User Manual

Homework 4

Aaron Zachariah

Contents

Application Overview

This application simulates the spread of an infection among a network of nodes, represented as a graph. The user must specify a file to load data from in the form of an adjacency list. All nodes in the graph can be in one of four states – susceptible, infected, recovered, or dead. Susceptible nodes are nodes that have yet to be infected, but might be in the future. Infected nodes are currently infected, and they will remain that way until the infection runs its course. Once the infection finishes, a node will either recover, or die. Recovered nodes cannot be re-infected, and neither can dead nodes (obviously).

The application offers a variety of ways to configure the nature of the simulation, including initialization methods and parameters and infection spread parameters. User will be able to load a graph from an input file, select an infection method to set the initial state of the graph, and simulate forward in time. Reference the sections on toolbar and menu bar features to see how to configure various parameters. Reference the section of Simulation Parameters to see the function of each parameter. Reference the section on Infection Methods to see the options for setting the initial state of the graph.

Please see `README.txt` on information regarding starting the application.

Simulation Parameters

The simulation has a variety of parameters which dictate its behavior. They are all configurable via buttons and user input. Below is information regarding all parameters and their default values. Note that for different types of graphs, it would make sense to change parameters form the default values.

`# of Threads` – The number of concurrent threads to run the simulation on. Default = 1

`n` – The number of nodes that will be infected when using the random infection method. Default value is 3

`s` – The degree threshold for the Degree infection method. All nodes with degree greater than s will be marked as infected. Default value is 5

`k` – The max number of nodes to infect via BFS traversal. Default value is 3

`d` – Also known as the Recovery Rate of the infection. When the infection runs its course, nodes will survive with probability d and die with probability 1-d. Default = 0.5

`t` – The infection duration in ticks. Infected nodes will stay infected for t ticks. Default value is 5.

`lambda` – The force of infection, which is the rate at which the infection spreads to other nodes. Default value is 1.25

Infection Methods

To run the simulation, a user must load a graph file, as well as set the initial state of the graph. There are three different ways to set the initial state of the graph.

`Random Infection` – Nodes will be randomly marked as infected. This method will mark n nodes as infected. Parameter n is configurable.

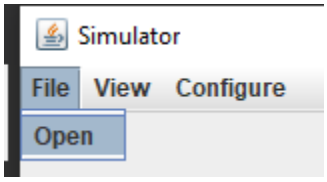
`Degree Based Infection` – Nodes will be marked as infected based on their degree. Nodes with degree greater than s will be infected. Parameter s is configurable.

`BFS Based infection` – A random starting node will be selected, and a BFS traversal will be run from that node. The traversal will cover at most k nodes, and mark all covered nodes as infected. Parameter k is configurable.

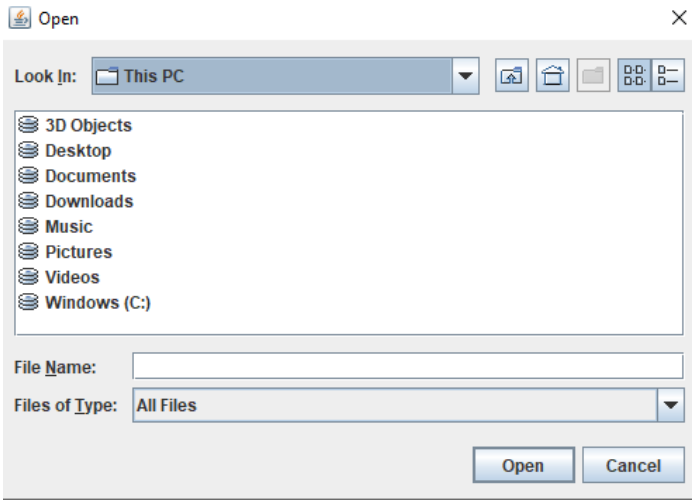
User Input

In order to start the simulation, the user must first load a network of nodes into the application.

To open a file and load it into the simulation, simply go to File -> Open, and select your input file.



Upon clicking open, a file selection box will appear. Simply select your input file and click open to load it into the simulation.



Input File Format

The input file must have a certain format in order to be properly loaded by the application. The File must be in the form of an adjacency list. An adjacency list represents a graph as an array of linked lists. The index of the array represents a vertex and each element in its linked list represents the other vertices that form an edge with the vertex.

Following this convention, the whole file can be though of as an array of lines, where each line of data corresponds to a linked list. Each line should start off with the index (or line number) which will be the vertex, and the subsequent data will be other reachable vertices from the first vertex listed.

A simple file example is shown below:

```
0 2 5 7
1 5 6 7 8
2 0 6 8
3 4 6 9
4 3 5 8 9
5 0 1 4 7
6 1 2 3 7 9
7 0 1 5 6 8 9
8 1 2 4 7 9
9 3 4 6 7 8
```

In the above example, each line's index is listed as the first element in the line. All subsequent elements are reachable vertices from the index vertex. For example, the vertex "0", has an edge to vertices 2, 5, and 7.

Input graphs for this application are meant to be undirected graphs, which means the edges have no direction. This means that if some node N has node M in its adjacency list, then node M must also have node N in its adjacency list.

Observe the nodes "0" and "2" from the above example. Node "0" is in the list for 2, and node "2" is in the list for 0.

Please see the references section for more information regarding adjacency lists.

All elements in the input files can be delimited by a space, a “|”, a “,” or a “;”. Delimiting the elements with any other characters will cause the data to be improperly interpreted.

Menu Bar Features

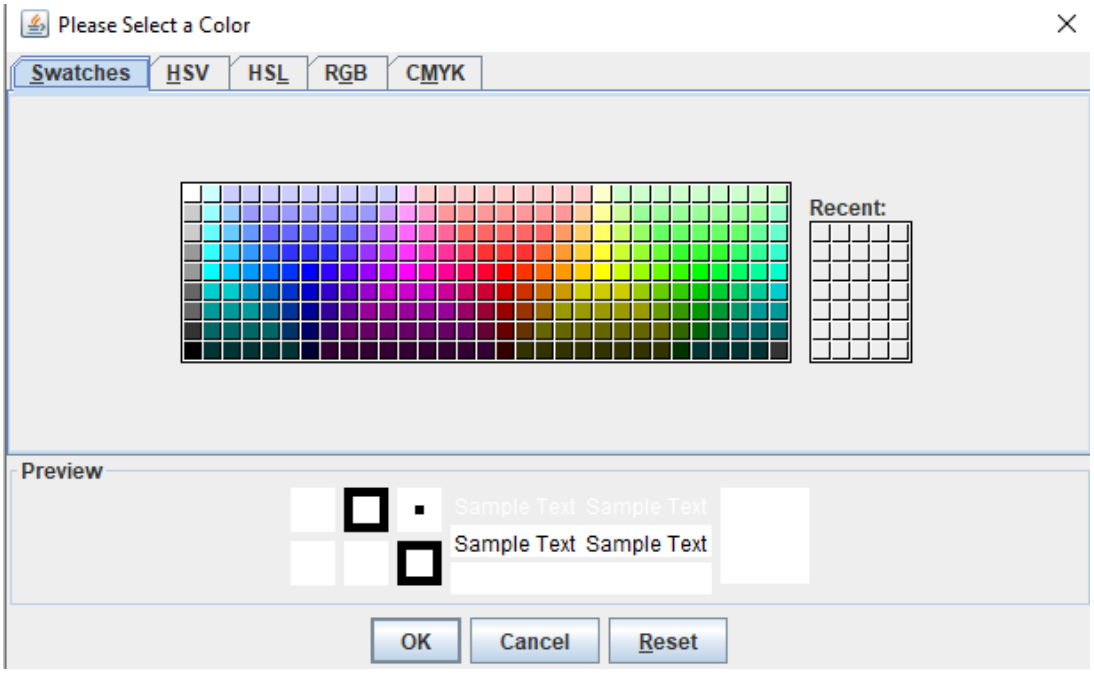
The Application contains a menu bar near the top of the window, which provides a variety of options to configure the application and simulation behavior.

File Menu

Please refer to the previous section for more information on the file menu and its uses.

View Menu

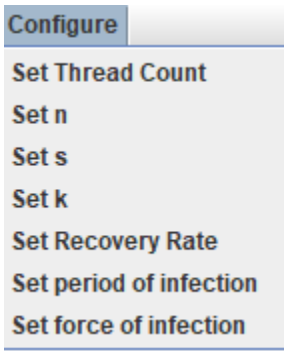
The view menu allows the user to customize the view of the application by changing the background color. To open the color selection dialog, go to View -> Set Background Color. The following color selection dialog will appear.



Simply select the color of your choice and click “OK”.

Configure Menu

The configure menu offers multiple ways to configure various parameters for the simulation.



To see more information about the parameters, refer to the Simulation Parameters section.

Set Thread Count

Opens a prompt which allows the user to specify the number of threads to use when running the simulation.

Set n

Allows the user to specify the number of nodes to infect when selecting random infection.

Set s

Allows the user to specify the degree threshold that determines if nodes get infected when using the Degree based infection. All nodes with degree greater than s will be infected

Set k

Allows the user to specify the max number of nodes to be infected when using BFS based infection. The # of infected nodes will be at most k nodes.

Set Recovery Rate

Allows the user to specify the recovery rate for infected nodes. If recovery rate is p, nodes will survive with probability p and die with probability 1-p.

Set Period of Infection

Allows the user to specify the number of ticks that an infected node will stay infected.

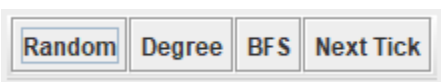
Set Force of infection

Allows the user to specify the force of infection, or the rate at which the infection spreads to other nodes

Toolbar Features

The toolbar offers a way for the user to use common commands easily. The toolbar contains buttons used for setting the initial state of the graph, and for simulating to the next tick.

To see more information on the infection methods used to set the initial state of the graph, see the section on Infection Methods.



Random

Clicking this button causes the button to use the Random Method for infecting nodes. Can only be used before the simulation has started

Degree

Clicking this button causes the button to use the Degree based Method for infecting nodes. Can only be used before the simulation has started

BFS

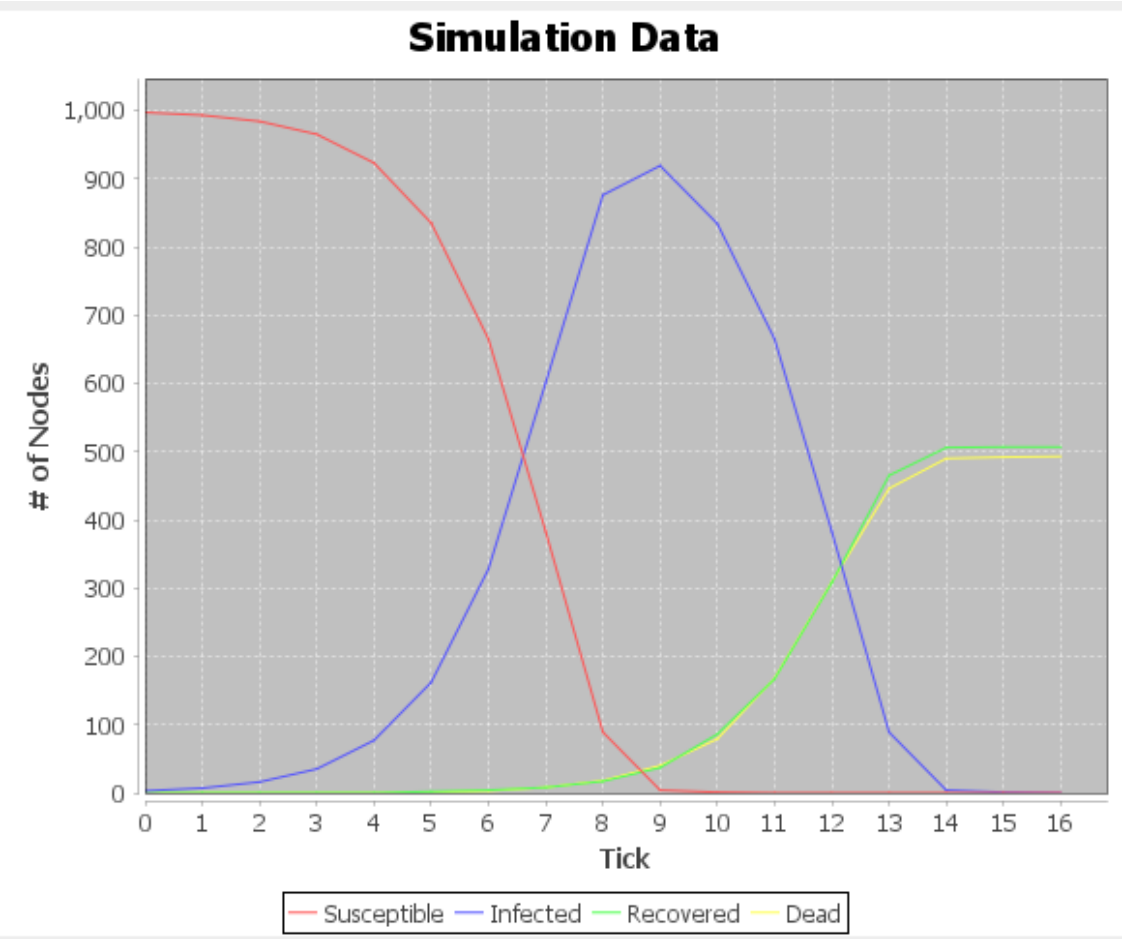
Clicking this button causes the button to use the BFS Method for infecting nodes. Can only be used before the simulation has started

Next Tick

Simulates to the next unit of time, and updates the state of the graph.

Line Chart

The application also features a line chart that is centrally located. The line chart is a visual representation of the trends in the state of the graph over time.



The chart displays the ticks on the x axis and the number of nodes on the y axis. The Chart features four series, one for each possible state of a node in the graph.

The chart also features a legend, so that each series is distinguishable.

Errors

There are possible error messages that may occur upon faulty user input. If you receive an error message, refer to the below error messages to find a possible fix.

`NumberFormatException` – This is a error which will occur when the application expects the user to input a number value, but the input cannot be parsed as a number. To fix this, simply make sure to enter a number.

`Thread Count Errors` – This will occur if the thread count specified is some invalid value. To fix this, make sure the thread count is an integer greater than 0.

`Parameter n Errors` – Will occur if the value specified for n is invalid. The n parameter must be greater than 0 but also less than the size of the graph

`Parameter s Errors` – Will occur if the value specified for s is invalid. The s parameter must be greater than 0.

`Parameter k Errors` – Will occur if the value specified for k is invalid. The k parameter must be greater than 0 but also less than the size of the graph

`Invalid Rate Errors` – Will occur if the specified recovery rate value is invalid. The recovery rate must be a decimal value greater than 0 and less than 1.

`Invalid Period Errors` – Will occur if the specified infection length is invalid. The length must be greater than 0.

`Invalid FOI Errors (Force of Infection)` – Will occur if the specified force of infection is invalid. The force of infection must be a decimal value greater than 0.