

An Agent-Based Model of Recurring Epidemics

Brendan Greenley

Computer Systems Lab 2008-2009

Thomas Jefferson High School for Science and Technology

February 10, 2009

Abstract

Even with today's modern scientific and medical breakthroughs, there exists the threat of a widespread epidemic. Could the Avian Flu wreck havoc on the human population like the Spanish Influenza nearly a century ago? Widespread epidemics have historically acted as a population control, as seen by the Black Plague. What characteristics are needed by a virus to successfully control a population but not risk completely wiping out its hosts? How do reoccurring epidemics control a population over long periods of time? If the behavior of individual agents changes, the outcome of an epidemic will also change, and the results can help paint a picture of what might happen in a real epidemic outbreak.

Keywords: epidemic, agent-based, modeling

1 Introduction

1.1 Purpose

If an epidemic were to occur, there would be two factors that would affect its duration and severity: The behavior and characteristics of the individuals within a population (interactions, movement, mating, immunity, and self-quarantine) and characteristics unique to the disease, such as its severity (in terms of mortality) and duration. By assigning different agents varying values representing characteristics such as susceptibility, and allowing for on-the-fly changing of outside factors such as the disease duration and base chance of disease transmission, one can find out how the population as a whole would react and change to an epidemic and any possible quarantine or response by government and health officials. Programming in NetLogo allows one to take advantage of a built in GUI that permits manipulation of values through sliders and input fields and has instantaneous visual output, as individual agents and their movements can be seen as well as a continuously updating plot of populations vs. time. (Figure 1)

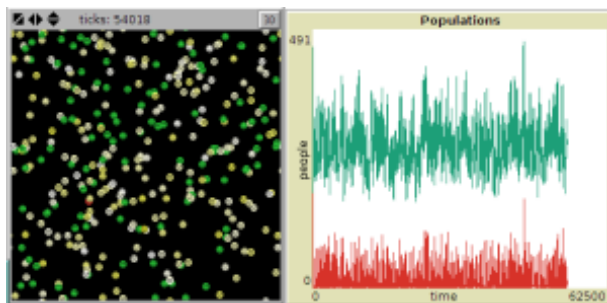


Figure 1. The epidemic model in the middle of a run

1.2 Background

When the epidemic model was first begun during first quarter, I chose system dynam-

ics modeling rather than agent-based modeling as the basis for my epidemic model. However, it quickly became clear that a System Dynamics approach would create difficulty in modeling a populations behavior for someone with a limited mathematics background like me. In December of 2008, I abandoned my System Dynamics model and entered the world of agent-based modeling. In agent-based modeling, rather than basing populations and trends on a predetermined coding block or equation, a world is populated with agents with a simple block of commands. However, since individual agents can have varying characteristics and interact with the environment in different ways, when the simplicity of one agent is multiplied by hundreds or thousands, one gets a very complex but useful representation of an entire population not possible through other model types. The epidemic model has a few, obvious rules. Firstly, there are agents who have the disease, those that do not have the disease but can be infected by those that do have it, and those that are immune. Agents die from the disease after a given interval of time, partially decided by the user and partially computer assigned (an average is inputted, the program then distributes values based on this average.) Susceptibility of an agent is also distributed by the model on a curve, as some people are more prone to disease than others.

2 Development

So far, the agents have primitive movement and mating procedures: no matter the age or condition of a person, they move one space in a random direction, and children are hatched on the same location as a parent as soon as the parents mate. The chance for an agent to mate occurs when one agent who has not

mated recently is next to other agents. The agent then checks to see if any of the agents around it are willing to mate. If so, a random number is generated, and based on it the agents either mate successfully and have child or fail and move on. If children are indeed created, their characteristics such as susceptibility are numerically represented as the average of their two parents own characteristics. Agent immunity is distributed over a distribution curve. As agents are assigned susceptibility values, the model determines whether or not they are immune. The value which a persons susceptibility must be to be granted immunity is set by the user on the immunity factor slider. Agents are visually represented by both varying color and facial expressions based on their susceptibility, immunity, or exposure to the disease. A green, smiling agent is immune, and regardless of the chance of transmission or its surroundings, it will not become sick (unless the virus mutates and the agent loses its immunity.) A white to yellow, neutral face represents a person susceptible to the disease, the more yellow an agent is the more susceptible they are. A red, frowning face represents an infected agent who can spread the disease to others. When run, the speed of the model can be set by setting the tick rate at the top of NetLogos GUI. Each tick interval, the position of individual agents positioned on a black square window representing the world being modeled are updated, and the plot of the total amount of people alive and infected is updated. As time passes, the graph scale shrinks as to fit in all the data. While this is great for tracking long-term progress, the local dips and spikes within individual shorter time intervals are lost. However, by using a feature built into NetLogo known as BehaviorSpace, one can comprehensively analyze data after a run has

completed rather than having to make conclusions based on solely the live observation of a model.

3 Results

In order to keep a population from crashing to zero or exponentially growing and slowing the model to a halt, it was found that any epidemic must occur in a severe and frequent enough manner to limit the population. Because it is hard to see general trends with these frequent oscillations, it was useful to import data into Excel using BehaviorSpace and then add a moving average trend line. (Figure 2.) It was also found that agent-based epidemic models can be very volatile. A program which runs seemingly fine for thousands of ticks may suddenly crash, a result of a combination of bad luck in transmission of the disease and poor agent decisions in regards to movement and mating.

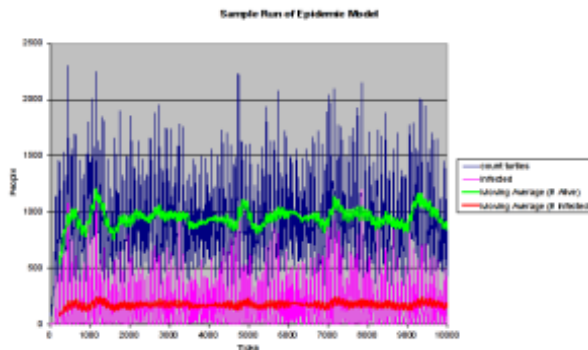


Figure 2. Post-run data analysis in Microsoft Excel

4 Conclusions

Agent-based models can help make generalizations about how a population reacts, but primitive behaviors, such as random movement and less than ideal mating procedures do limit the accuracy of the model. However, these procedures will be improved to get a

more realistic epidemic model, and additional features such as quarantines will also be included.

References

- [1] B. John Oommen, Dragos Calitoiu, *Modeling and simulating a disease outbreak by learning a contagion parameter-based model*, Proceedings of the 2008 Spring Simulation Multiconference, SpringSim 2008, Ottawa, Canada, April 14-17, 2008
- [2] Macal, C. M. and North, M. J., *Tutorial on agent-based modeling and simulation part 2: how to model with agents.*, In Proceedings of the 38th Conference on Winter Simulation (Monterey, California, December 03 - 06, 2006). L. F. Perrone, B. G. Lawson, J. Liu, and F. P. Wieland, Eds. Winter Simulation Conference. Winter Simulation Conference, 73-83.
- [3] Bobashev, G. V., Goedecke, D. M., Yu, F., and Epstein, J. M., *A hybrid epidemic model: combining the advantages of agent-based and equation-based approaches.*, In Proceedings of the 39th Conference on Winter Simulation: 40 Years! the Best Is Yet To Come (Washington D.C., December 09 - 12, 2007). Winter Simulation Conference. IEEE Press, Piscataway, NJ, 1532-1537.