CS556 Project 1 - Modeling Worm Propagation in Large Networks

Alperen Tercan
Department of Computer Science
Colorado State University
Fort Collins, US
alperen.tercan@colostate.edu

I. Introduction

In this report, we will investigate how an internet worms spread under topology constraints. This analysis can be considered as a generalized version of [1], which models and analysis propagation of Code Red worm without topology constraints. No topology constraint is equivalent to a fully connected graph.

II. Methods

In this section, our experimental setup will be described.

A. Simulation

To simulate the spread of infection, we run a discrete-time simulation where each round infected nodes try to infect their susceptible neighbors according to a spread model that will be discussed in subsection II-B. The simulation continues until there isn't any node left that can be infected.

When the cure is introduced, we extended our approach to a turn-based simulation. Each round consist of two phases: infection and cure. During the infection phase, infection spreads to uncured and uninfected nodes. Then, in the cure phase, cure spreads according to the same spread model. The simulation continues until there aren't any infected nodes left.

B. Spread Model

When modeling the spread of the infection, we have considered three different approaches. Firstly, we considered an unlimited resource scenario where an infected node can infect unlimited number of adjacent susceptible nodes at each simulation step. Following RedCode example, this would be assuming the number of threads an infected computer can run in parallel is infinite. However, since the networks that we use are topology constrained; number of threads is still limited by the degree of the node. In this model, every round, each infected node samples d random values from a Bernoulli distribution with $p = p_{\text{inf}}$, where d is the degree of the node and p_{inf} is the infection probability. Each value decide whether the corresponding neighbor will be infected or not. We implement this model from the susceptible node point-of-view. So, for each susceptible node, number of infected neighbors, k, is computed. Then we sample a random Bernoulli random variable with $p = 1-(1-p_{\text{inf}})^k$. This approach reduces number of random variables to sample.

A second approach was to limit the number susceptible neighbors an infected node can try to infect every round by a hyperparameter m. For example, in RedCode example, m was 100, the number of threads. It is important to note that for scale-free graphs, degree of some hub nodes can be very large; hence, making a realistic choice of m important. Whereas, maximum degree of a node in binomial and small-worlds graphs usually small enough to make unlimited case realistic enough, ie. no node will be overwhelmed by the number of susceptible neighbors it has. In this model, every round, m susceptible neighbors are chosen for each infected node. Then, m Bernoulli random variables with $p = p_{\text{inf}}$ are sampled for each infected node to decide whether these neighbors will be infected or not.

A third approach was proposed by Prof. Ray. This limits the number of an infected node can infect instead of try to infect. While the difference may seem subtle, its practical impacts are important. For example, when $m = 1$, probability of an infection between an infected node and its susceptible neighbor is $\frac{1}{d}(1-(1-p)^d)$ in the third approach whereas it is $\frac{p}{d}$ in the second; where d is the number of susceptible neighbors of the infected node.

In this report, we will use mainly approach 1 and call it unlimited resources model. If not stated otherwise, assume that this model is used.

We will also do some experiments with the third approach, and we will call it single infection model. While the second approach is also interesting; we won't be presenting any results with it due to space and time constraints.

C. Graph Topologies

In this work, we experiment with three random graph topologies: binomial, scale-free, and small-worlds. An important decision is to decide how to define comparable graphs with these different topologies. As each topology is connectivity-wise inherently different; which is also the fact that we're investigating; how to make a fair comparison.

For this, we will compare different topologies consisting of same number nodes, n, and edges, m. We will sample the binomial graphs uniformly from all graphs with n nodes and m edges. Similarly, we will generate the scale-free graphs as each new node attached to $\lfloor m/n \rfloor$ existing nodes.

To generate a small-worlds graph, firstly. a ring of n nodes is created. Then, we connect each node to $2\lfloor m/n \rfloor$ closest neighbors; which give m total edges. However, unlike the other topologies, small-worlds graphs have another parameter to control the topology of the network, rewiring probability p_r. After connecting to closest neighbors, we replace each edge (u, v) with probability p_r. New edge is (u, w) decided by uniformly choosing an existing node w. This rewiring effectively creates shortcuts between the small-worlds and make the graph more similar to a binomial graph. In the experiments, we will show that while this parameter greatly affects the propagation; our analysis is consistent across its values.

D. Experiment Setup

As we're working with random graphs, the observed propagation behavior can be severely affected by the used instance. In order to reduce the variance, we generate multiple instances of each topology and run the simulation on each several times. For Program 1, we used 5 instances of each topology and run the simulation 10 times on each. For Program 2, as a more distributed approach was followed when running the experiments; 12 instances and 4 runs on each were used. This allowed us to run experiments in parallel on more workers without needing to share the graph. These random graph are generated via networkx package. [2]

When presenting the behavior, we will report the mean and standard deviation of runs, implicitly assuming that the behavior metrics of a graph topology are sampled from a normal-like distribution. While we will leave out the theoretical justification as it is not in the scope of this work, Figure 1 empirically shows that this is true.

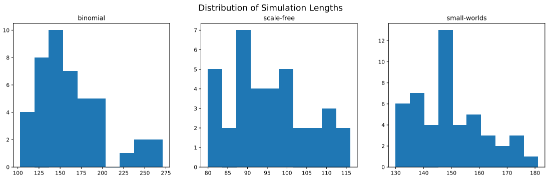

Fig. 1. Distribution of simulation lengths across different random seeds and graph instances for each topology with $p_{inf} = p_cure = 0.03$ and graph size (10000,50000). The results show that they show a gaussian-like distribution.

III. Results - Program 1

In this section, we will present some of the main results of Program 1, ie. infection without cure.

A. Comparing Across Infection Probabilities

Figure 2 is our first main result. As can be seen, infection takes shortest time in scale-free graph and longest in small-worlds. A significant observation is that spread in small-worlds graphs gets close to the binomial case as rewiring probability increases. This is expected, as creating shortcuts by rewiring makes small-worlds graphs to lose their distinct character of consisting of "small islands" and get closer to a binomial graph. Due to this behavior, we'll leave out $p_{\text{rewire}} = 0.35$ case out in the future graphs for clarity.

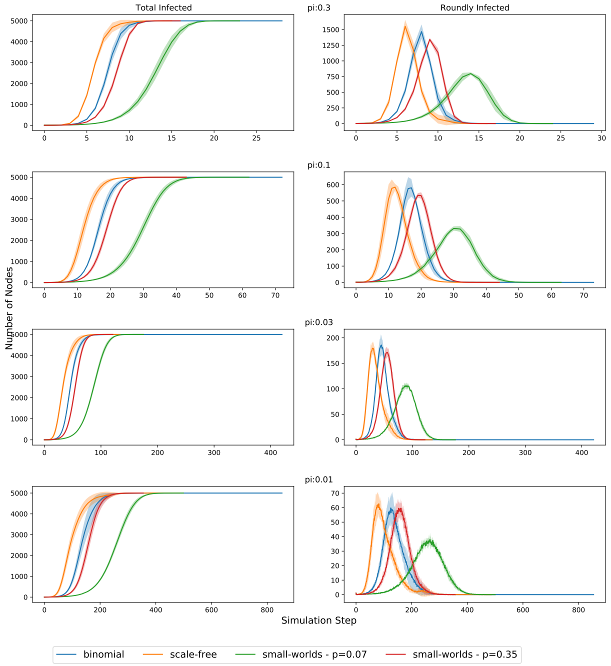

Fig. 2. Comparing infection behavior with unlimited resources of different topologies under different infection probabilities. Note that there are two different versions of small-worlds graph: one with rewiring probability 0.07 and other with 0.35. Graphs has 5000 nodes and 20000 edges. Shaded regions show $\pm\sigma$. Note that shaded regions usually not visible, indicating the statistical significance of our results.

As expected, as infection probability decreases, time to infect all nodes increases.

Comparing Figure 3 with Figure 2 shows another interesting insight into relation between spread models and graph topologies. For high probabilities of infection, infection spreads faster than scale-free graphs. Because we allow only 1 infection per infected node, no matter their degree, hub nodes actually becomes bottlenecks. They cannot utilize their highly connected status and since we keep the number of edges same across topologies, others part are less connected than a binomial graph.

This effect is not visible for small probabilities because hubs aren't able to infect that many neighbors anyways. In other words, infection probability becomes the bottleneck instead of infection limit.

Figure 4 shows that while it is not visible from infection curves; in fact it takes longer for binomial networks to be fully infected. The reasons will be discussed in subsubsection V-A3.

B. Comparing Across Graph Sizes

Figure 5 and Figure 6 show that the comparison between different topologies holds across different network sizes for large networks.

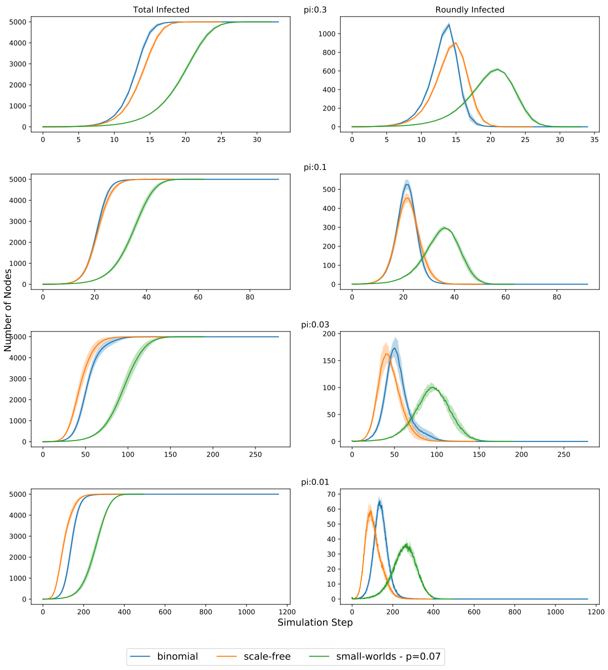

Fig. 3. Figure 2 figure but with single infection spread model. Also small-worlds with $p_{\text{rewire}} = 0.35$ is left out.

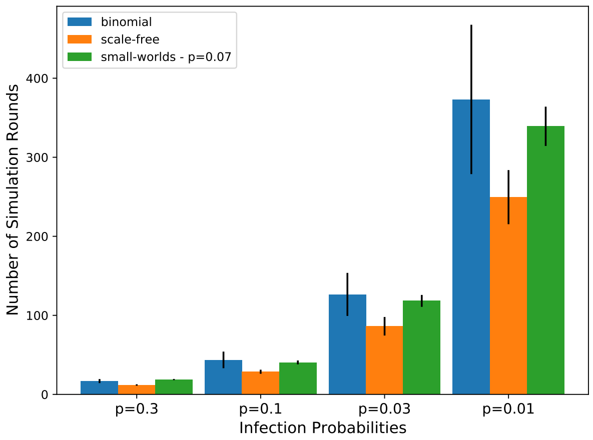

Fig. 4. Time to fully infect the network for different infection probabilities and different topologies. Network size is (10000,50000) and unlimited resource spread model is used. Error bars show $\pm\sigma$.

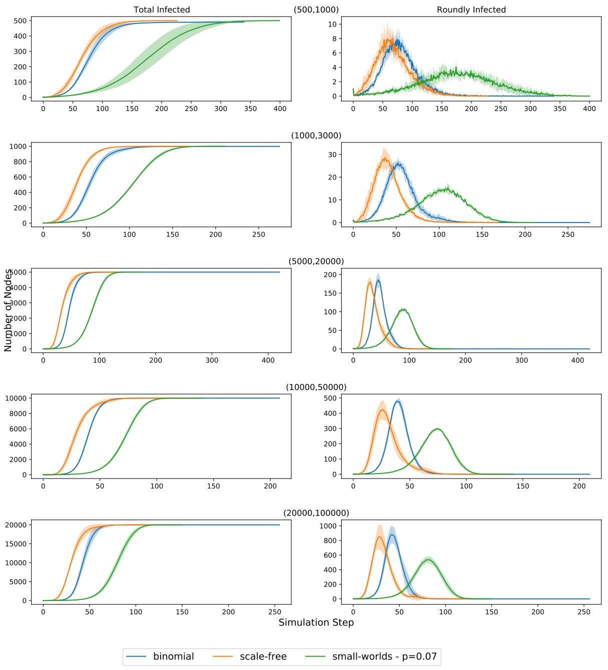

Fig. 5. Comparison of infection rates on graphs of different sizes. All figures use $p_{inf} = 0.03$ and unlimited spread model.

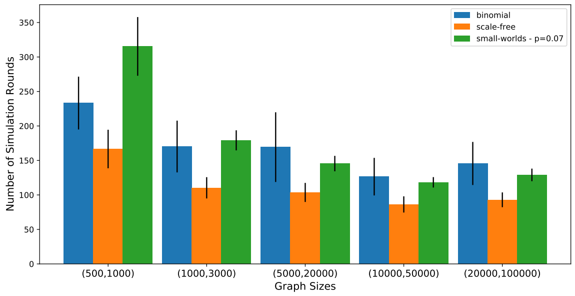

Fig. 6. Time to completely infect the network. All figures use $p_{inf} = 0.03$ and unlimited spread model. Error bars show $\pm\sigma$.

An interesting observation is that average degree is a much more important metric than the network size when comparing time to infect a network.

While increasing network size while holding average degree slightly increases time to fully infect, as can be seen by comparing $(10000, 50000)$ and $(20000, 100000)$; the smaller networks has much higher time to fully infect because of their smaller average degree. This suggests that infection spread scales really well.

Note that for the smallest networks, binomial takes shorter to fully infect than small-worlds. This will be addressed in subsubsection V-A3.

IV. Results - Program 2

In this section, we will present some of the main results of Program 2, ie. when a cure is introduced.

A. Introducing A Cure

Figure 7 is provided to show how simulation unfolds. As our analysis focus on infection rates, total and roundly cured results will be omitted in the future figures. Notice that total infection figure still has the S-shape. This is because we

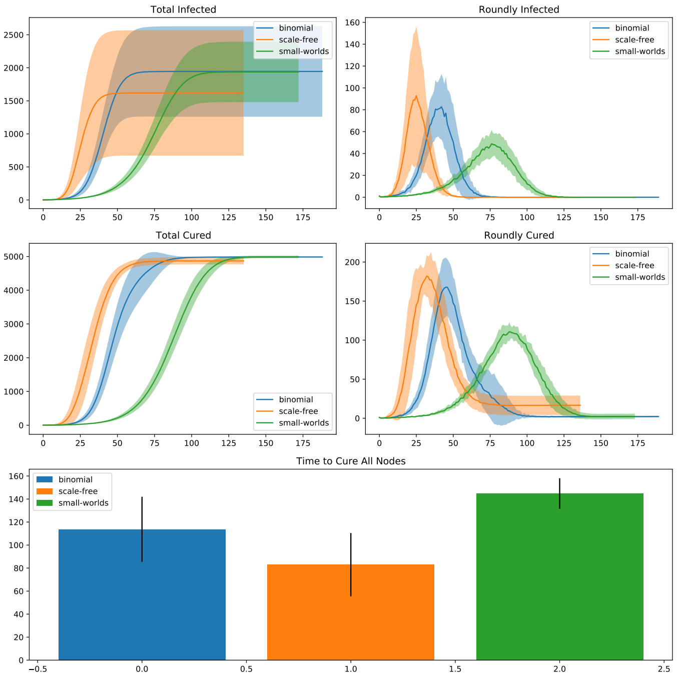

Fig. 7. Example simulation results when a cure is introduced. Graph size (5000,20000) and $p_{inf} = p_{cure} = 0.03$.

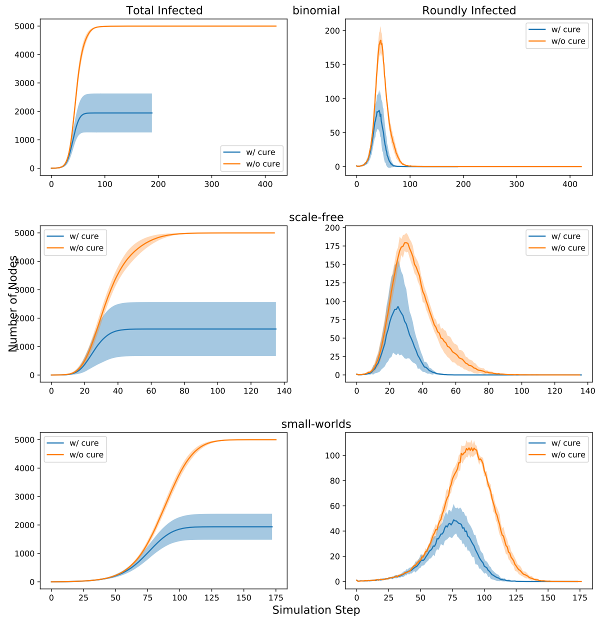

Fig. 8. A comparison of w/ cure and w/o cure spreads across different topologies. Graph size (5000,20000) and $p_{inf} = p_{cure} = 0.03$.

count previously infected but cured nodes too, following the convention of [1].

B. With Cure vs Without Cure

Figure 8 shows how introducing a cure affects the infection spread. As expected, it reduces number of infected nodes. In the next two sections, we will see how varying infection probabilities and graph sizes affect this difference.

C. Compare Across Different Infection Probabilities

Figure 9 show that scale-free networks end the epidemic fastest and usually fewer nodes get infected. However, variance is much higher than other topologies; suggesting the variance is high across different instances and simulation runs. We believe that the differences are due to how soon hub nodes get infected.

Also note that while networks have 5000 nodes, for low infection probability only a portion of them get infected at all. This happens because the cure spread immunizes nodes and severely reduces spread of infection. However, when infection probability is high, infection spreads faster than cure and weakens this effect.

Figure 10 shows a similar pattern with Figure 4 for high infection probabilities; as in these cases impact of the cure is very small.

However, for small infection probabilities; simulation ends earlier in binomial networks than small-worlds networks, in contrast to Figure 4. We believe that this happens because having a cure solves ”parts of network with very small connectivity” problem of binomial networks. Since the epidemic ends when there aren't any infected nodes but not when everyone cured; there is no need for cure to spread to small-connectivity regions of binomial graphs. This claim is supported by the fact that in $p_{inf} = 0.01$ case; significantly fewer nodes are infected at all in binomial networks than small-worlds networks.

An important observation is in variance is much higher when cure is introduced. While this could be attributed to the changed experiment setup, which uses more instances of networks and fewer runs on the same graph, the results for $p_{inf} = 0.3$ in Figure 9 suggest otherwise. In Figure 2 standard deviations are similar across different infection probabilities; however, they significantly reduce in Figure 9 when infection probability increases. This indicate that the high variance is caused by the stochasticity of how ”smartly” the cure will spread because we don't see this when infection probability is high, removing impacts of cure.

D. Compare Across Graph Sizes

Figures Figure 11 and Figure 12 are provided for the sake of completeness, and don't directly add to the discussion.

V. Discussions and Answers

In this section, we will discuss the results to answer the questions.

A. Question a)

All of our experiments with program 1 show S-shape curve for total infection and bell-shape for roundly infection. This occurs because of the epidemic mechanics of a finite population. Initially, there are a few infected nodes, so they can only

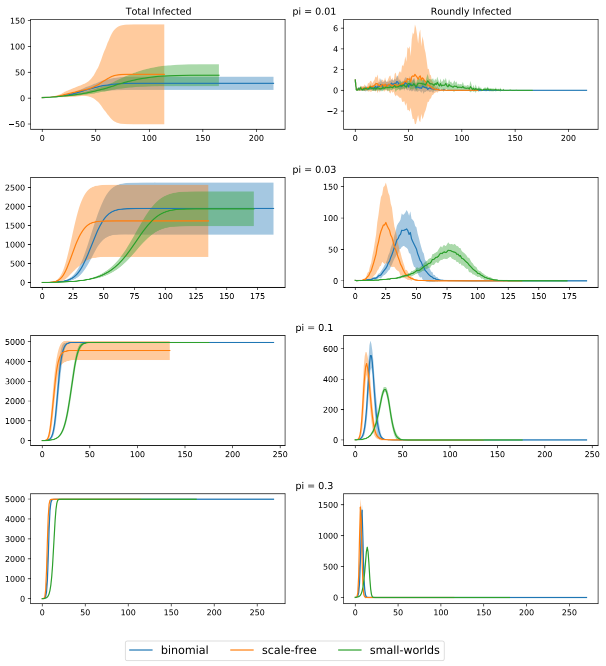

Fig. 9. Comparing infection behavior with unlimited resources of different topologies under different infection probabilities when there is a cure. Graph size is $(5000, 20000)$. Shaded regions show $\pm\sigma$.Network size is (5000,20000) and unlimited resource spread model is used. Cure spread probability is always $p_{cure} = 0.03$.

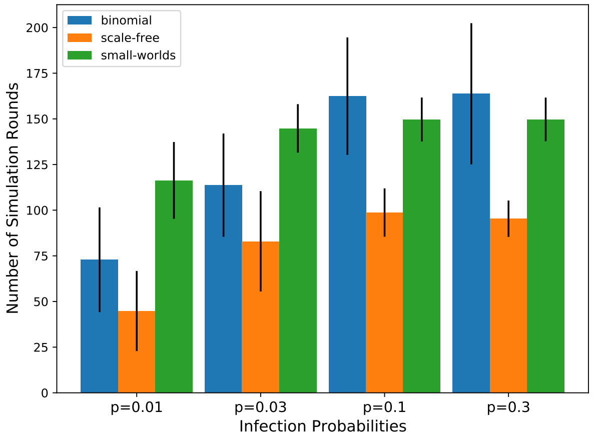

Fig. 10. Time to fully infect the network for different infection probabilities and different topologies. Network size is (5000,20000) and unlimited resource spread model is used. Cure spread probability is always $p_{cure} = 0.03$.

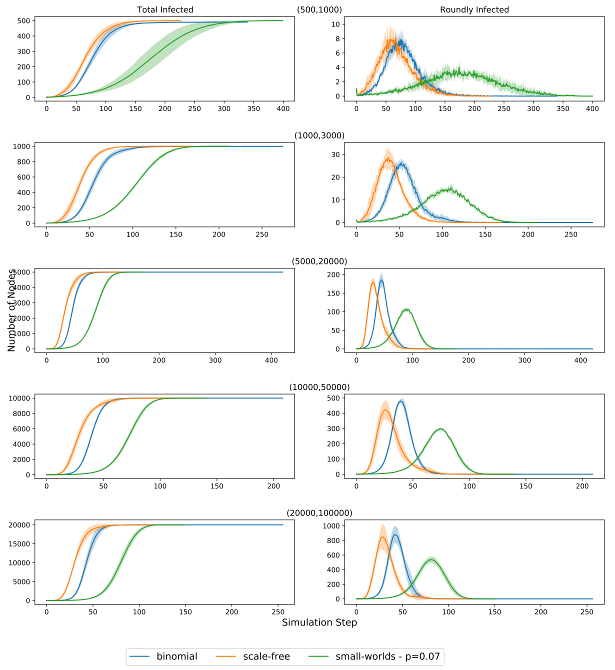

Fig. 11. Comparison of infection rates on graphs of different sizes. All figures use $p_{inf} = p_{cure} = 0.03$ and unlimited spread model.

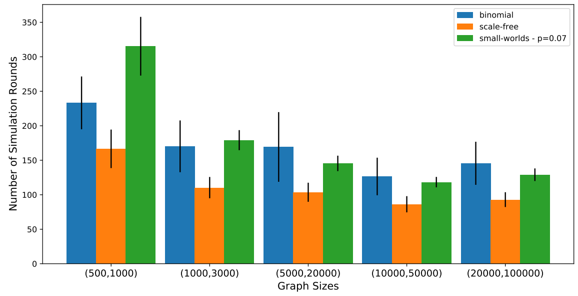

Fig. 12. Time to completely infect the network. All figures use $p_{inf} == p_{cure} = 0.03$ and unlimited spread model. Error bars show $\pm\sigma$.

infect a small number of nodes every round. As the number of infected nodes increase, number of nodes they infect each round also increases. Since the number of infected nodes is still very small in comparison to whole population, it is safe to assume that most of their neighbors are still susceptible. Due to aforementioned observations, we see an exponential-like increase in roundly infections at the beginning.

However, at some point, number of infected nodes reaches to a point where most neighbors of infected nodes are also infected. Hence, the roundly infection numbers saturate. Then, finding susceptible nodes and infecting them gets increasingly difficult as only rather isolated nodes remain susceptible to this point. So, we see an exponential-like decrease in this part. As total number of infected nodes is just a running sum of this bell-shape; S-shape naturally follows. Intuitively, same arguments also hold for total infection too.

1) Scale-free Networks: Also, all of our results show that (eg. Figure 2) the infection rate depends on the network topology. While the analysis done above hold for all; hence, we still see bell-shape and S-shape curves; the curves significantly and consistently differ between topologies.

Particularly in the unlimited infection resource scenario(see subsection II-B for more information), infection spreads much faster in Scale-free networks. This is due to the fact that these networks are much more likely to have some hub nodes that are connected to many nodes. This speeds up the spread in two ways. Firstly, hubs are very likely to quickly get infected because as they have many neighbors; it is very likely that some of their neighbors are infected and gets them infected too. Then, infected hubs can very quickly spread the infection and they don't saturate for a long time, again thanks to having many neighbors.

Please see subsection III-A for an analysis of the spread model Prof. Ray proposed. It presents and discusses an interesting phenomenon of hubs becoming bottlenecks instead of super-spreaders.

2) Small-worlds Networks: Moreover, we see that small-worlds networks having the slowest spread for the most part of the infection process. This is due to the fact that small-worlds graphs mostly look like a ring where each nodes are only connected to their closest neighbors. This in a way makes spreading a one-dimensional process; hence, greatly reduces the intersection between infected and susceptible nodes, ie. the infection can spread through only a few nodes.

But Watts-Strogatz networks also have a rewiring mechanism that creates shortcuts. These shortcuts create a possibility of jumps across the ring. These jumps allow the infection spread starting from different parts of the network. Figure 2, indeed shows that increasing rewiring probability greatly speeds up the spread.

3) Binomial Takes Longest to Fully Infect: It is important to note that these analysis are made in terms of time to infect the "most" but not the "whole" of the network. When we look at the average time to infect the full network; binomial graphs take the longest.

This happens because in binomial graphs, it is likely to have some parts with very small connectivity and some parts with more connectivity. The parts with very small connectivity tends to be small too; so, most of the infection process is not affected from this. But, when we measure the time to fully infect the network; it can take a very long time to get these nodes infected.

On the other hand, while the arguments made in subsubsection V-A2 still hold; all nodes in small-worlds networks initially have the same degree. This changes with rewiring; but for small rewiring probabilities, it would be mostly maintained. Hence, there aren't such nodes with very small connectivity. This is also shown in Figure 4. Also, notice that error bars are much larger in binomial graphs. This is because they are the least structured random graphs among these three; hence, they exhibit more diverse behavior. In this case, some instances contain more small connectivity parts and some contain less; hence, leading to a wide spectrum of different infection times.

An exception is when the average degree is very small, as shown in Figure 6. Since the average degree is already too small in this case, the analysis of parts with very small connectivity doesn't effectively apply. For example, in $(500, 1000)$ case, small-worlds nodes are only connected the nodes on their right and left; effectively creating a chain. This is already a thinly connected as possible, any less connectivity would create disconnected nodes.

B. Question b)

There are several similarities with the current pandemic. Firstly, Figure 13 shows the daily and total cases, which is very similar to first half of the number of infected nodes curves that we obtained our experiments. This was expected as both are supposed be following a generic epidemic model.(See [1].)

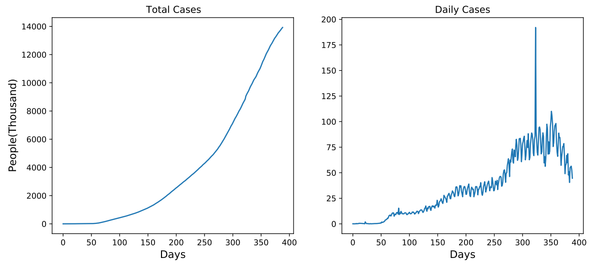

Fig. 13. Worldwide daily and total cases of COVID-19. Raw data is obtained from https://ourworldindata.org/coronavirus-dataandvisualized.

Moreover, our results on how different topologies affect the spread give insights on how to reduce impacts of the pandemic.

For example, the fast spread in scale-free networks suggest that having active hubs in the society is problematic. These hubs can be markets, schools, workplaces. This is a loose comparison as these places are not actual nodes but groups of nodes with very high connectivity within and many connections to outside. However, the argument still holds that removing such places would help.

Also, the small spread in small-worlds networks in comparison to binomial networks is a supporting evidence for travel bans. Many critics, rightly, point out that no travel ban is perfect; and there are still people travelling between countries/cities, particularly between neighboring ones. But our results show that even such an imperfect travel ban is still quite useful. Those travelling people are accounted for in our model, through shortcuts of Watts-Strogatz and nodes being connected to their neighboring nodes. And still we see much smaller spread than binomial graphs.

C. Question c)

It reduces the infection rate. See section IV, and in particular subsection IV-B for a detailed analysis.

D. Question d)

Without cure case was explained in detail in subsection V-A; so, we skip it to avoid being repetitive.

When a cure is used, as presented and discussed in subsection IV-C, the infection can be cleaned from the networks with hub nodes faster. Particularly, see how binomial networks compare to small-worlds in with and without cure scenarios.

Moreover, networks with hub nodes exhibit a much higher variance. We believe that spread in this nodes highly depend on when hub nodes are cured. Again, see subsection IV-C for future details.

E. Question e)

Building on our previous results and discussions, we believe that curing the nodes with highest degrees first would be a much better approach than random curing. This method should have the most impact in scale-free networks, which exhibits very high degree hub nodes. And it should be the least useful in small-worlds networks, as degrees of their nodes are very similar to each other.

To confirm our hypothesis, we generated 3 graph instances with (5000,20000) size from all three topologies and identified two nodes with highest degrees. Then, we run Program 2($p_{inf} = p_{cure} = 0.03$) on each graph twice: 1) initially curing random 2 nodes 2) initially curing the identified hub nodes.Figure 14 shows that our approach leads to significant benefits. Also, our analysis on how each topology will be affected was accurate.

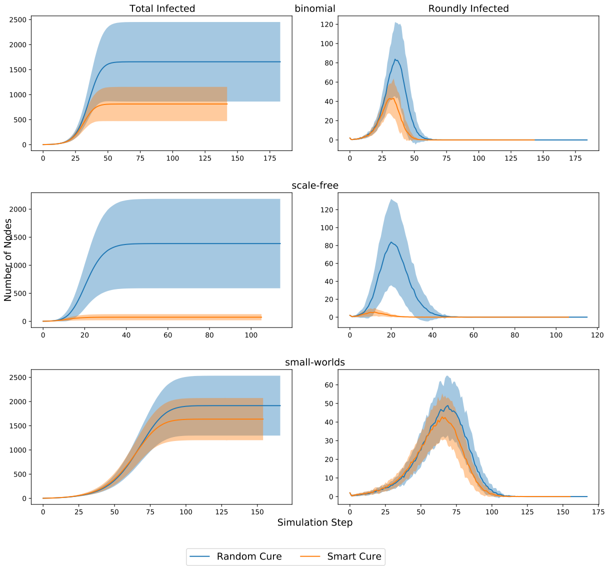

Fig. 14. Infection spread comparison between random initial curing and ”smart” initial curing.

VI. Conclusion

In this work, we’ve investigated worm propagation in topology constrained networks both with and without a cure. Our results show that topology of the networks has a great impact in infection spread behaviors.

Finally, in section V, answer and discuss the questions in project proposal. Source code for this work can be found on www.github.com/alperentercan/worm-propagation.

References

[1] C. C. Zou, W. Gong, and D. Towsley, “Code red worm propagation modeling and analysis,” in *Proceedings of the 9th ACM Conference on Computer and Communications Security*, ser. CCS ’02. New York, NY, USA: Association for Computing Machinery, 2002, p. 138–147. [Online]. Available: https://doi.org/10.1145/586110.586130

[2] A. A. Hagberg, D. A. Schult, and P. J. Swart, “Exploring network structure, dynamics, and function using networkx,” in *Proceedings of the 7th Python in Science Conference*, G. Varoquaux, T. Vaught, and J. Millman, Eds., Pasadena, CA USA, 2008, pp. 11 – 15.