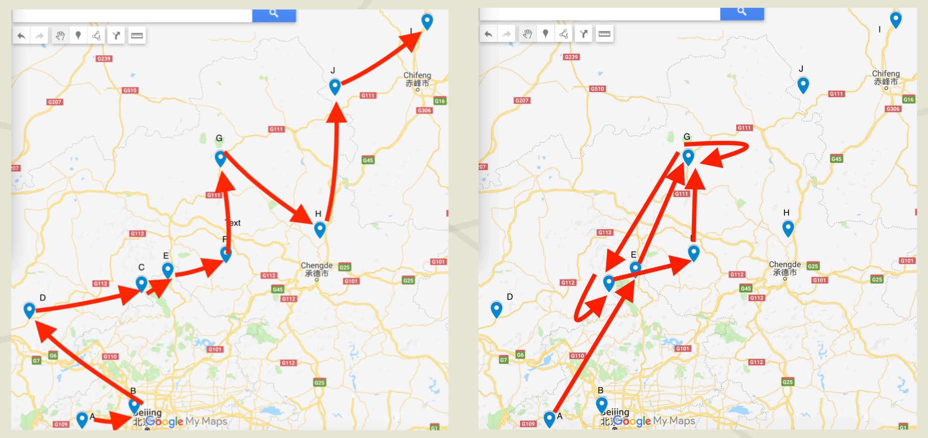


Figure 2. Two route designs for rescue team

Implementation

Main data structures and algorithms used in our project:

1. Built adjacency matrix for people connected in different cities.
2. Built Rescue team and City struct to record the information.
3. Wrote HashMap in C based on array. Clear(), size(), insert(), ect.
4. Used Dijkstra algorithm to find the shortest path in the second mode.
5. Calculated the distances between cities using magnitudes and longitudes of the cities.

Data

1. Ebola virus disease data

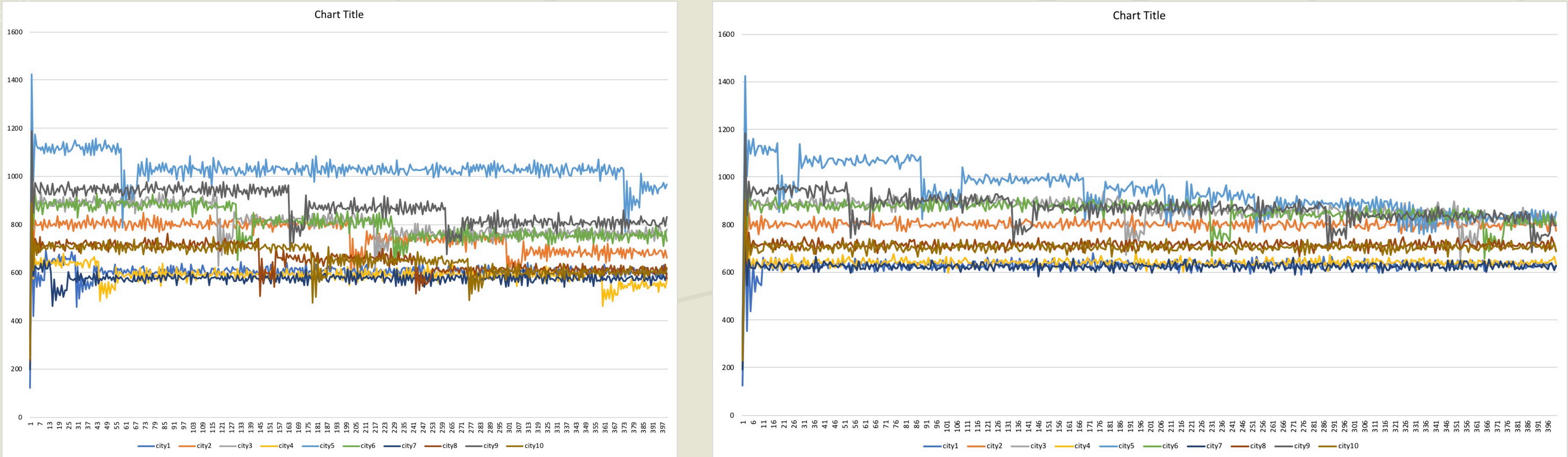
We downloaded the Ebola virus disease report from WHO official website, and found the number of total population, confirmed infected people, and recovered people in cities with high infected rate in Africa.

2. City location

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

Analysis

In experiments, we set average contact rate α as 0.4, infection rate β and ξ as 0.4, recovery rate γ as 0.7. And we set spread time as 1000. We calculated the total number of infected people at each time in each city for the routes in two routes.



(a) Distance Mode (b) Emergency Mode

Figure 3. Change of number of infected people in 10 cities in given time

In the next step, we will tune the parameters (different probabilities in our virus spread model) and analyze the effects of two modes on single city and compare their efficiencies.

Expected results:

Different cities will have different number of infected people after a period. And Emergency Mode will be more effective.