

---

# Network Software Modelling Assignment 2

---

*Students:*

Louis CARNEC

Vijay KATTA

Adedayo ADELOWOKAN

*Student #:*

15204934

15202724

15204151

April 23, 2017

Master of Science (Business Analytics)  
UCD Michael Smurfit Graduate Business School  
2016-2017

# 1 Real world phenomenon: Spread of disease through airports network

The world is more closely connected than ever before by modern transportation networks. Air, sea and land transportation continues to expand in reach, speed of travel and volume of passengers and goods carried. Pathogens and their vectors can now move further, faster and in greater numbers than ever before [3]. Epidemics can occur more easily and the emergence of novel pathogens exacerbates the situation [1]. Understanding the dispersal behaviour and identifying the outbreak of an epidemic in its preliminary stages enables critical response planning. Network topology properties that surround the debut location can explain much of the early stage variation in the spread of diseases [2].

In this simulation we will model the spread of disease through the US airport network. It is inspired by the work of Yager and Taylor [4] which model the spread of pathogens through airports around the world using a directed graph. We will extend their work by testing how graphs (subgraphs of the US networks graph with specific properties and Erdos-Renyi graphs with different edge existence probabilities) and initial infection nodes (with different centrality measures) affect the spread of disease through the network.

## 2 US Airport Network Model

Our real-world network graph is undirected where nodes represent airports and edges are routes. We used open source data from <http://openflights.org> along with population of US cities from [https://unstats.un.org/unsd/demographic/products/dyb/City\\_Page.htm](https://unstats.un.org/unsd/demographic/products/dyb/City_Page.htm). Redundant edges and unconnected nodes were removed from the graph. Edge weights were calculated by getting the average population of the two cities connected by edges and normalising for all edges. In creating the graph, only edges connecting airports with the US were added. The graph created deviates from the real world epidemic scenario. Firstly, exclusively flights within the US are considered. In reality, disease spread through air transport is a world wide phenomenon. In a world wide scenario, we could expect the spread of disease across US nodes to be even quicker as more edges would connect US airport hubs. Secondly, the graph is undirected, this is due to the fact that there is no data for passenger numbers on specific routes. We assume that passenger on routes is proportional to the population of the city which is served by an airport. It is also assumed that routes are bidirectional, to simplify.

## 3 Simulation Rules

The states of each node/airport are either; susceptible to infection (green), infected (yellow) or dead/closed down (red). All nodes are initially disease free and thus susceptible to infection except from a node which is infected at the beginning, step 0. Once a node is infected, it can infect any of its neighbours with probability related to the edge

weight connecting it to its neighbours. In the case where the airport can close down or dies (these terms will be used interchangeably), if the airport has closed down it cannot spread anymore disease, there are no outgoing flights.

Nodes are coloured based on the status - green is susceptible, yellow is infected and red is removed/dead. The four  $K_{2,2}$  graphs in Figure 1 each represent the state of nodes in a graph at one time step. (a) demonstrates the initial state, where node A is infected (yellow) and all other nodes are susceptible (green). In (b), nodes B and D are infected with probabilities  $w_{AB}$  and  $w_{DA}$  respectively, as they are the neighbours of node A. (c) shows that all nodes are infected and in (d), the next time step, nodes B and D are red, they are dead/closed indefinitely.

When a node is dead, it cannot further spread the disease. Therefore, if for example the initial node dies before spreading the disease to a neighbour, for example if it has a low probability of spreading the disease, the spread of the disease will be stopped prematurely.

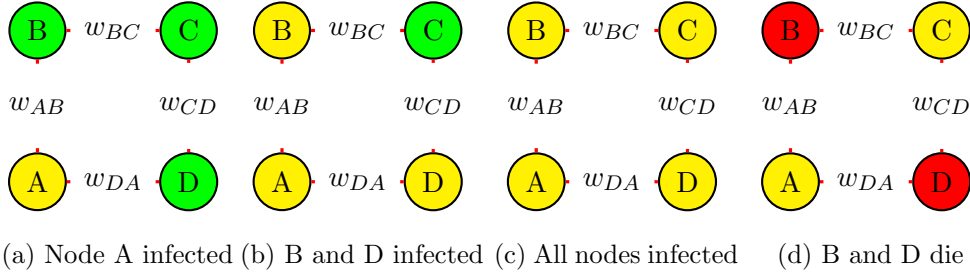


Figure 1: Spread of Infection Simulation Rules

## 4 Simulation and Graph Properties

### Simulation Properties

We will investigate two main simulation properties; with and without airports being able to close or nodes dying. The hypothesis is that the spread of the disease throughout the network will be slowed down by airport closing as, at each time-step, they cannot spread the disease to neighbours once closed down.

Each simulation depends on the probability of being infected at a given timestep ( $p_{infection}$ ) and the time after infection at which the node dies. In both cases  $p_{infection} = 0.4$  for simplicity, that is the probability an infected node will infect a neighbour is 0.4, if the edge weight is 1.0. The second simulation, where airports can close, also depends on the time after infection when the airport closes (set in the initial simulation to 4 time steps). We would expect both parameters to have an effect on the spread of disease. The greater the probability of infection, the faster the spread. The longer it takes for the airport to close, the faster the rate of infection. And vice versa.

## Graph Properties

We hypothesise that the simulations will depend on:

- Diameter: the largest distance between any two nodes in the graph. The greater the diameter, the longer it should take for the disease to infect the whole graph.
- Centrality of initial node. The more central an airport/node the faster the disease should spread to other nodes in the network. We will investigate degree centrality and betweenness centrality.

## 5 Experiments

The experiments were conducted using 25 time steps, as a result some simulations did not run until the whole graph was infected. Due to the large number of results produced by the experiments, a subset will be presented here. When comparing initial node centrality between the real world graph and generated graphs, the Erdos-Renyi generated graphs were created with 143 nodes to mimic the real-world graph of the same order.

Both simulations were tested on the airport network and its subgraphs and a range of Erdos-Renyi random graphs for different sizes and probabilities of edge existence. The full US airport graph has a diameter of 4, that is any two airports within the graph have at most 2 airports between them. Subgraphs of the US airport graph are investigated; a subgraph with edge weights  $> 0.75$  (diameter is infinite, graph not connected), subgraph with edge weights  $< 0.5$  (diameter = 3), subgraph of 20 largest degree centrality nodes (diameter = 2), and the minimum spanning tree of the airport graph (diameter = 8). Erdos-Renyi graphs' diameter depends on the choice of  $n$  (the order of the graph) and  $pn$  (probability of an edge existing).

Tests were also conducted using specific nodes for initial infection based on centrality. The nodes with highest and lowest betweenness and degree centrality, as well as the centre node were infected at time 0 to test their effect on both the real world and Erdos-Renyi graphs.

## 6 Results

### Diameter

#### Real World Graph and Subgraphs

The spread of the disease through the graphs differs greatly as we expected depending on the graph we are testing on for both scenarios 1 and 2 (see Figures 4,5,12,13). In scenario 1, the disease spreads quickly, by the 5th time step, 90% of the graph is infected but the rate of spread greatly levels out thereafter. Comparing large and low edge weight graphs, the spread for the large edge weight graph is slow whereas it is fast in the low edge weight case. The disease does not spread to the whole graph in the 25 steps of the run but levels out. This has to do with the fact that the whole graph is not connected

(diameter = inf). With the 20 largest degree centrality node subgraph, the spread is much quicker than in all other graphs, the disease has spread to the graph in 3 time steps. The minimum spanning tree is slow to spread, it takes some time before the disease reaches a high centrality node, at which point the disease spreading rate increases. In the second simulation, in both subgraphs (largest and lowest edge weights), the disease does not spread to the entirety of the graph. In the case of largest edge weights the disease spreads at a slower rather to that of the lowest edge weights subgraph. For our largest degree centrality node sub graph, the disease spreads at very high rate and by the 7<sup>th</sup> time step the whole graph is closed/dead. Although interesting, these results do not tell much about the effect of graph diameter on disease spread. It is more likely that the properties of the nodes within the subgraphs (edge weights, degree centrality of nodes, minimum spanning tree nodes) explain the rate of disease spread for the different subgraphs tested.

## Centrality

### Erdos-Renyi

When comparing how the degree centrality of the source node affects the spread of the disease. We compared the spread of the disease throughout the graph using initials nodes with the highest and lowest degree centrality. The behaviour between both nodes are quite similar with a slightly slower rate of infection in the case of the node with lowest degree centrality. Using an initial node with max degree centrality, the graph is infected by the 8<sup>th</sup> time step. The more edges a node has the quicker we would expect it to spread the disease to the rest of the graph. Table 2 shows that for the generated graphs, using min degree centrality always results in slower graph infection. See Figure 6 in Appendix. In the second simulation, in which airports can close, in both cases where the initial node is of lowest or highest degree centrality we observed that nodes begin to die off when the number of infected nodes is at its peak. Interestingly, for graphs of size  $n=150$  and  $n=250$ , the time it takes to spread the disease is closer for both centrality measures than in simulation 1. See Figure 14.

Examining the effect of using initial nodes with minimum and maximum betweenness centrality, we find that in the first simulation when the source node has high betweenness centrality the disease spreads at a higher rate than using minimum betweenness centrality. In the second simulation, the behaviour is similar to that of the first scenario and as noted above the death rate is also similar. See Figure 7 in Appendix.

### Real World Graph

In our real world graph, a minimum degree centrality node, the rate of spread is similar to the Erdos-Renyi graphs. The rate of the spread is a lot faster using a max degree centrality source node in the real-world graph compared to the spread observed in the Erdos-Renyi graphs. In the second simulation we observe the same situation to airports being closed off, it starts at the point where the infected nodes reaches its peak. The

behaviour is also similar to that of the observed Erdos-Renyi graph. See Figure 8,9,14.

Betweenness and degree centrality nodes, whether minimum or maximum, have similar behaviours in both generated and real-world graphs.

## Generated Graph Order

Interestingly, we find that graph order is inversely proportional to the time until all nodes are infected/dead, as well as graph diameter (more nodes results in more edges thus lower diameter). Comparing both simulations, for smaller graph order ( $n = 150, 250$ ) simulation 2 takes less than half the time to infect the graph of simulation 1, while it takes more than half the time for larger order graphs ( $n = 500, 1000$ ).

GraphType	Order	Diameter	Center Of Graph	Min Deg Centrality	Max Deg Centrality	Min Btwnes Centrality	Max Btwnes Centrality
Real-World	143	4	9	inf	8	12	inf
Erdos Renyi	150	4	6	11	5	7	5
Erdos Renyi	250	3	5	6	4	6	4
Erdos Renyi	500	3	5	5	4	5	3
Erdos Renyi	1000	3	3	3	2	3	3

Table 1: Simulation 1 : Time Step - When entire network is infected

GraphType	Order	Diameter	Center Of Graph	Min Deg Centrality	Max Deg Centrality	Min Btwnes Centrality	Max Btwnes Centrality
Real-World	143	4	inf	inf	inf	inf	inf
Erdos Renyi	150	4	10	13	11	12	11
Erdos Renyi	250	3	10	11	10	10	9
Erdos Renyi	500	3	8	8	8	9	8
Erdos Renyi	1000	3	7	8	7	8	7

Table 2: Simulation 2 : Time Step - When all nodes are closed/removed

## 7 Conclusions

The tests we initially ran using subgraphs of the airport graph with specific node properties, confused the cause of disease spread time. The properties of the nodes within the subgraphs dictated disease spread instead of diameter size. The generated graphs using different probabilities of edge existence did show that higher edge existence probability (lower diameter and higher graph density) resulted in faster disease spread. Thus the more connected the airport network, the faster the spread of disease. We would recommend airports to conduct preventive measures such as screening passengers at hub airports first and close those if required to avert an epidemic situation, essentially reducing the diameter of the airport network and slowing disease spread.

The starting node used affects the spread of disease throughout the graph. Examining spring layout plots for each time step of the simulation and node state percentage plots we uncovered some interesting cases. If the initial node used has no edges, unsurprisingly the disease does not spread to the graph. If the node has edges with low weight, the spread of disease is slow until it reaches a node with high centrality. Thus, if a disease is initiated in a small regional airport, the spread will turn into an epidemic once a major airport is infected. In the second scenario, we found that for some runs

using airports with low edge weights / betweenness centrality / degree centrality, (low edge weights results in lower probability of spreading to neighbour node) the infected airport will shut before the disease spreads to neighbouring nodes. Closing an infected airport can stop the spread of disease. The sooner the airport closes, the less likely the disease is to spread.

## Appendix

### Simulation 1 - Nodes Cannot Die

#### Testing Diameter: Graphs, Subgraphs and Edge Probability

##### Erdos-Renyi Graphs

index	pn	order	size	density	cluster coefficient	diameter	nodes deg $\geq n$	largest component
0	1	143	117	0.011523687580025609	0.0	inf	0	97
1	3	143	325	0.032010243277848911	0.03655788655788656	7	0	143
2	5	143	527	0.051905840638235001	0.05108083447244284	5	14	143
3	7	143	757	0.074559243573328077	0.0733257571774318	4	49	143
4	9	143	885	0.087166354771988572	0.08693534635371428	4	87	143
5	11	143	1111	0.10942578548212351	0.1088499366852766	3	128	143

Table 3: Simulation 1: Graph properties for different probabilities  $pn$  of edge existence between nodes in Erdos-Renyi graphs

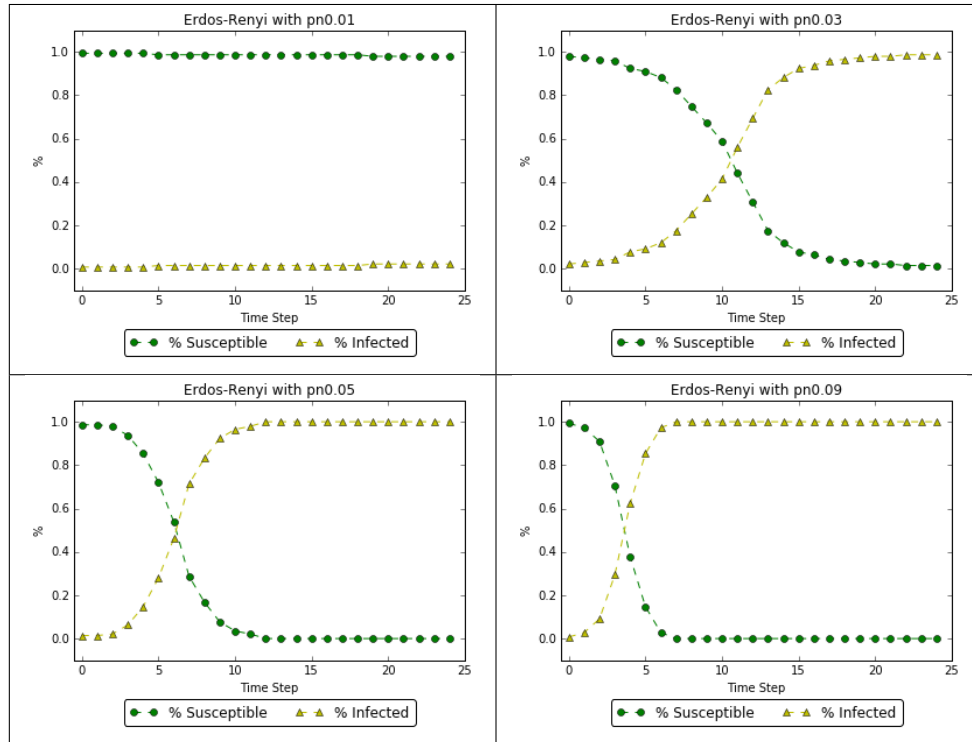


Figure 2: Simulation 1 Erdos Renyi : Plotting Node States for Graphs for  $pn = 0.01, 0.03, 0.05, 0.09$



## Real Airport Graph and Subgraphs

Name	order	size	density	cluster coefficient	diameter	nodes deg $\geq$ n	largest component
Real World Graph	143	1452	0.14301191765980498	0.6410089238208931	4	73	143
Largest Edge Weights	13	10	0.12820512820512819	0.0	inf	1	7
Lowest Edge Weights	27	102	0.29059829059829062	0.6458430700260764	3	13	27
20 Largest Deg. Cent.	20	177	0.93157894736842106	0.9423609156581293	2	20	20
Min Span. Tree	143	142	0.013986013986013986	0.0	8	3	143

Table 4: Real World Airport graph and subgraphs graph properties

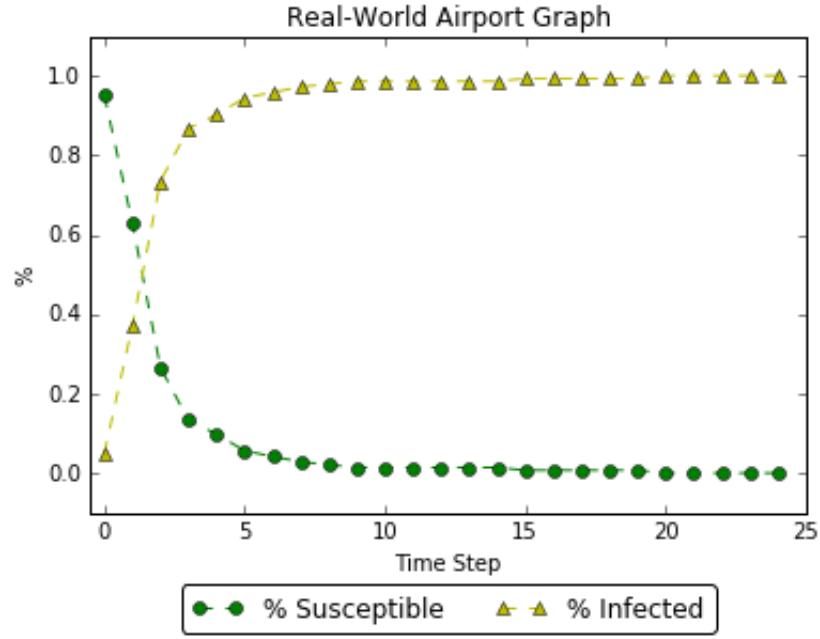


Figure 3: Simulation 1 Real-world : Airports graph properties

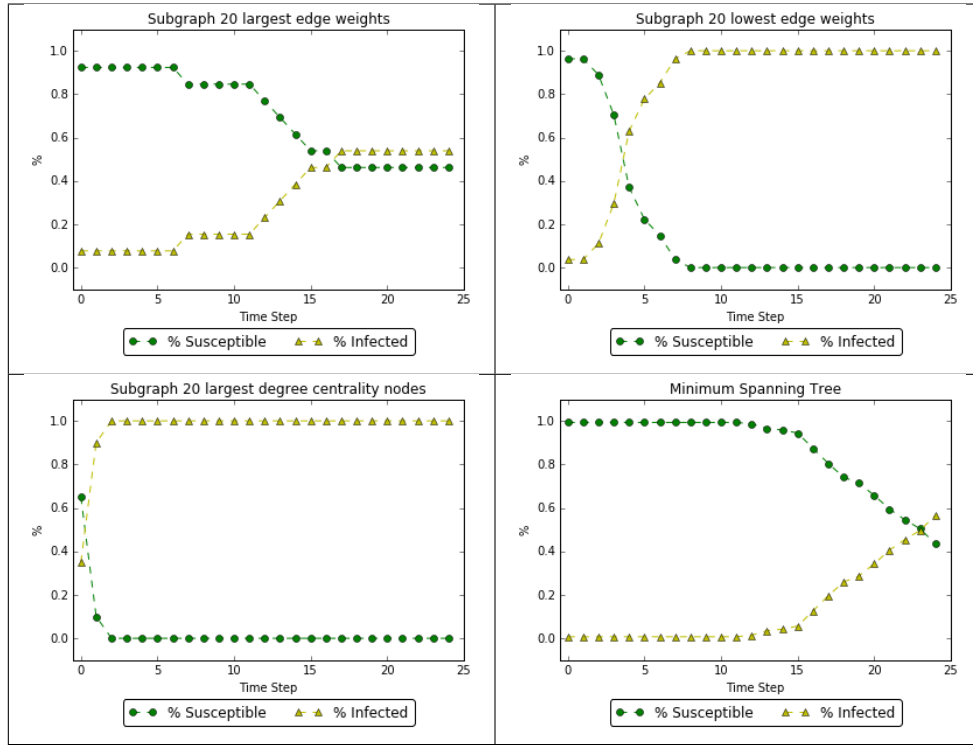


Figure 4: Simulation 1 Real-world : Subgraph node state plots

## Source Node Centrality

Erdos-Renyi Graph with  $p = 0.09$

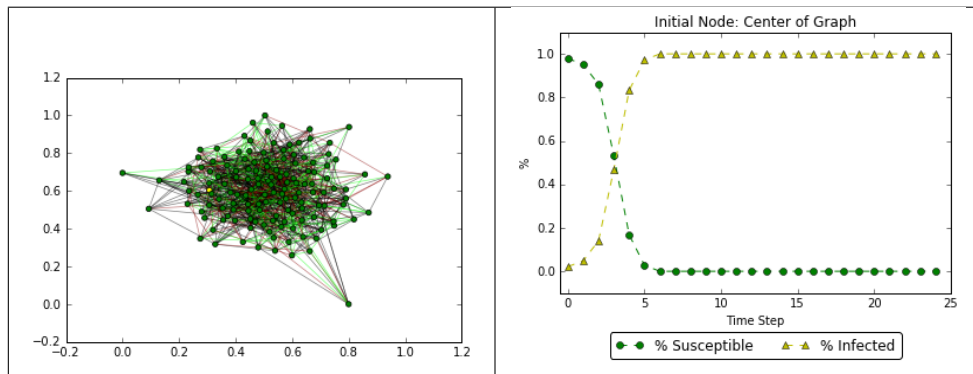


Figure 5: Simulation 1 Erdos-Renyi : Initial Node Center Of Graph

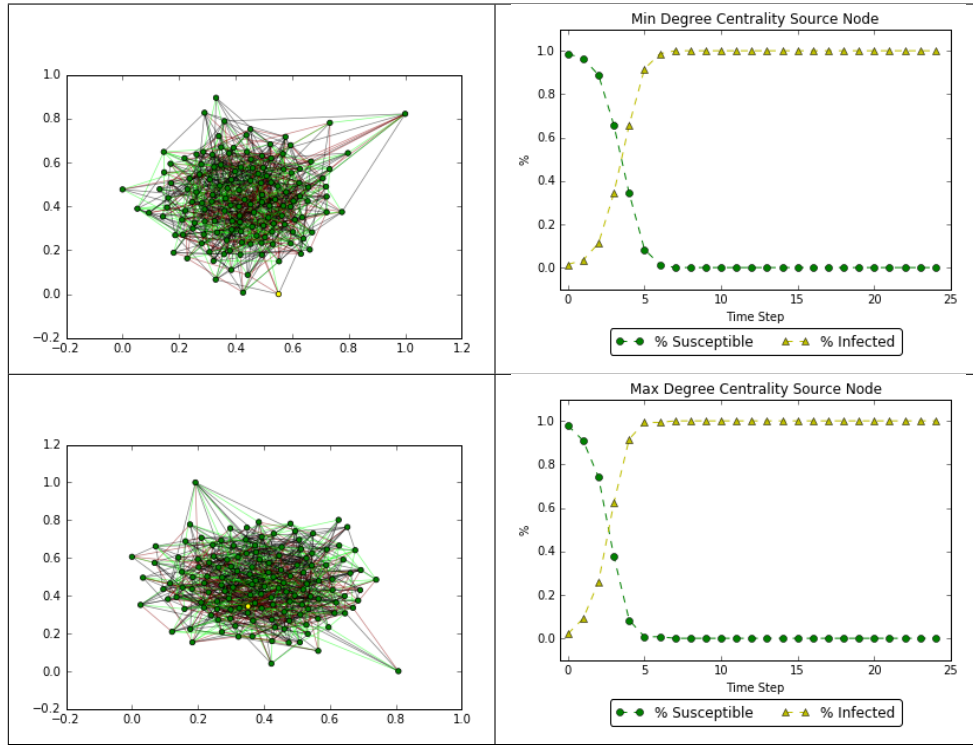


Figure 6: Simulation 1 Erdos-Renyi : Max and Min Degree Centrality SourceNode

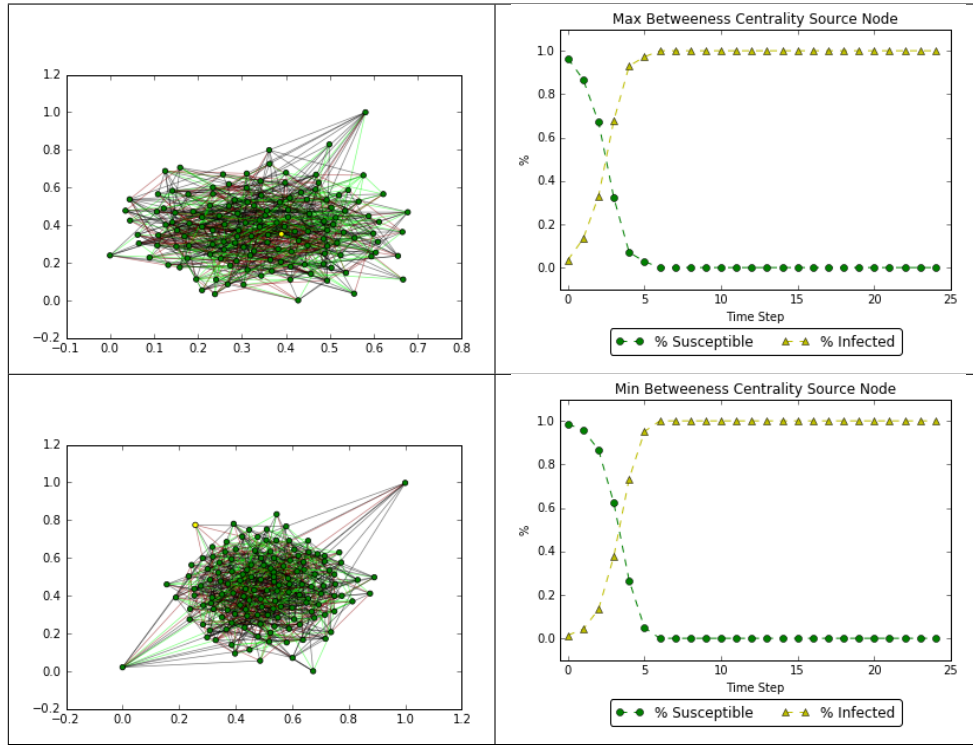


Figure 7: Simulation 1 Erdos-Renyi : Max and Min Betweenness Centrality SourceNode

## Real World Graph

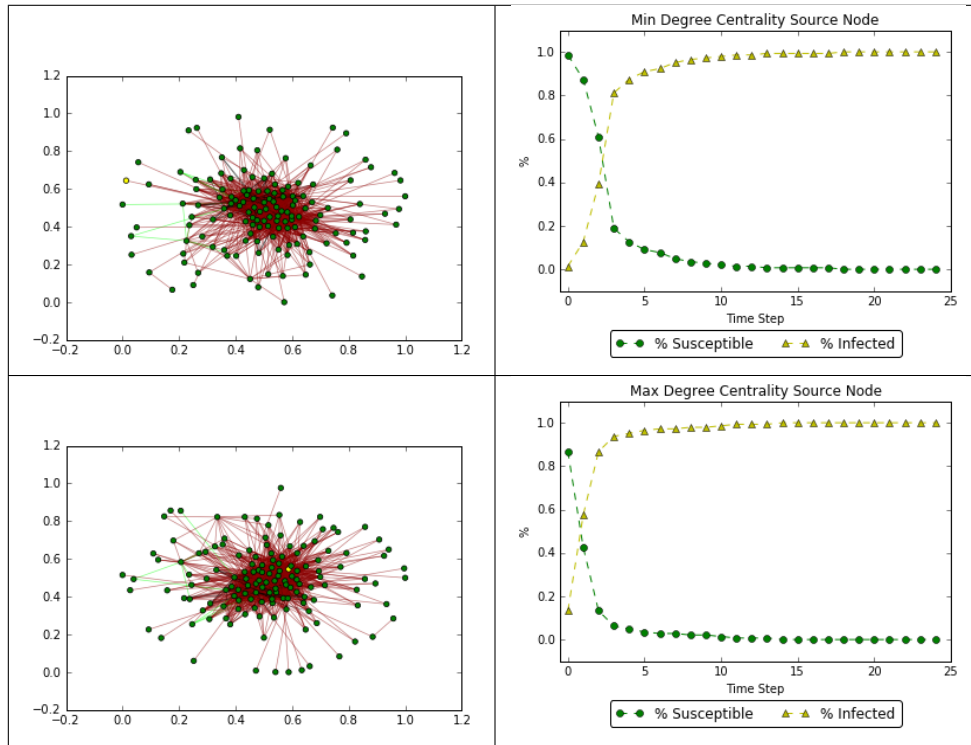


Figure 8: Simulation 1 Real : Max and Min Degree Centrality SourceNode

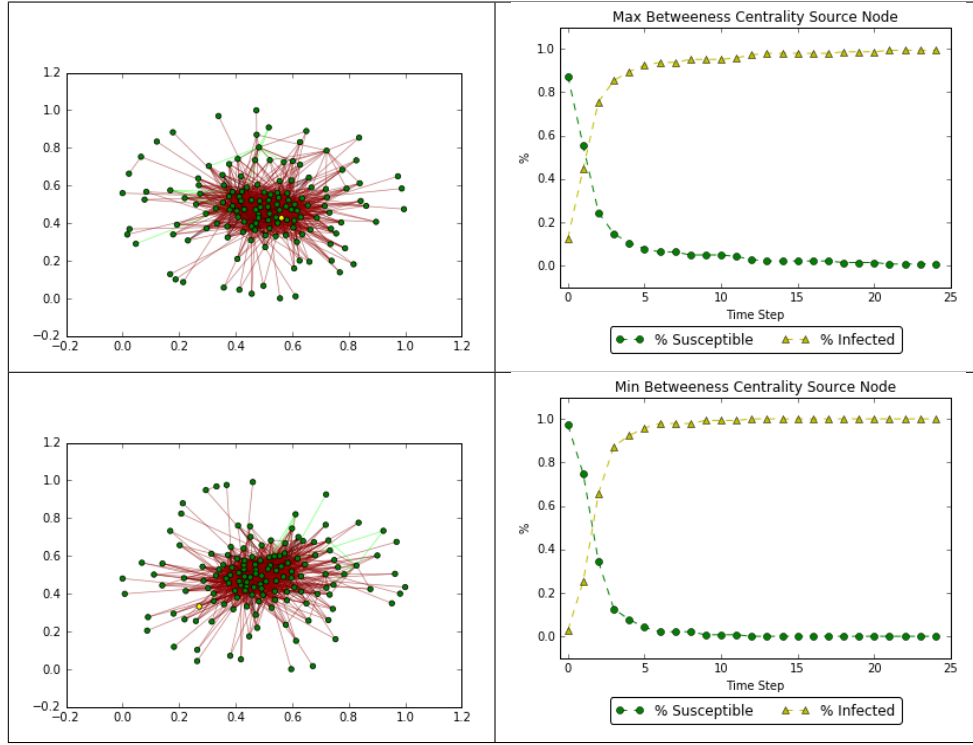


Figure 9: Simulation 1 Real : Max and Min Betweenness Centrality SourceNode

## Simulation 2

### Erdos-Renyi Graphs

index	pn	order	size	density	cluster coefficient	diameter	nodes deg $\geq n$	largest component
0	1	143	103	0.010144784792672116	0.008857808857808859	inf	0	81
1	3	143	310	0.030532847434255887	0.017199467199467203	inf	0	141
2	5	143	530	0.052201319806953611	0.045021276314982595	5	12	143
3	7	143	734	0.072293903279818772	0.07440109217101569	4	49	143
4	9	143	889	0.087560326996946714	0.09322985284361361	4	86	143
5	11	143	1162	0.1144489313503398	0.11609641591068731	3	128	143

Table 5: Simulation 2 Erdos Renyi : Graph properties for  $pn = 0.01, 0.03, 0.05, 0.09$

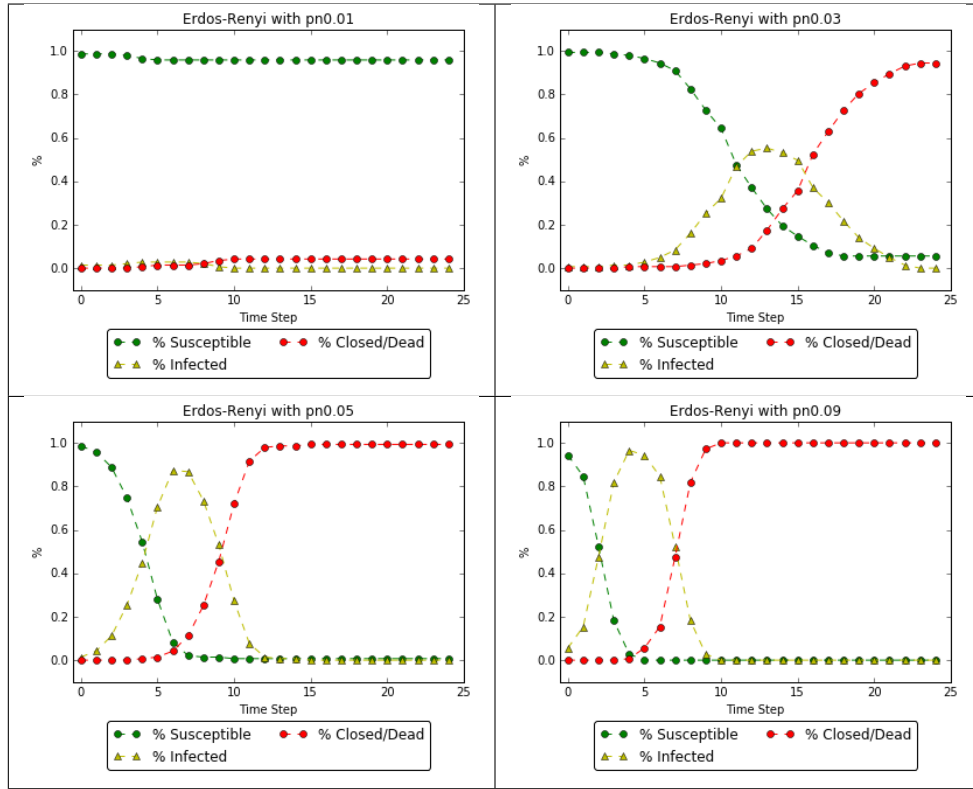


Figure 10: Simulation 2 Erdos Renyi : Plotting Node States for Graphs for  $p_n = 0.01, 0.03, 0.05, 0.09$

## Real Airports Graph and subgraphs

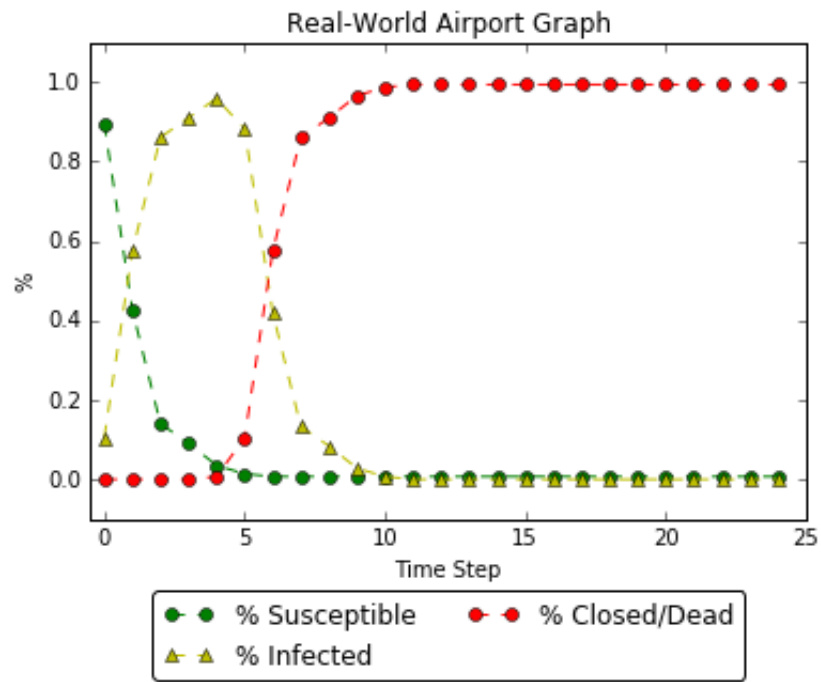


Figure 11: Simulation 2 Real-world : airport graph properties



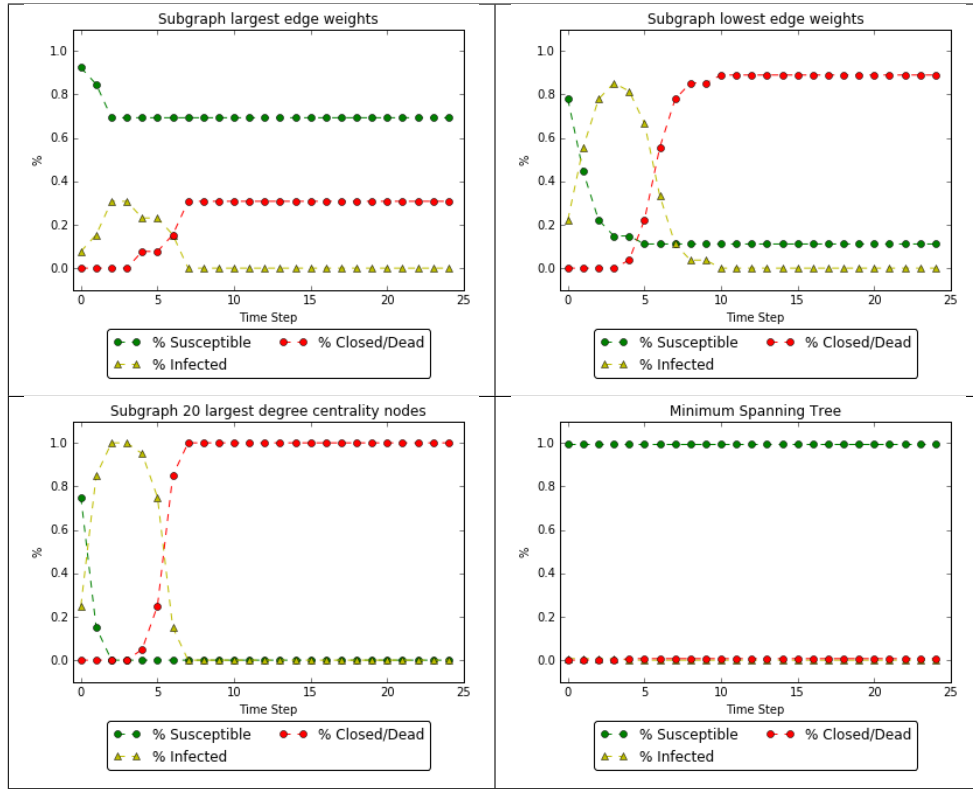


Figure 12: Simulation 2: Real-World Subgraphs

## Source Node Centrality

Erdos-Renyi Graph with  $p = 0.09$

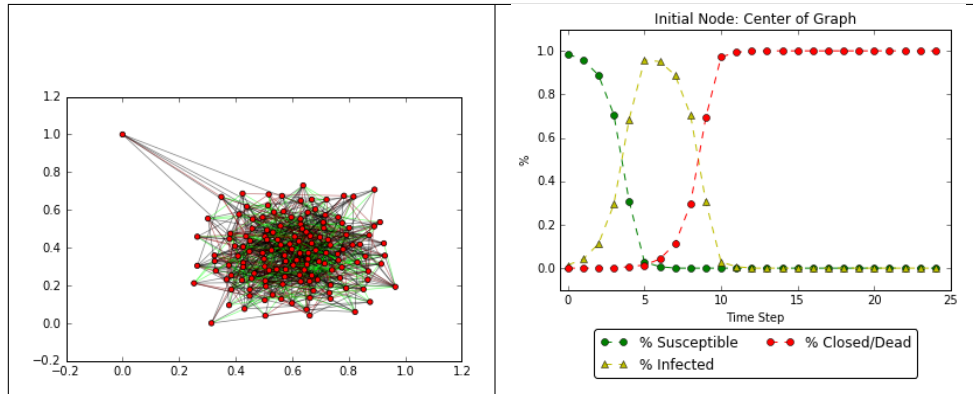


Figure 13: Simulation 2 Erdos-Renyi : Initial Node Center Of Graph

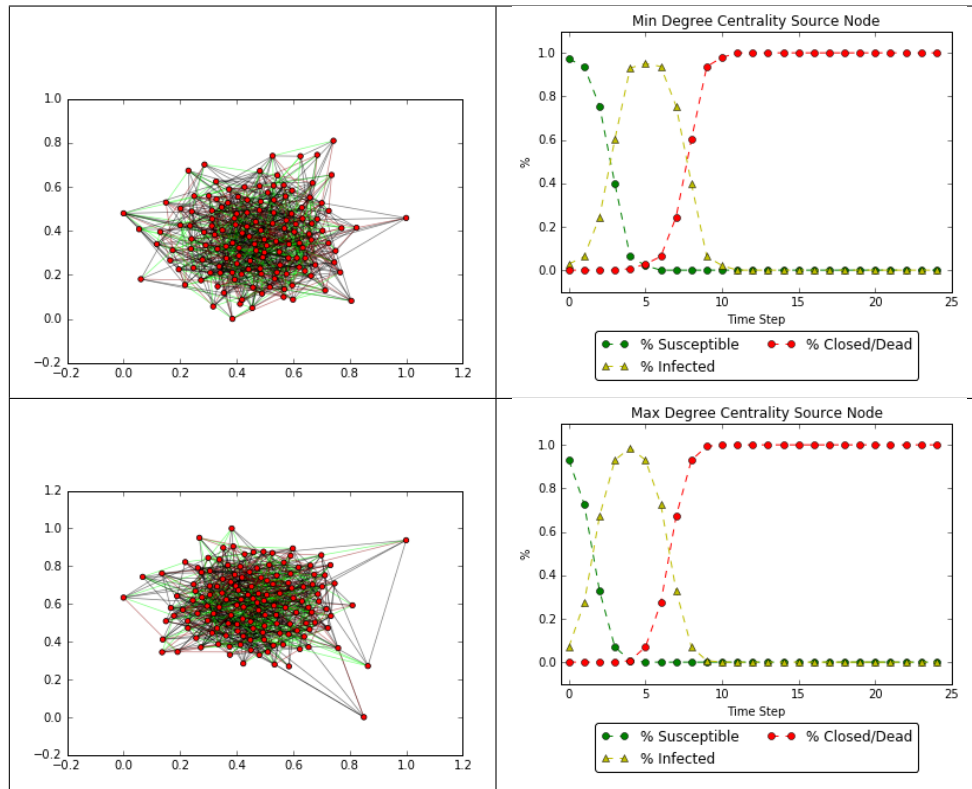


Figure 14: Simulation 2 Erdos-Renyi : Max and Min Degree Centrality SourceNode

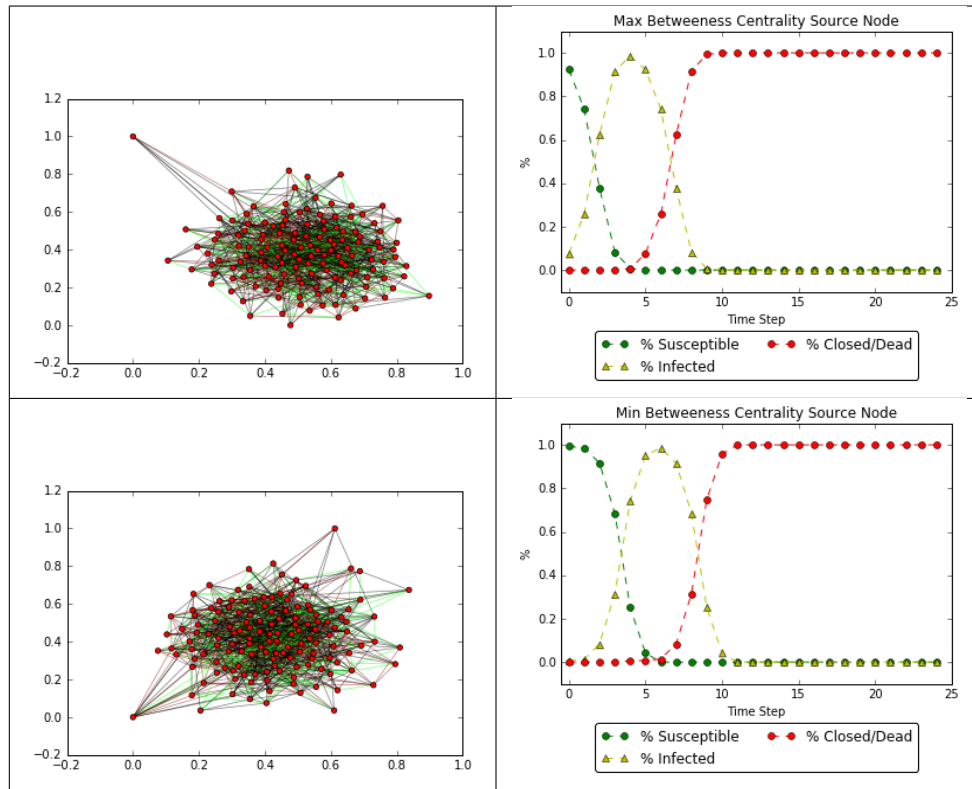


Figure 15: Simulation 2 Erdos-Renyi : Max and Min Betweenness Centrality SourceNode

## Real World Airports Graph

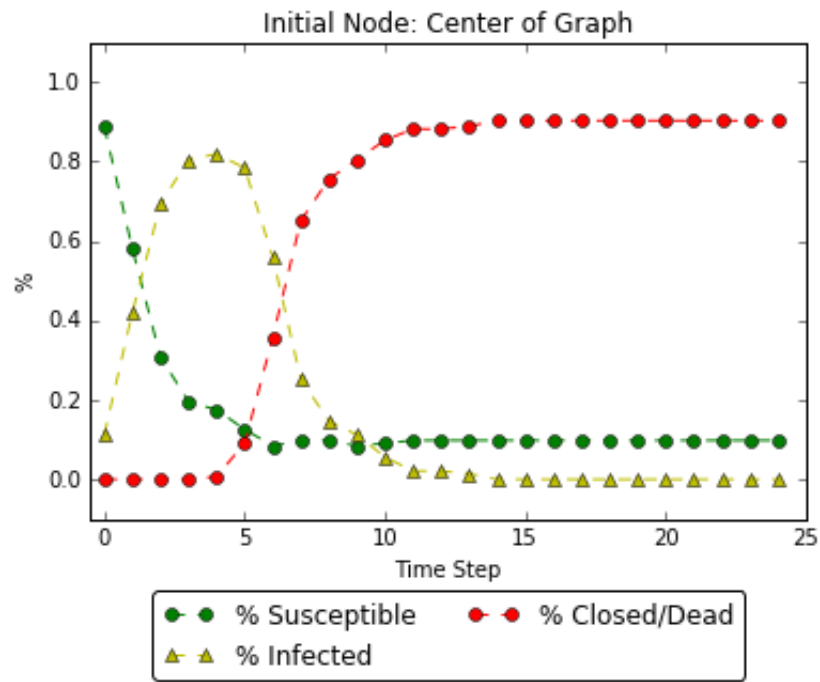


Figure 16: Simulation 2 Real-world : airport graph properties

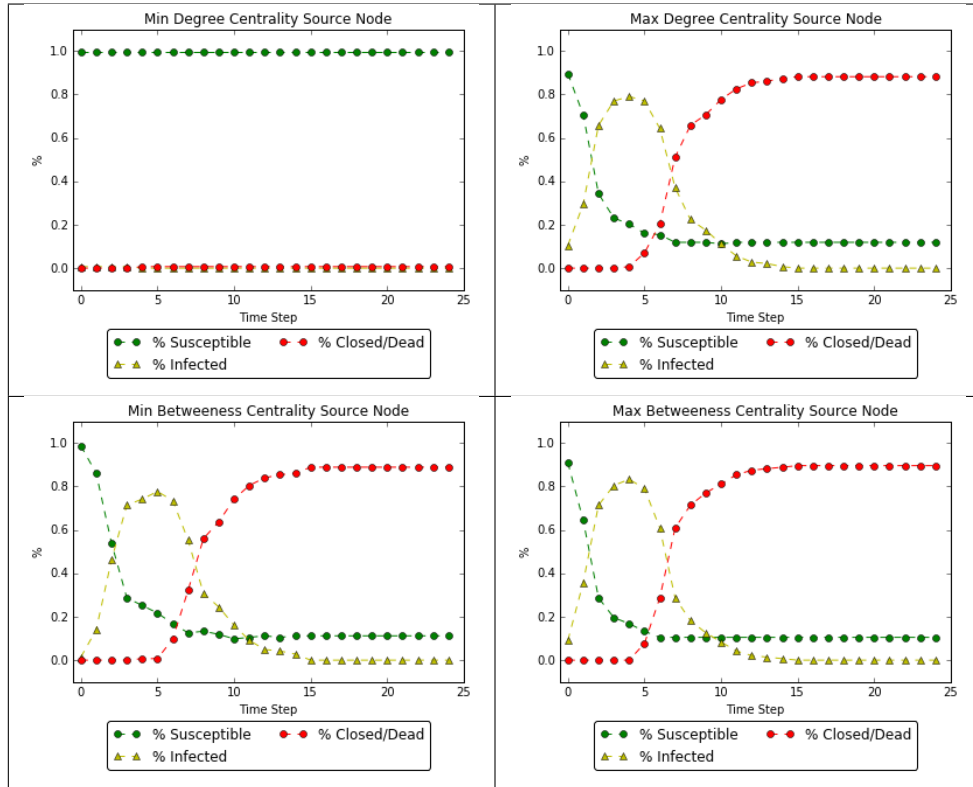


Figure 17: Simulation 2 Real : Max and Min Degree and Betweenness Centrality SourceNode

## References

- [1] JONES, K. E., PATEL, N. G., LEVY, M. A., STOREYGARD, A., BALK, D., GITTELMAN, J. L., AND DASZAK, P. Global trends in emerging infectious diseases. *Nature* *451*, 7181 (2008), 990–993.
- [2] LAWYER, G. Measuring the potential of individual airports for pandemic spread over the world airline network. *BMC infectious diseases* *16*, 1 (2016), 70.
- [3] TATEM, A. J., ROGERS, D. J., AND HAY, S. Global transport networks and infectious disease spread. *Advances in parasitology* *62* (2006), 293–343.
- [4] YAGER, N. A., AND TAYLOR, M. Edge-based control of disease propagation through the world-wide airport network. <https://github.com/nicholasyager/airport-disease-modeling>, 2014.