UNIVERSITY OF SWAZILAND

Modeling Of disease Propagation Using Percolation theory

by

Hlophe Nkosinathi,143146

A project submitted in partial fulfillment of the
Bachelor Of Science

in the

Faculty of Science
Department of Physics

November 2019

Certification

This is to certify that the work in this project was carried by Hlophe Nkosinathi in the department of physics, University Of Swaziland

Name of supervisor: Dr. S.K Mkhonta

Signature of supervisor:

Date: November 2019

Department Stamp

"I simple wish that, in a matter that so closely concern the well-being of mankind no decision shall be made without all the knowledge which little analysis and calculation can provide"

Daniel Bernoulli

Abstract

In this paper we study the spread of diseases using percolation theory. The spread of disease can be simplified by a model created to study the interaction that represents transmission dynamics in real world systems. The fitting and testing of the model was done using data from previous epidemic of Ebola In Equatorial Guinea in 2014. The model was generated using a computer code written in python programming language. The bigger picture of this work was in identifying disease control strategies that deflects epidemics from invasive to non-invasive regions at low effective cost and also be able to predict future disease out-break. The main contributor to disease spread is the immunization rate and for this model the critical immunization rate was found to be 68%.

Acknowledgements

I would like to thank every member of my family especially my mother and father who supported me through my university life and life in general.
To my supervisor Dr S.K Mkhonta I do not have enough words to describe my deepest gratitude for advice, ideas and suggestions he made in making this project a success and sacrificing his time to concentrate on assisting in making this project a better one. Not forgetting the department of Physics for the opportunity they gave me to do a project in the department.
Lastly I would like to thank my colleagues whom we spent sleepless nights, without their inputs, advice and encouragement this project would not been successful. To all my friends I thank you for their inputs, support and advice especially those who proof read this project. I thank you all.

Contents

List of Figures

Abbreviations

EBOV	**Eb**ola **V**irus disease
P_c	Percolation threshold
$\mathbf{P}(\rho)$	Percolation probability
SIS	**S**usceptible **I**nfectious **S**usceptibl
SOC	**S**elf **O**organized **C**ritically
WHO	**W**ord **H**ealth **O**rganization

Chapter 1

Introduction

1.1 Complex many body interacting systems

Many things in the world and universe around us consists of large number of interacting particles and appear to be extremely complex. For example the enormous number of atoms which make up a solid system.

In many cases collective behavior can be predicted, interpreted or modeled by considering the interactions between neighboring particles or entities. For example, the behavior of granular matter e.g. a box marbles, can be predicted by considering local excluded volume interactions.

1.2 Percolation theory

Percolation theory: Is an approach to model complex many-body interacting systems by simplifying the interactions between individuals, allowing the systems to evolve from random initial configurations. The interacting system is usually represented by a sites in spatial grid and the interactions limited to neighbors only. Percolation theory can be applied to any system which can be reduced to very simple interaction between individuals.

Network percolating model is an abstraction of many practical problems while the spread of diseases can be described by percolation models. The problem of spread of disease has been the focus of many academia and medical. In recent years the complex network is a reflection of the effective means to study realistic networks.[1]

There is increase interest in the use of percolation models to analyze and predict the progress of disease spreading in closed structured population of animals and plants. The wider utility of approach has been limited by several restriction assumptions which are made to incorporate the diversity of variables which contribute to spread through contact of nearest-neighbor transmissions, the choosing of a suitable lattice and long range dispersal.

1.3 Applications for percolation theory in Physics

- **Magnetism:** Dilute magnetic semiconductor such $Cd_{(1-x)}Mn_xTe$ consists of magnetic ions (Mn) in a background of nonmagnetic particle. In a quenched system the Mn ions like to align their spin anti-parallel and cannot interchange places with their non-magnetic counterpart. Hence the magnetic behavior of this system depends fraction x of the magnetic ions as well on the spatial arrangement of the Mn ions. In a real system the Mn ions are arranged in random clusters in the semiconductor sample. In this case percolation theory can be applied to determine the critical concentration x_c above which the semiconductor changes from being para-magnetic (non-magnetic state) to ferromagnetic state(magnetic).

- **Diffusion in disordered media:** Percolation theory has also been utilized to determine the electrical conductivity of disordered semiconductors and for a model of switching in amorphous semiconductors. An experiments used for testing the conductivity of these semi-conductors is using a conducting paper with holes randomly punched in it as shown in figure 1.1.The results of the experiment are shown in figure 1.2(a), which is a plot of the ratio of

the conductance of the paper to its initial value against the concentration of holes. It is found that the conductivity drops to zero much less sharply than P(ρ) near the critical concentration of the holes which is 0.60 for open site.[4]

- **Disease or information propagation in a random network:** The picture of electron hopping through a random network of conducting region in a disorder conductors is applicable to the dynamics of disease propagation. Disease propagates in a random network of susceptible and non-susceptible (immunized) individuals. In regions of susceptible is isolated in sea of immunized individuals, the disease propagation will be limit to finite regions. In this work it explores the problem of disease propagation.

1.4 Disease Modeling

This research is based on disease modeling sorely because many disease take long time for scientist to understand how they propagate and this means it is going to take some time to come with effective methods of containing or stop the spread. This helps in devising better strategies of fighting these diseases and to be able to prepare us to tackle future outbreaks of a certain disease A lot of money is spent trying synthesize vaccine and distribution of the vaccine or medication to the affected population. Disease modeling help in understanding how different diseases spread and how best can the spread be stop or minimized using minimal resources, that is finding the critical immunization probability of the disease.

1.5 Studies where percolating models were used to study disease spread

- **Modeling pathogen and immune responses to treat tuberculosis:** Current therapy is difficult for patients to adhere to, as it requires 6 months of

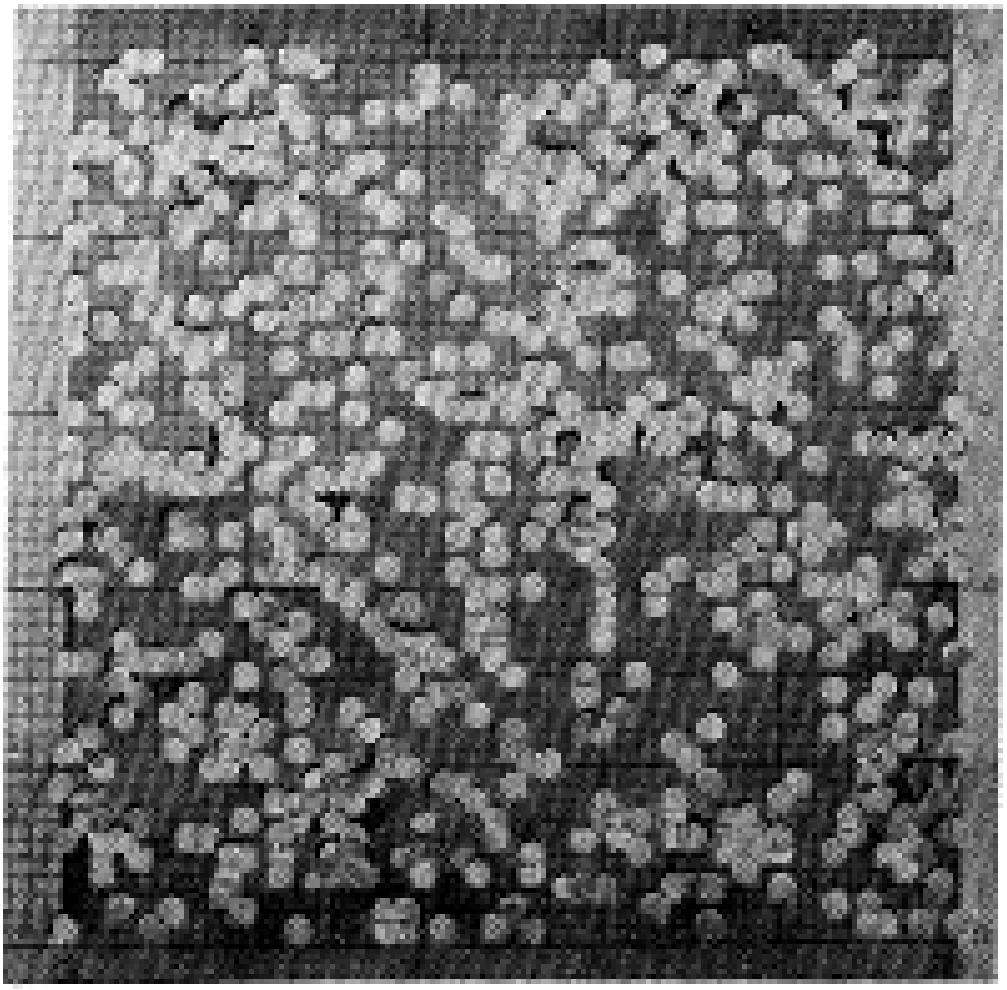


FIGURE 1.1: *Photograph of the sheet of conducting paper at the stage where the concentration of holes is 0.268*

treatment with multiple antibiotics. Computational approach was developed to match realistic model that describes the immune response to infection with the bacteria that causes tuberculosis. Results from the model is used to develop improved therapies that optimize the combined effects of antibiotic treatment and the immune response.

- **Treatment of heart disease:** Models of the mechanics of blood vessels, blood flow and and heart valves have been developed using percolation theory. This model is used to optimize the design of implanted devices such as artificial heart, valves and coronary artery stents.[8]

- **Optimizing conditions for tissue repair** Researchers are developing a modeling toolkit that would predict the best bio-material composition for

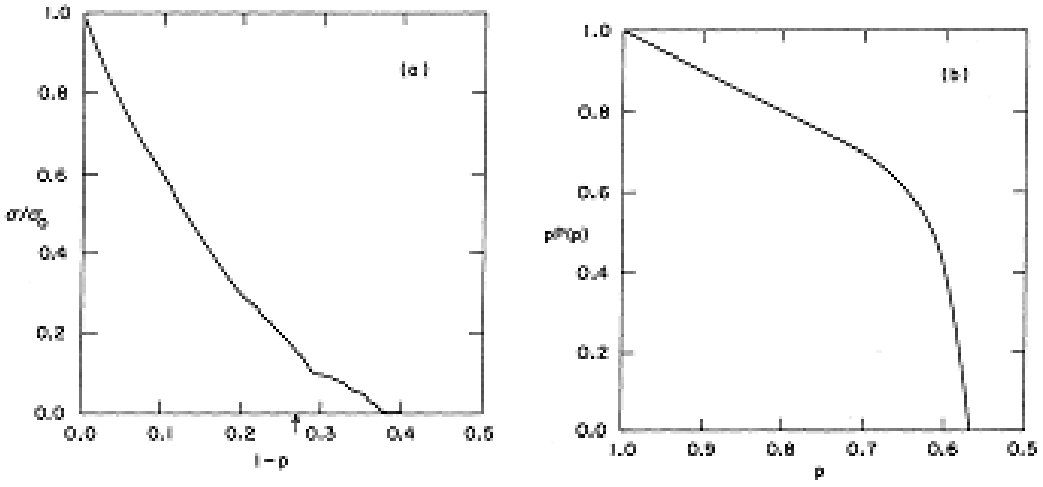


FIGURE 1.2: *(a) Graph of the conductivity as a function of the concentration of holes (1-ρ). The bulk conductivity of the conducting paper is σ0. The arrow shows the point at which the photograph of Fig 1.1 was taken, (b) Graph of ρ(P)ρ for the site problem on the square lattice, where is the concentration of open sites*

different biological environment such as in different parts of the body. The current focus is on load-bearing bio-materials for repair of bone and related tissues.

1.6 Tools

Python programming language was used. Python version 2.76 software was installed in windows operating system. Python is an easy language to learn and understand, it is free on the market. It is also good in making contour plots simulations and graphs and these all is done using a few libraries.

1.7 Disease used in comparing with model

The 2014 Ebola virus disease(EBOV) outbreak in west Africa is the largest of the genus Ebola virus to date. The EBOV make a good choice as the threat of this epidermic is increasing each passing day and the panic from Africa and the

world is increasing. The outbreak began in Guinea December 2013 and it later spread to Sierra Leon, Liberia and Nigeria. The number of infected individual continue to rise even though public health intervention have been introduced. The absence of control at present mean we should look for other means of trying to minimize the outbreak which one of them is through implementations of results from computational models.[3]

1.8 project goals

The goals of this project are summarized below:

1. Develop a python code to simulate disease spreading in a random network of susceptible population.
2. Use the model to confirm that critical immunization rate for random network is around 60% as predicted by literature. That is disease can only spread if the population immunization rate is above 60% otherwise it will limited to lo cal regions.
3. Study the effect of reducing the transmission rate.
4. Study the effects of long-range dispersal. In the usual percolation model for disease spreading the infection source is immobile, here we would extend the model to study the case when the infections source can disperse from time to time. In this limit we can use the model to interpret data from Ebola epidemics that evolve through long range dispersal. This long range dispersal are seen in figure1.3.

1.9 Project Organization

The rest of the work is organize as follows. In chapter 2 we give a details of the Percolation model. For example percolation model has three distinct states:(i)

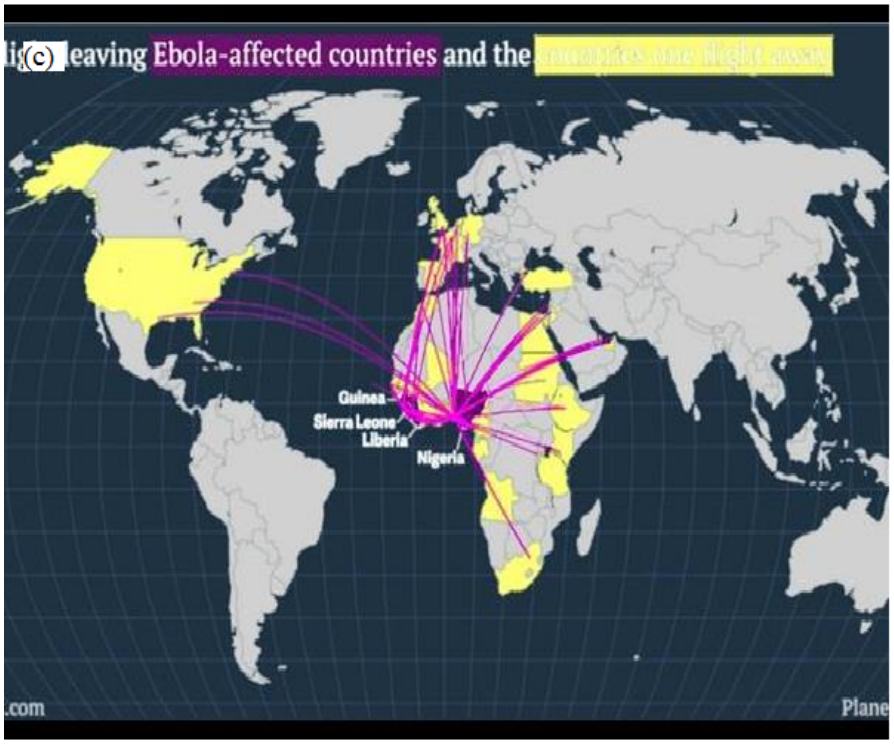


FIGURE 1.3: *The ebola affected countries mainly due to long range dispersal*

sub-critical state ($\rho < \rho_c$), (ii) critical state for $\rho = \rho_c$, (iii) super-critical state for $\rho > \rho_c$. Also background information on disease spreading using percolation theory using the computational algorithm for disease spread.

In chapter 3 we explain the different method and procedure that will be used to collect data. How are we going to group the results from the simulation to different group that will compared with results from the spread of EBOV.

Chapter 4 will contain graphical results from the simulations of the spread. These graphical results are differentiated according to transmission rate, immunization rate and long range dispersal. These results from the simulations will be compared with those from EBOV. The analysis and discussion of the results will then follow.

Chapter 5 a short conclusion will follow after the discussion of results in chapter 4. Recommendation on how we can contain the spread of disease and the prediction of future epidermic using results in chapter 4

Chapter 2

Theory/Methodology

Many phenomena within materials science, physics, and biology are associated with percolation theory. In particular, it has been shown that bond percolation is equivalent to the class of susceptible infectious susceptible(SIS) epidemic models on a network.

The basis of epidemiology models are base on population randomly mixing. Knowing the structure of the networks and how these deviate from the random-mixing allows the use of models to compute the epidermic dynamic of a disease at population scale from individual level infections.[4]

In such a model, all nodes start out susceptible to a new disease. If a node is infected, it will try to infect its neighbors for a fixed time(t),after which some will recover become susceptible again while some will be immune to the disease this is shown in figure **??**. How widely the disease spread on the network depends on the percolation probability ρ, and on the rate of immune individuals since all recovering individuals will be susceptible again. If ρ is greater than the critical percolation threshold ρ_c, then there is a finite probability that the disease will span the entire infinite network,thus becoming an epidemic.[5] In nature, percolation phenomena are often found near the critical probability ρ_c. A possible explanation of this is the concept of self-organized criticality (SOC),where complex systems drive themselves to critical states without the need for fine-tuning of the parameters [6].

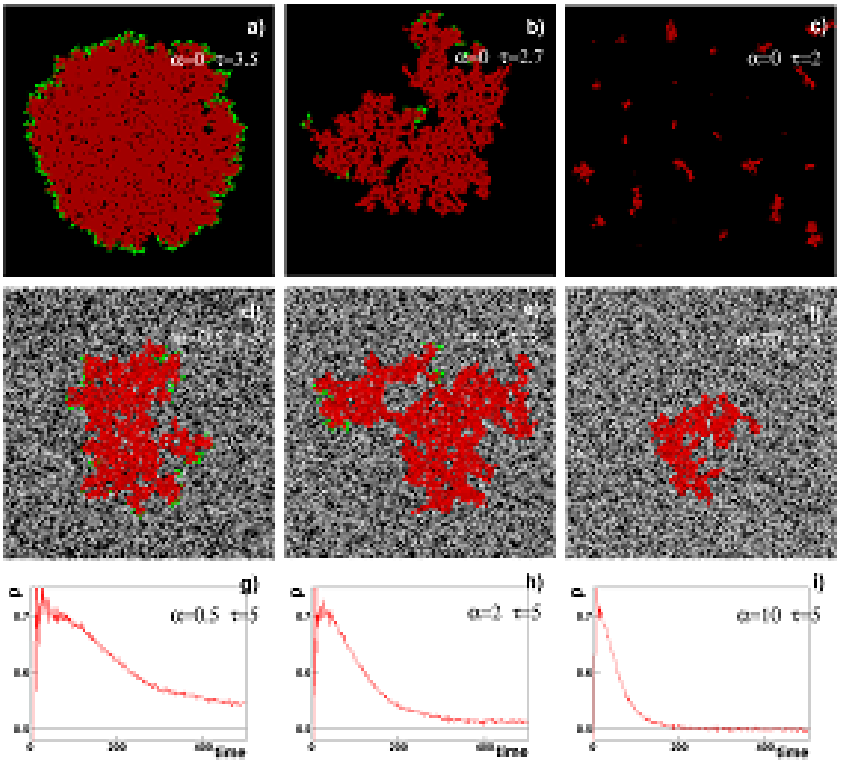


FIGURE 2.1: *Spread of diseases on a lattice of size $L = 256$ for different input parameters. Black nodes are healthy, and bright nodes carry many diseases. Accentuated clusters represent a particular disease If $\rho \approx 0.5$, diseases will spread in fractal shapes of critical percolation clusters. In the super-critical ($\rho > 0.5$)and sub-critical ($\rho < 0.57$) cases, diseases will grow to span the entire network or quickly die out,*

In these, the self-organization either arises as a result of very different time scales or through people running away from regions that are suspected to the source of a deadly disease e.g(ebola). For instance, a percolation system can self-organize to the critical threshold by dynamically adjusting the probability ρ, such that the percolation cluster keeps growing at a specific rate.

If the percolation probability is ≈ 0, the disease spread is rare and a few individuals will be infected. When t is very large the disease will have plenty of time to infect its neighbors this is seen in figure 2.1.

2.1 Background information

Mathematical models have become important tools in analyzing the spread and control of infectious diseases. The model formulation process clarifies assumptions, variables, and parameters; moreover, models provide conceptual results such as thresholds, basic reproduction numbers, contact numbers, and replacement numbers. Mathematical models and computer simulations are useful experimental tools for building and testing theories, assessing quantitative conjectures, answering specific questions, determining sensitivities to changes in parameter values, and estimating key parameters from data.

Understanding the transmission characteristics of infectious diseases in communities, regions, and countries can lead to better approaches to decreasing the transmission of these diseases. Mathematical models are used in comparing, planning, implementing, evaluating, and optimizing various detection, prevention, therapy, and control programs. Epidemiology modeling can contribute to the design and analysis of epidemiological surveys, suggest crucial data that should be collected, identify trends, make general forecasts, and estimate the uncertainty in forecasts. [7]

2.2 Algorithm from the computational implementation of the model

The following is the algorithm that was used to implement the python code for the spread of transmissible disease for specific population size and stipulated immunization and transmission rates.

```
Algorithmic description of a percolation model of disease propagation

N <= 130    {Population=N^2}

                            {Immunize a randomly chosen fraction of the population}

for i = 1 to N do
 for j = 1 to N do
  Choose a random number r
  if r < x then
   immunize person P[i][j]
  end if
 end for
end for
Select a person at random and infect them

                             {Now let the disease propagate}

While number of infected people increase do
 for i = 1 to N do
  for j = 1 to N do
   if P[i][j] is infected then
    infect all unimmunized neighbors
   end if
  end for
 end for
end while
Output coordinates of immunized people

Output coordinates of uninfected people

Output coordinates of infected people
```

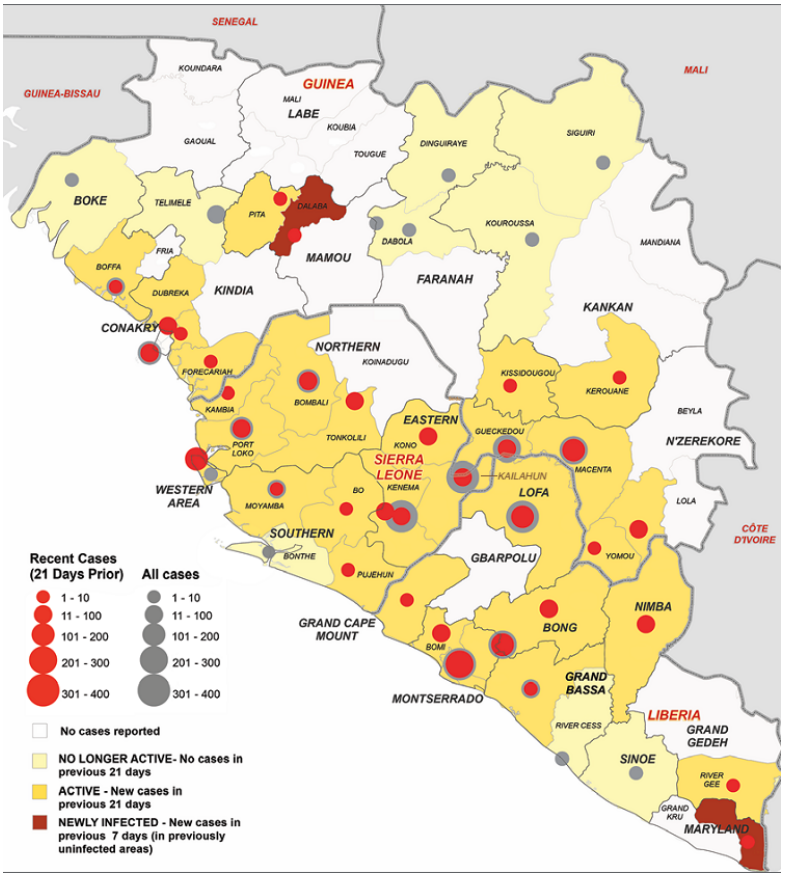


FIGURE 2.2: *Districts Affected by Ebola Virus Disease in Three Countries in Africa*

2.3 Ebola Virus Disease(EBOV) and Long range dispersal

The model was made to be flexible long range dispersal(jumps). These means that we can alter the model so that we can get a situation where by an infected person travel to a new environment and start an infection at a new environment. The EBOV will be used for testing the viability of the model. The EBOV is a good choice since the input information for the first time 9 month is available. Secondly the EBOV is currently the most feared disease in Africa with the highest transmission and the lowest immunization rate thus making it a perfect choice for the study.

The disease is spread is increasing causing a panic. This is resulted by the jumps. Intercontinental traveling can cause regional virus outbreak into a global health

event. These jumps has a great influence on the rate at which infection spread but without them disease spread relatively slower. These kind of spread create a wave-like growth where by the infected population increases radiates outwards from the a central core in a roughly circular fashion, these make it easier to identify the source of the disease and possible be able to stop other susceptible individual from getting infected.[2]

2.4 Assumptions were essential for the model to work

The following assumption were made to simplify the model and make the simulation to run for a short time without losing The objective of the model:

* Assumed a square lattice village.
* The minimum number of infected at the origin is 1 and the maximum is 5 and these people can be found anywhere in the village.
* The disease spread through contact with nearest neighbor and the interaction is purely random.
* The last day of running the simulation was made to be day 180.
* The window period was made to be one day.

2.5 Procedure and Method

The use of python software compiler to come with results which will be analyzed later. Specifically the description of data set acquired for use in the modeling, the procedure to interrupt and modify it.For interrupting the simulator the kill button is used this is essential if the simulated results are way out of line with the expected end results. The model is then rerun until we get the required results.

Python software must incorporates most of its packages eg(pylab,numpy,random) for the software to produce the desired results. The Implementation of the the algorithm describing the spread of a disease and how each of the steps are carried out is below.

2.5.1 How the python code works?

The python code has three stages,namely input,computation and output stage seen in figure 2.3. From figure 2.3 the input stage it is where the conditions for specific disease are satisfied. The input parameters constitutes of: immunization rate,transmission rate, long range dispersal, the population size, the number of nearest neighbor are included. It is important in knowing the number of average interactions a person may have. An infected person is traced is first seen at a random place inside the region.

The second stage is the main one, this is where all calculations and computations are done. In this stage a infected individual will start infecting all nearest susceptible individuals.This is repeated until every susceptible individual is infected or the simulator has reached its last day of simulation. The number of nearest neighbors for model is set to be 8. An infected person can only be able to start spreading the disease 24 hours later, such a short incubation period does not apply to most diseases.

All these data is recorded by the simulator and store the results, graphs are needed and the contour plots these leads us to the last stage. This is where all that the simulator has done is displayed. This part gives us information on the number of new infections recorded per day, accumulation infection against the number of days. Contour plots of accumulations rate vs time and survival rate. The spread of ebola virus data was obtained from world Health organization(**WHO**) will be plotted on the same graph as that from the simulator.

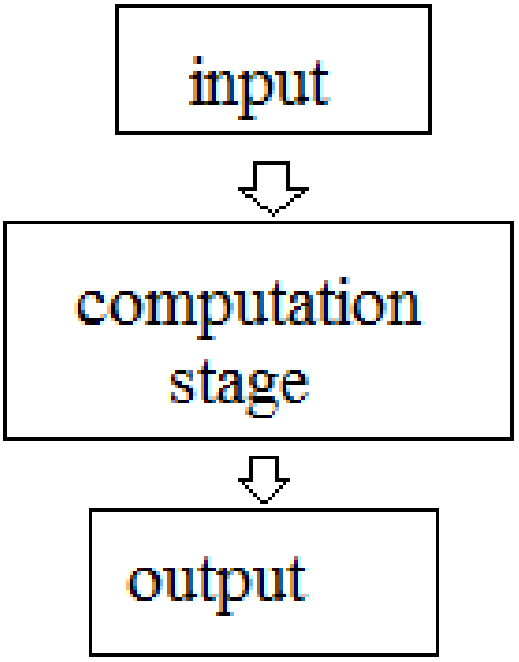
The contour plots will show three different colors.Firstly the sky blue represent the proportion of the population that is susceptible to the infection.Secondly the dark

blue represents the proportion of the population that is immune to the infection. Lastly the red color that represents the proportion that is infected and these is the type that is responsible for the spread of the infection.

For the population size of 16900 changing the immunization rate and transmission rate, a combination of the two that gives results which are closer to reality means we can predict the immunization rate with precision if we know the transmission rate for the same population size.

2.5.2 computation stage

The simulator will b e run for immunization rate of 30%, 60% and 68%. This provide a wide range to analyze at the behavior of the simulator, while simultaneously analyzing the spread of a disease that satisfy the state condition. The transmission rate will b e from 8% to 40% and comparing will be done with that of ebola virus disease which the transmission rate was predicted to be around 27%.The results from ebola spread in west Africa for the first 160 day and fit into the model. We can predict the results after 180 days for a start. The model can be used to predict for later dates.



.

FIGURE 2.3: *the block diagram of the three stages of the simulator*

Chapter 3

Results And Analysis

3.1 Disease spreading at different immunization rate

The results from the simulations for varying immunization rate at constant transmission rate and short range dispersal (an infected person is not allowed to travel freely in the region). An infected the infection starts anywhere inside the region. Figure 3.1 show results from the simulations with the red color representing the infected population, the light blue representing susceptible and the dark blue represents the immunized proportion of the population. It is observe from the simulations that the immunization rate play a significant role in the spread of the diseases. Increasing the immunization rate results in a decrease at the rate at which diseases spread this result agrees with literature.

The contour plots from figure 3.1 show that the disease continue to spread throughout the immunized population if the immunization rate is below 68% this means that the percolation threshold(p_c) for a disease that spread like that in figure 3.1 is 68%. These result help when trying to contain a disease from causing an epidemic.

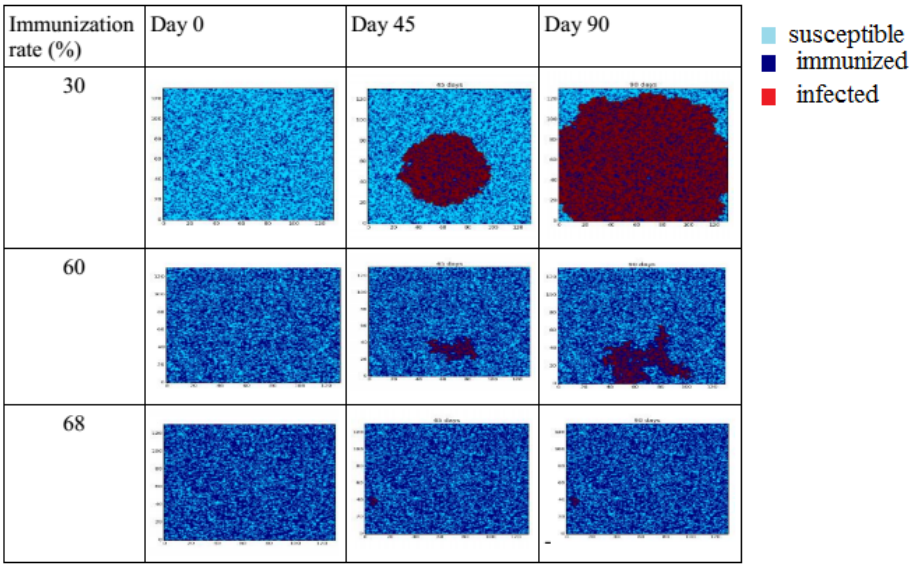


FIGURE 3.1: *the contour plots for for immunization at 30%,60% and 68% for a 130 by 130 square lattice for the first 90 days.transmission rate kept at 40%*

The graph in figure 3.2 shows that the number of new cases per day increases almost exponentially until we reach a maximum at day 80 where the number of new cases is about 200. After day 80 the number of new cases starts decreasing at a fast rate than they were increasing. At day 120 the disease stops spreading because almost every susceptible individual is already infected. Figure 3.3 shows the drop in the number of new cases per day if the immunization rate is doubled.The results from this graphs testify the significance of the immunization rate. The results from figure 3.3 shows that the maximum numb er of infected individuals p er day is 28 which is almost frac14 of the maximum obtained in figure 3.2. The results from figure 3.2 and 3.3 show that the maximum is reached at almost the same day. Increasing the immunization rate reduces the speed at which a disease spread but the disease continue to spread beyond day 120. this is implied from the fact that increasing immunization rate results in an increase in the barrier made by the immunized population that the disease has to overcome.

Increasing the immunization rate to 68% results in figure 3.4 for the same transmission rate of 40%. The immunization of 68% is just above p_c. From figure 3.4

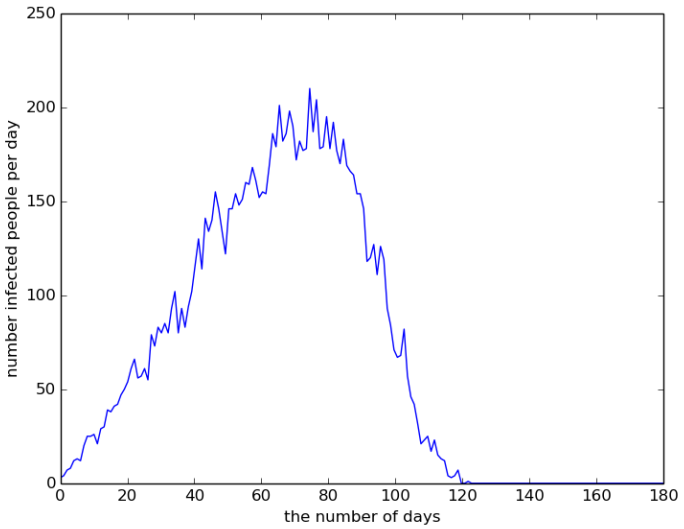


FIGURE 3.2: *the number of new cases per day for the first 180 days at transmission rate of 40% and immunization rate of 30%*

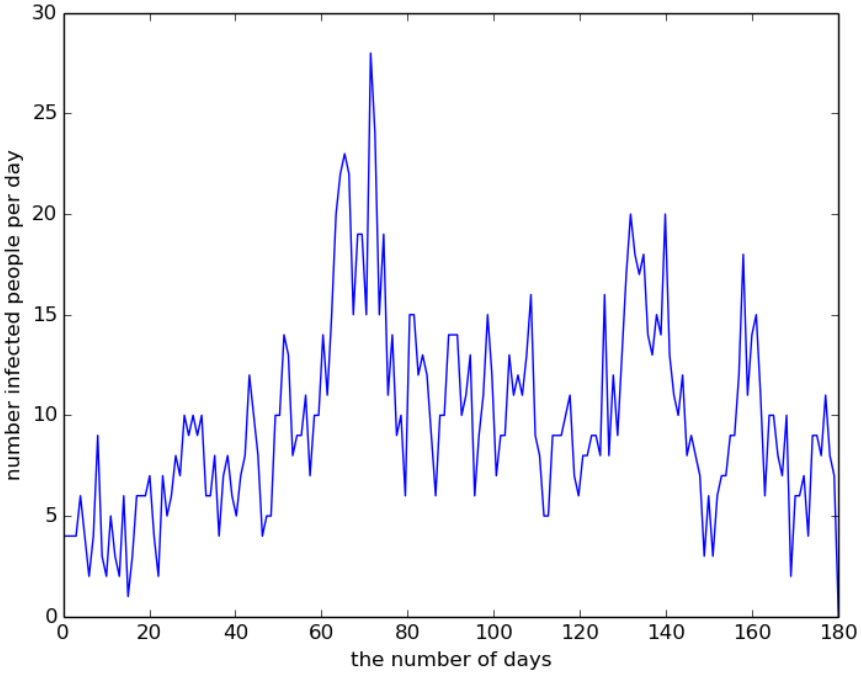


FIGURE 3.3: *the number of new cases per day for the first 180 days at transmission rate of 40% and immunization rate of 60%*

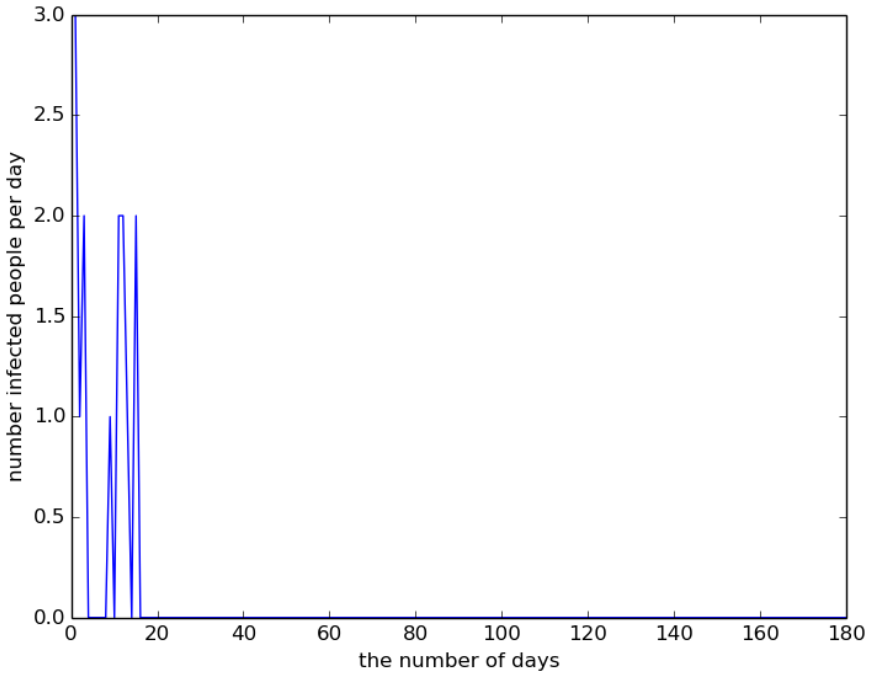


FIGURE 3.4: *the number of new cases per day for the first 180 days at transmission rate of 40% and immunization rate of 68%*

we observe that the maximum number of new cases per day is 3 which is much lower compared to figure 1.3 and 3.3. It is observed at percolation threshold the number of new cases is very low compare to the immunization rate below the p_c.The disease stops spreading after 18 days. It is observed that the number of new cases per day in some days in zero this is due to the immunization rate which is above percolation threshold.

3.2 Effects of the transmission rate and long range dispersal

The model show different types of results obtained from the model. The results of main importance now is the effects of transmission rate on disease propagation and long range dispersal. From the model, after the simulator is run at two different immunization rate; 30% and 60% it gives results in figure 3.5. The contour plot in figure 3.5 shows that an increase in transmission rate results in an increase in

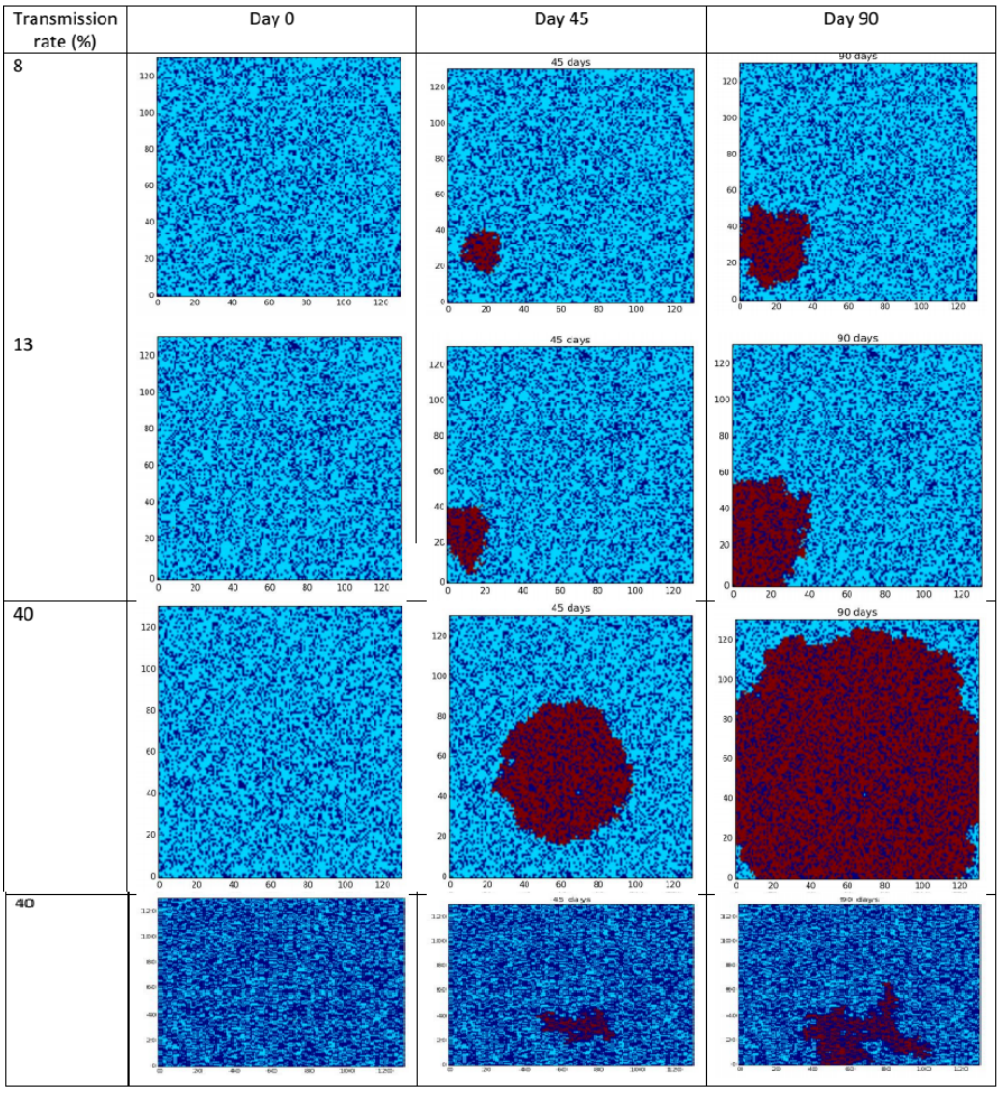


FIGURE 3.5: *the light blue represent susceptible , dark blue immunized proportion and red represents infected proportion. The change in transmission rate of a population at constant immunization rate of 30% and the last part show immunization rate at 60%*

the number of infected individuals in the region. This is observed in figure 3.5 that when the transmission rate is at 40% only a small fraction of the susceptible population remain uninfected.This kind of spread can also be explain bye the water wave like spread outwards. This is as a results of panic from the population as they try to run away from the source they move in all direction that causing the spread like that in figure 3.5. With these results it can be used to predict the behavior of the population toward as certain epidemic disease.

The results from figure 3.5 show no pattern meaning the population's response to

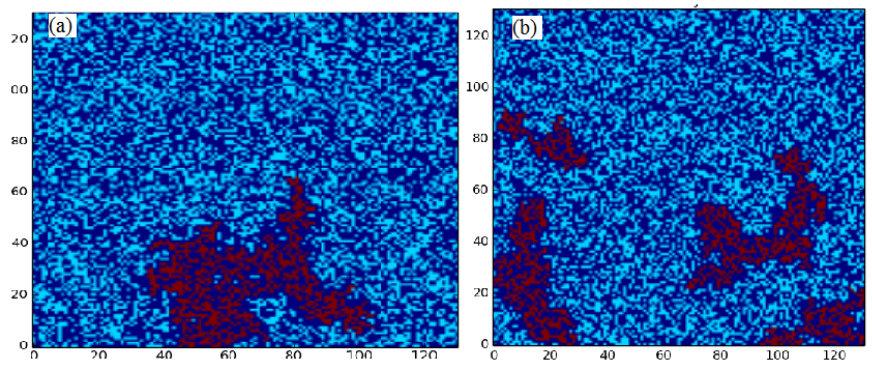


FIGURE 3.6: *(a) show disease spreading at immunization rate of 60%, transmission rate of 40% and zero long range dispersal. (b)show disease spreading at immunization rate of 60%, transmission rate of 40% and average of 5 long range dispersal.*

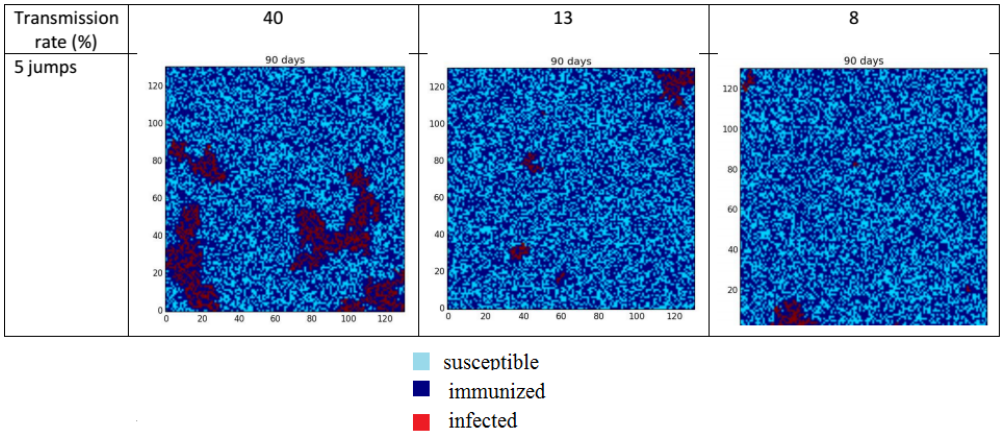


FIGURE 3.7: *the change in transmission rate of a population at constant immunization rate of 60% with an average of 5 long range dispersal*

the epidemic scary and it make it easier for healthy personnel to educate and raise awareness to the public about the epidemic. Comparing figure 3.5 and last part of figure 3.5, it is observed that the spread of epidermic is directly proportional to the transmission rate. Also the rate at which individuals get infected increase irrespective of the immunization rate. The effects of long range dispersal is shown in figure 3.7 and 3.6 for day 90 of the simulation. These results they show the dependence of the transmission rate and long range dispersal on the proportion of the population that is infected. figure 3.7 shows that the main contributor to the speed at which diseases is not the ability of infected people being able to travel but on the transmission rate. After it has been understood how immunization rate, transmission rate and long range dispersal affects the spread of diseases, this the

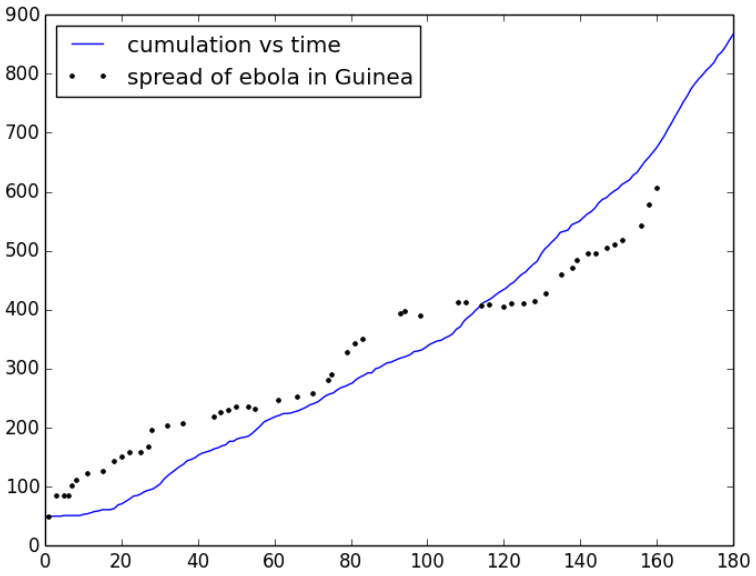


FIGURE 3.8: *total number of infected people daily when transmission rate at 13% immunization rate are 60% (between 20-March to 20-Oct 2014 in Guinea)*

brings us to the last part of the results which compares the spread of ebola virus disease in two of the worst affected countries in Africa Guinea and Sierra Leon shown in figure 3.8 and 3.9 respectively. These results they show the validity of the model. Although at some points the error is large but in both cases the value of the model approach the true value of the as time increases. This mean at later time model give more accurate results. The transmission rate in Sierra Leon is lower than that in Guinea. This could be a result of the awareness that people in Sierra Leon had when the epidemic started in their country. For both countries the average number of long range dispersal was five.

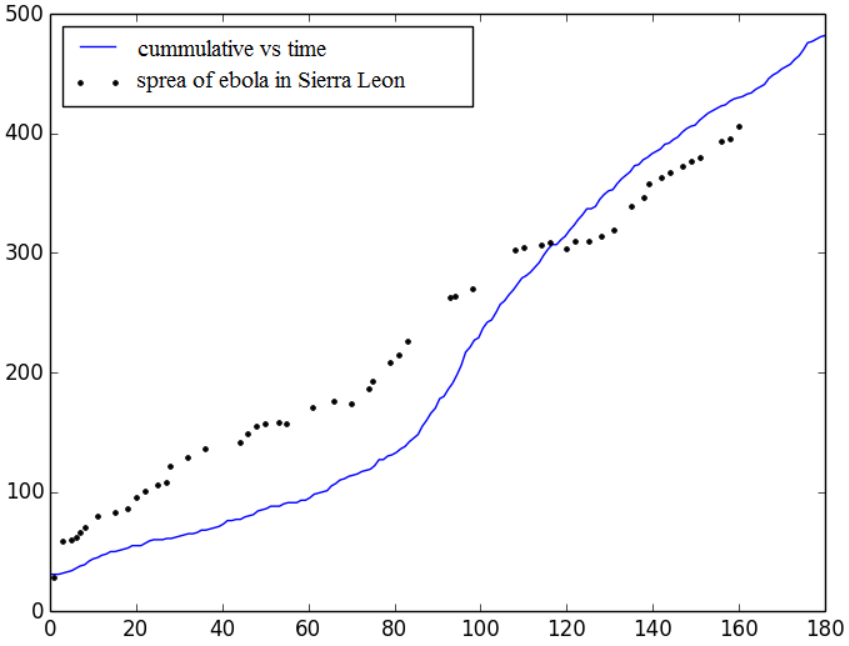


FIGURE 3.9: *total number of infected people daily when transmission rate at 12% immunization rate are 61% (between 27-May to 20-Dec 2014 in Sierra Leon)*

Chapter 4

Conclusion

The research was a success. The python code that simulate the spread of diseases was developed and it was verified using the spread of ebola virus disease in Guinea and Sierra Leon. The critical immunization probability for the model was found to be 68% which has an absolute error of 0.13 to the 60% predicted by literature. The rate of spread of disease also depends on how the population react to the threat possessed by the epidemic, if the population panic the rate of spread increases.

It was found that decreasing transmission rate increases the time it takes for the disease to spread throughout the whole susceptible individual clusters. Long range dispersal increase the speed at which disease spread. The main contributor to spread of diseases is the immunization rate and transmission rate. The results show that the rate at which any disease spread will be different for different regions.

4.1 Recommendations

To contain any disease first the percolation threshold of the disease must be known. If a vaccine of a certain disease has been synthesize the number of vaccinated population should just be above the critical percolation probability. For the continuation of the project the recovery rate should be considered birth and dead rate should be considered.

Bibliography

[1] H. Herbetv L. Leviven, Acceleration of evolutionary spread by long-rang dispersal (2014).

[2] D. Bernoulli and S. Blower, An attempt at a new analysis of time mortality caused by smallpox and of the advantages of inoculation to prevent it,p275-288 (2004).

[3] Cristia. Althus, Estimating the reproduction number of Ebola virus in west Africa (2014)

[4] Matt J. Keeling and Ken T.D. Eames , network epidermic, p 295-307, (2005)

[5] J. Juul and K. Sneppen , Lo cally self-organized quasi-critical percolation in a multiple disease model(2011)

[6] P. Bak and K. Wiesenfeld, Phys. Rev. Lett., 59, 381 (1987).

[7] H.W. HETHCOTE,Three basic epidemiological models, in Applied Mathematical Ecology, p 119-144,(1989)

[8] www. nibib.nih.gov, Computational Modeling(16-April-2015)

Python code used for diseases modeling

```
# Python Code for description of a percolation model of disease propagation
from numpy import *
import itertools
from collections import Counter
from random import *
from pylab import *
from mpl_toolkits.mplot3d import *
from math import *

N = 130 # population is N*N
x = 0.6# proportion of immunised
t =0.13 #it is the probability that the person transmit\
 the disease after getting infected
n_last = -1
n = 0
duration =180
P = np.zeros([N+1,N+1],float)
size = range(N+1)
minp = range(N+1)
maxp = range(N+1)
ni = zeros(duration,int)
g = zeros(duration,int)
pi = zeros(duration,int)



for i in range(N+1):
     minp[i] = i-1
     maxp[i] = i+1
minp[0] = 0
maxp[N] = N
```

```
b = linspace(0,duration,duration)#the array for the number of days


for i in size:
     for j in size:
         r = rand()  # immunize a fraction 'x' of population
         if r< x:
             P[i,j] = 0 # immunize
         else:
             P[i,j] = 1  # leave uninfected

P[N*rand(),N*rand()] = 3
# infect the first individual manually
imshow(P,origin="lower")
totals = Counter(i for i in list(itertools.chain\.from_iterable(P)))
print "The number of new infected people at the origin is:",totals[3],\
      "The number of immunized people after origin is:",totals[0],\
      "The number of uninfected people at the origin is:",totals[1]
show()
#  keep going until as many people  are infected
for day in range(duration):
     for i in size:
          for j in size:
               if (P[i,j] ==3):
                    # if infected, then infect the neighbors
                    n=n+1 # count how many infected
                    if (P[minp[i],j]==1):
                         if t >= rand():
                              P[minp[i],j]=2
                         else:
                              P[minp[i],j]==1
                    if (P[maxp[i],j]==1):
```

```
            if t >= rand():
                P[maxp[i],j]=2
            else:
                P[maxp[i],j]==1
        if (P[i,minp[j]]==1):
            if t>= rand():
                P[i,minp[j]]=2
            else:
                P[i,minp[j]]==1
        if (P[i,maxp[j]]==1):
            if t>=rand():
                P[i,maxp[j]]=2
            else:
                P[i,maxp[j]]==1
        if (P[minp[i],minp[j]]==1):
            if t>=rand():
                P[minp[i],minp[j]]=2
            else:
                P[minp[i],minp[j]]==1
        if (P[maxp[i],maxp[j]]==1):
            if t >= rand():
                P[maxp[i],maxp[j]]=2
            else:
                P[maxp[i],maxp[j]]==1
        if (P[maxp[i],minp[j]]==1):
            if t >= rand():
                P[maxp[i],minp[j]]=2
            else:
                P[maxp[i],minp[j]]==1
        if (P[minp[i],maxp[j]]==1):
            if t>=rand():
```

```
                        P[minp[i],maxp[j]]=2
                    else:
                        P[minp[i],maxp[j]]==1
for i in size:
     for j in size:
          if (P[i,j] ==2):
               P[i,j]= 3
#if day == 5:
 #    P[N*rand(),N*rand()] = 3
 #after 20 days infected person jump to another village
if day == 10:
     P[N*rand(),N*rand()] = 3



if (day == duration/8):
     imshow(P,origin="lower")
     title("22 days")
     totals = Counter(i for i in list(itertools.chain.from_iterable(P)))
     print "The number of new infected people after"\
     ,duration/8," days is:",totals[3],\
        "The number of uninfected people is:",totals[1]
     show()
if day == 40:
     P[N*rand(),N*rand()] = 3 #after 40 days infected\
     person jump to another village


if (day == duration/4):
     imshow(P,origin="lower")
     title('45 days')
```

```
        totals = Counter(i for i in list(itertools.chain.from_iterable(P)))
        print "The number of new infected people after\
         ",duration4," days is:",totals[3],\
              "The number of uninfected people is:",\
              totals[1]
if day == 60:
        P[N*rand(),N*rand()] = 3 #after 60 days infected\
         person jump to another village

if day == 80:
        P[N*rand(),N*rand()] = 3#jump after day 80


        show()
if (day == duration/2):
        imshow(P,origin="lower")
        title('90 days')
        totals = Counter(i for i in list(itertools.chain.from_iterable(P)))
        print "The number of new infected people \
        after",duration/2," days is:",totals[3],\
          "The number of uninfected people is:",totals[1]
        show()
if day == 100:
        P[N*rand(),N*rand()] = 3 #after 100 days infected\
         person jump to another village



totals = Counter(i for i in list(itertools.chain.from_iterable(P)))
ni[day] = totals[3]-totals[2] # number of new\
 infection daily
pi[day] = totals[3]
```

```
imshow(P,origin="lower")
title('last day')
show()
totals = Counter(i for i in list(itertools.chain\
.from_iterable(P)))
print "The number of new infected people in last \
day is:",totals[3],
     "The number of uninfected people is:",totals[1]
for q in range(1,duration):
     g[q-1]=ni[q]-ni[q-1]
print "The immunization rate ",x,", and the transmition\
 rate is ",t
plot(b,g)#number of new infection per day
xlabel('the number of days')
ylabel('number infected people per day')
xlim(0,duration)
#ylim(0,N)
show()
plot(b,pi+48,label="cumulation vs time")
data=np.loadtxt("values.txt",float)
x=data[:,0]
z=data[:,1]
plot(x,z,"k.",label= "spread of ebola in Guinea")
# the scatter graph of ebola spread in Guinea
legend(loc='best')
show()
```