Simulation of the Spread of a Virus Using Agent Based Modeling

Matt Wade
Computer Systems Lab 2007-2008

Abstract

My goal is to make an agent based modeling simulation that shows the spread of a cold through a school. The simulation will start with an amount of infected students and healthy students received as inputs and will show how much the virus spreads or possibly recedes over time. The program will answer the question as to how quickly and fully different types of sicknesses will be able to spread through the population of a school once introduced. This will show how likely it is for a disease to be spread by a set amount of sick people coming to school with the sickness.

Development

My program has three main classes. An Agent class which defines what values an Agent will store and how to construct it, a class that creates the GUI, and the Model class which is where all of the calculations occur. In the Model class there is an ArrayList containing all of the Agents in the simulation. With these Agents the class has to define a step() function which moves the simulation forward. This function has to update the locations of all of the agents, check to see if any of them get infected, and check to see if any recover from being sick. First the model goes through the list of Agents and moves them all to the next location in their schedule. In order to check for any new infections it goes through the entire list of Agents finding each sick Agent. Whenever a sick Agent is found the program finds any healthy Agents in the same location and checks a randomly generated number against the sick Agent's infectiousness value. If the random number is lower then the healthy Agent is switched to sick and the method continues on through the rest of the list. One additional check of a random number is done against the infection chance in order to compensate for the possibility of getting sick through outside influences when traveling from one location to the next. To check if any Agents recover from sickness the list Is parsed, checking each Agent’s recovery time value and if this value equals zero the Agent becomes healthy.

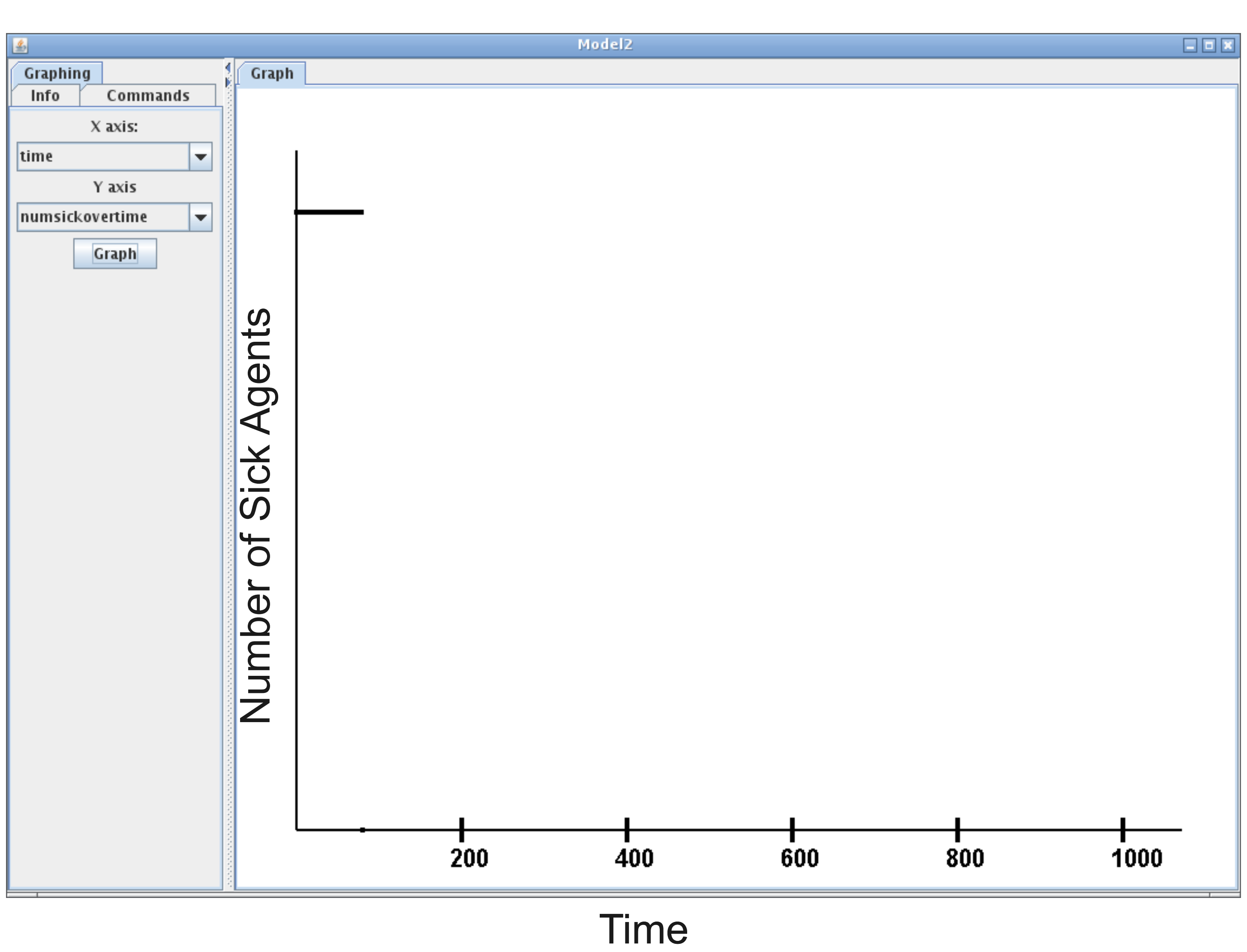

Figure 1. Results gained from actual research value. The value became zero after the initial sick Agent recovered his health.

The GUI class contains another important function, the ability to graph data gathered through the simulation. In the Model class I added new ArrayLists to retain information over time for all of the main variables (number of sick agents, number of healthy agents, infections per step, etc.). Whenever the graphing method is called the Model class gets information from two drop down menus as to which ArrayLists are going to be the x and y axis variables and sends them to the graph class. The graph class then takes these variables and goes through the ArrayList graphing a scatterplot of the data. To obtain a conclusion from the program, I ran the simulation through multiple trials each with the same inputs. I made there be 1999 healthy agents and 1 sick agent, spread across 80 classes. This meant there was an average of 25 agents in each class which is normal for a school.

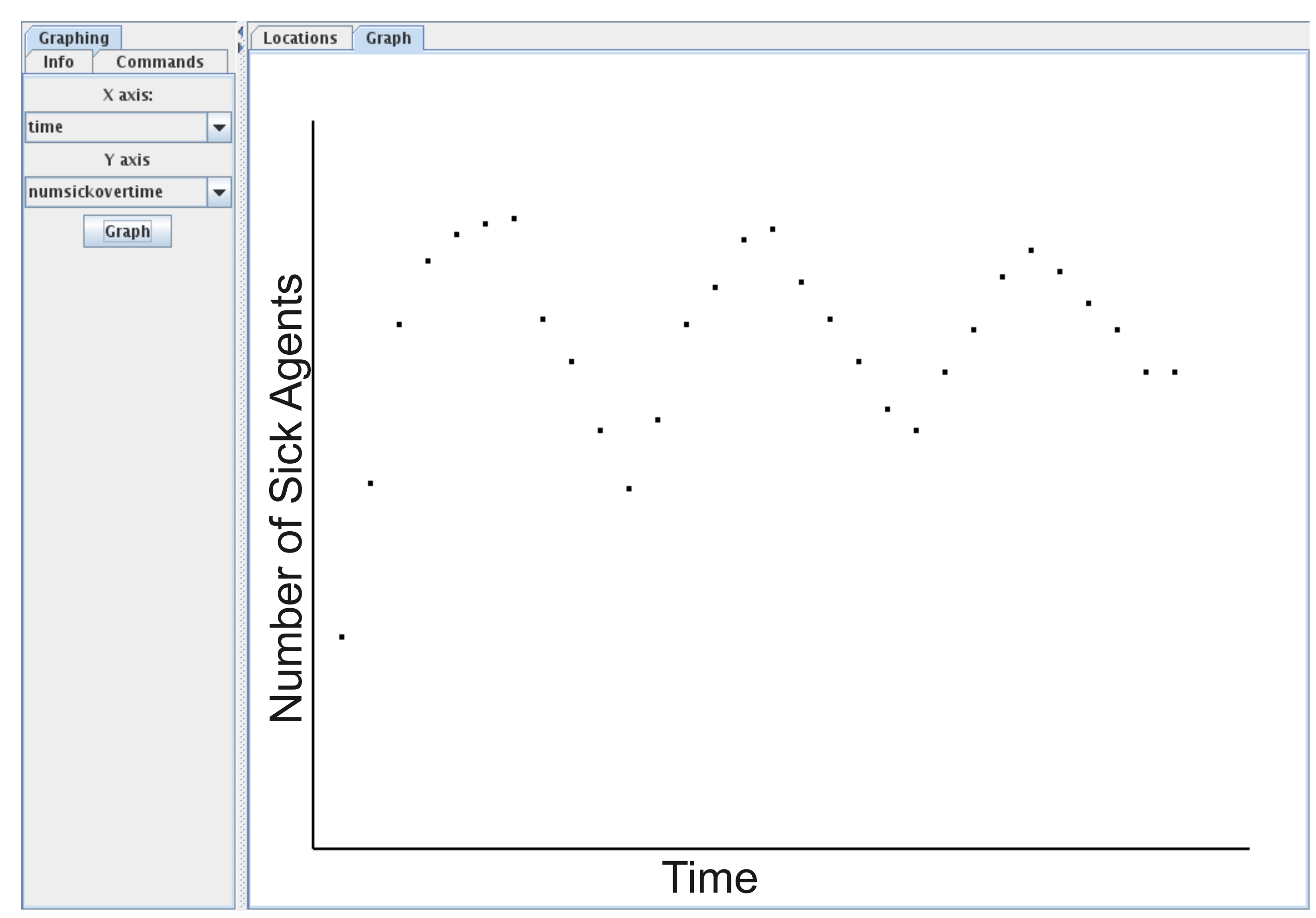

Figure 2. Results using research value multiplied by 2.

Results

Using the infection percentage value found from research, the simulation ran with zero other agents becoming infected each time. This seems odd, because if none of the many people that the sick agent came in contact with became sick in a any trials, the illness would not ever spread to anyone and would die out. This situation has not happened in real life because the flu is still a concern. I think this is due to having such a small number of agents. The data I used was for the percentage of the population of the world that becomes sick with the flu each year. This value includes all the people who do not come into contact with an infected person, which is likely a much larger amount of people than the percentage of agents in my simulation that are in contact with the sick agent. Since the agents are in a more concentrated area than the people in the research data, this results in the infection chance being lower than it actually should be.

To counteract the above effect I tried using other values for the infection chance by multiplying the value by two, five, and ten, but each time this resulted in too many agents being infected. Multiplying by five and ten resulted in unrealistically large numbers of infections with the simulation balancing out at more than 50 percent of agents being infected. Multiplying by two was slightly better usually resulting in around 100 sick agents. Going strictly by the simulation with the research data, I can conclude that if you are sick with the flu there is no reason that you should avoid going to school or work unless you are physically unable to, since it is unlikely that you will spread the flu around.