Network Destroy-Repair Game - Milestone 2

Runqi (Tony) Huang
Carnegie Mellon University
Department of Electrical and Computer Engineering
AndrewID: runqih
runqih@andrew.cmu.edu

Vikas Kashyap
Carnegie Mellon University
Department of Electrical and Computer Engineering
AndrewID: vhuvinah
vhuvinah@andrew.cmu.edu

***Abstract*—Robustness is an important functionality of real-world networks, as it provides the resilience to guard against external attacks. Suppose an adversarial agent were to attempt to destroy the network, what kind of mechanism should he employ and how should the network defend itself? In this project, we analyze four real-world networks, where resilience to external threats is essential to the network's users and operations. We design two sets of algorithms that either randomly or selectively attack and defend the network. Specifically, we will vary the proportion of edges that are being targeted and defended. Through simulation, we will show that both types of algorithms cause the network's robustness to deteriorate over time, regardless of the attack and defense proportionality, but the network under the smart algorithm is relatively more robust than under the random algorithm. This report is a continuation of the previous milestone 1 report.**

I. Introduction and motivation

Real-world interactions and relationships can often be formulated in terms of networks and graphs, and these networks play a significant role in people's daily lives. In this immensely complicated world, perfection is idealized but never achieved, and seldom is a network designed without flaw. Without proper safety features, networks will inevitably be exploited by malicious actors (e.g., hackers) and failures of the network can lead to severe consequences, sometimes with human lives at stake [1]. From the perspective of the hackers, how should he design the attack scheme to maximize the damage? From the perspective of the defenders, how should he defend the networks to minimize the damage? The formal description to the above problem leverages the concept of network robustness, which measures the network's resilience and ability to withstand external attacks.

Given the importance of network robustness, this project will investigate four scenarios where safety is integral to the performance, operation, and user experience of the networks. We will leverage real-world data sets that are highly relatable and design two sets of attack and defense algorithm. One algorithm is random in nature, while the other algorithm will more intelligently (based on certain rules to be discussed below) target the edges to be destroyed and repaired. We will simulate the network's behavior under the proposed algorithms and monitor the network's robustness as time evolves. Through a careful selection of the data sets and experiment methodology, we study different network destroy-repair scenarios and contribute to a better understanding of network robustness.

II. Previous work

Recent years have witnessed the rise of deep learning, including deep learning on graph structures, which has shown promising results. Deep graph networks are similarly susceptible to adversarial attack, and one method adopts reinforcement learning that learns to attack the network based on long-term rewards [4] [5]. Due to a lack of reward mechanism, this project will adopt an alternative, heuristic-based approach. However, it can be seen that by introducing the proper reward mechanisms, networks, including ones we will study, may potentially strengthen themselves through a reinforcement learning approach.

There are various metrics to measure network robustness [6] [7]. One of the metrics is the Molloy-Reed criterion [1] [3], which connects network's integrity with the network's degree k. Specifically, the criterion states that the network has a giant component if the following condition is satisfied:

$$\kappa := \frac{\langle k^2 \rangle}{\langle k \rangle} > 2 \tag{1}$$

In this case, this means that the higher the value of κ, the more robust the connectivity of the network becomes. We have chosen Molloy-Reed criterion over other metrics such as shortest path global efficiency metric [8] to save compute resources as the latter uses Dijkstra algorithm to compute the shortest path and measure robustness.

III. Approach and steps

A. Main questions

The main question in this project focuses on the effectiveness of the two algorithms in protecting the robustness of the networks. Will the smart algorithm provide a stronger protection than the random algorithm? Is the smart algorithm equally effective across the four networks?

To achieve this end, we will approach this project by following three main steps, and in this section, we will provide the details to each of the steps.

B. Step 1: Gathering data sets and setting assumptions

We will use the Stanford SNAP data set [2]. This data set includes a variety of network scenarios with a large number of nodes and edges, which can serve as a good representation of their real-world counterparts. We have selected four networks for the following reasons:

1) Social network: Criminals can disguise themselves and connect with people on social network. These unfriendly connections can be fraudulent and lead to financial loss. We will study the Twitter network dating to 2012 [2], where the nodes represent the users and edges the connections between the users. Note that the edges are unweighted and directed, representing following on social network.
2) P2P network: On peer-to-peer network, malware can be transmitted to users and useless data (poison) can also be injected to damage the network. We have selected the Gnutella network, dated year 2002 [2], where the nodes represent the hosts in the network and edges connections between the Gnutella hosts. The edges are unweighted and directed.
3) Autonomous system (AS): Security poses a fundamental challenge to the internet. In some cases, national security can be at stake without strong network integrity. We have selected the AS network from Oregon dating to March 2001 from [2]. The nodes are the routers in the AS and the edges are the physical connections between the routers. The edges are unweighted and undirected.
4) Email network: Spams can become an annoying headache and detrimental to user experience. Phishing emails also poses security risk and exposure of personal information. We have selected the Enron email communication network from [2]. The nodes of the network are email addresses and if an address i sent at least one email to address j, an association is established between the two parties. The edges are unweighted and undirected.

Having chosen the networks, we make the following assumptions about the networks and experiments:

1) Full access to the network: we assume that the attacker and defender have complete knowledge to the networks. In other words, the actions of the attacker and defender can be directed to every edge in the network.
2) Representative of real-world networks: we assume that given the large number of nodes and edges in the data sets, the data sets are representative of their real-world counterparts. For any conjecture about the network robustness to have a reasonable chance to be valid, this assumption needs to hold true.

C. Step 2: Understanding the properties of the networks

In this step, we will analyze a set of attributes of the network, including the number of nodes, number of edges, average degree, diameter, and average and global clustering coefficient. These concepts will provide the basic structural information on the network.

D. Step 3: Selecting evaluation metrics and implementing algorithms

In this step, we will survey the appropriate metrics to measure the robustness of networks. The metric that we will adopt is the Molloy-Reed criterion [1] [3], which as discussed previously will measure the existence of giant components in the networks.

We will then design an attack algorithm that removes edges in the network and the corresponding defense algorithm. Initially, we will design an algorithm that randomly selects the edges for removal and repair. Subsequently, we will improve our algorithms by incorporating heuristics on the structural information of the network so that the algorithm can more intelligently select edges that are more vulnerable.

E. Analysis

In addition to the Molloy-Reed analysis, we will run two more analyses on the networks, one qualitative and one quantitative. For the qualitative analysis, we will visualize the networks and compare the level of edge connections under each scheme. On the other hand, for the quantitative analysis, we will compare how degree distribution evolves over time under each scheme. Combining them with the Molloy-Reed criterion, we can then draw conclusion regarding the effectiveness of the algorithms.

F. Scalability of the approach

We believe that our algorithm and analysis is applicable to large-scale network. In our data set, the Twitter network has close to 2 million nodes (see Table I). As of 2021, the Twitter network has approximately 200 million users. Although this number is 100 order of magnitudes larger than our data set, our design principles still apply. Nevertheless, attacking and defend larger networks will require much higher computational resources, and experiments will therefore take a longer time.

IV. Experiment Results and analysis

A. Properties of the networks

As outlined in step 2, we have analyzed the fundamental attributes of each of the four networks, and the results are summarized in Table I.

TABLE I
Fundamental properties of the four networks

	Social	P2P	AS	Email
Is the network directed?	Yes	Yes	No	No
Are the edges weighted?	No	No	No	No
Number of nodes	81,306	62,586	10,900	36,692
Number of edges	1,768,149	147,892	31,180	183,831
Avg degree	43	4.73	5.72	10.02
Diameter	7	11	9	11
Avg clustering coef	0.40	0.0027	0.501	0.716
Global clustering coef	0.11	0.0039	0.0386	0.0213

As we can see from the Table I, the selected networks have a high number of nodes and edges, with one network having 1.77 million edges. As a result, we believe that these networks have sufficiently large size to be representative of real-world networks. In addition, as we can see from the degree distribution plot in Figure 1, (due to limited space, we show the degree distribution plot for social network only,

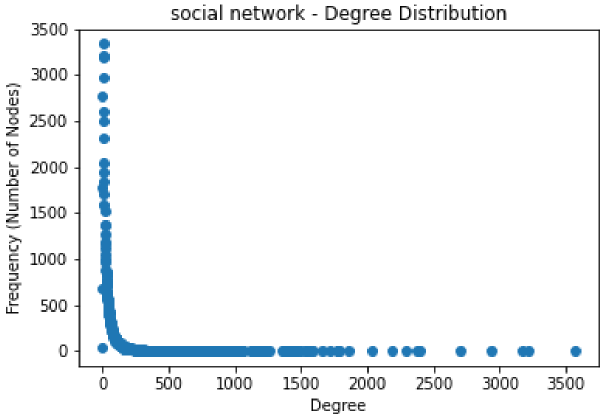

Fig. 1. Degree distribution of the social network data set. Most of the nodes in this network have low degree, while selected individual have degree higher than $3,500$, which may belong to accounts of public figures and/or celebrities.

and we will show further analysis on P2P network in the following sections), the Twitter network has few individuals with higher degree, and common sense suggests that these may belong to public figure account. Therefore, we conclude that the preliminary results make sense in terms of their accuracy in representing a real-world scenario.

B. Description of the algorithms

Having analyzed the basic attributes of the network, we proceed to the algorithm designing phase. The first algorithm we have designed in stochastic in its essence. In this case, we assign a probability p of attack and defense. The attacker will randomly select $p\%$ of edges and remove those edges from the network. In response, the defender will randomly add back the same number of edges to networks. The random algorithm is summarized in Algorithm 1.

The second algorithm will leverage more heuristic in its selection of edges. In this case, the attacker will first rank the nodes with the highest degree and select the top $p\%$ nodes. Subsequently, for each of the edges connecting to the nodes, the edge will be removed with probability p. In response, the defender will select a set of top $p\%$ nodes with the highest degree. For each node in this set, a node is randomly selected from the remaining nodes and an edge is added if it does not exist between the nodes. This is continued until the number of removed edges are added back to the network. The smart algorithm is summarized in Algorithm 2.

C. Experiment setting and simulation results

Note that although we have executed the experiments on all four networks, due to limited space, we will only report the results for P2P network for illustration purpose.

1) Experiment parameters: With the two sets of algorithms ready, we proceed to simulate the response of the networks under the two schemes. The parameters of the experiments are summarized in Table II. Note that we will perform the experiments with three different probability thresholds, in order to observe how the extent of attack and defense affects the network's robustness.

Algorithm 1 Random Attack and Defense

```
Input: p = probability of selection, T = number of steps
Initialize the network (assumed n nodes and m edges)
for t=1 ... T do
    Execute attack (A):
    A1) Randomly sample $p\%$ of the edges;
    A2) Remove the selected $np$ edges from the network;
    Execute defense (D):
    D1) Randomly sample 2 nodes;
    D2) Add an edge between the selected nodes if there is no edge;
    D3) Go to Step D1 until $np$ number of edges are added back to the network.
end
```

Algorithm 2 Smart Attack and Defense

```
Input: p = probability of selection, T = number of steps,
Initialize the network (assumed n nodes and m edges)
for t=1 ... T do
    Execute attack (A):
    A1) Sort the nodes based on its degree and select the $p\%$ of the nodes with the highest degree;
    for node =1 ... np do
        A2) Randomly select $p\%$ of the edges connected to the node;
        A3) Remove the selected edges from the network;
    end
    Execute defense (D):
    D1) Sort the nodes (SN) based on its degree and select the $p\%$ of the nodes with the highest degree;
    for node =1 ... SN do
        D2) Randomly select one node from the remaining nodes;
        D3) Add edge between the two nodes if no edge exists;
        D4) Go to Step D2 until $np$ number of edges are added back to the network.
    end
end
```

TABLE II
PARAMETERS OF THE EXPERIMENTS

Parameter	Description	Value
T	Number of steps to simulate	1,000
P	Probability of selection	1%, 5%, 10%

2) Discussion on Molloy-Reed values (κ): We will first analyze the values of κ. From Figure 2, we compare κ for the P2P networks under the two schemes. As can be observed, both κ's remain above the critical threshold of 2, indicating the existence of a giant component under both the random and smart scheme. However, κ for the random algorithm is about 50% lower than that for the smart algorithm, indicating that the network is relatively more robust under the smart algorithm.

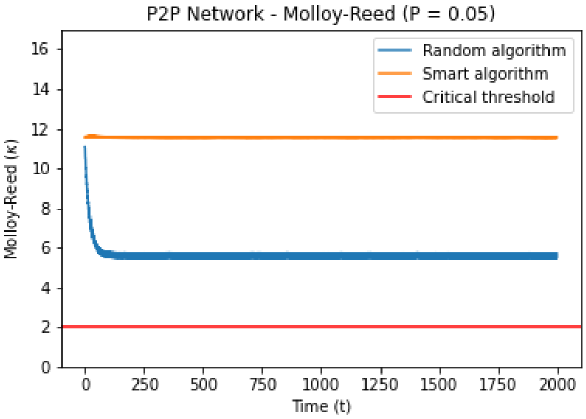

Fig. 2. κ for P2P network under random and smart algorithm are plotted against time. The red line represents the critical threshold value of 2, above which indicates the existence of a giant component. Note that the plot contains 2,000 steps, whereas the parameter $T = 1,000$. This occurs because at every step, one attack and one defense move is executed (i.e., there are 2,000 moves in total), and κ is recorded for each move.

3) Discussion on network visualization: The visualization of the networks also confirms our conclusion above. For ease of visualization, we draw only the highest 1% of the nodes (in terms of degree) and the edges connected to them.

We plotted the initial P2P network in Figure 3 (without any attack and defense), and the network shows a large number of edges connected between the nodes. The final stage of the networks under random and smart algorithms are plotted in Figure 4 and Figure 5, respectively. Figure 4 shows that under the random algorithm, most of the edges in the network have been destroyed. The nodes have thus become isolated, and the network is fragile. However, Figure 5 shows that the network is able to retain most of its edges and protect its robustness.

4) Discussion on degree distribution: We provide further evidence on the strength of the smart algorithm by comparing the network's degree distribution. The degree distribution under the random algorithm is shown in Figure 6. Initially, there is one node with degree of 91; however, as time progresses, there is no longer any node with a high degree and the degree of all the nodes have fallen below 20.

The degree distribution under the smart algorithm is shown in Figure 7. Comparing to the degree distribution under random algorithm at time $t = 500$, one node has degree of 86 and there remains a large number of nodes with degree higher than 20. Therefore, the comparison of degree distribution also confirms our hypothesis that the networks is more robust under the smart algorithm scheme.

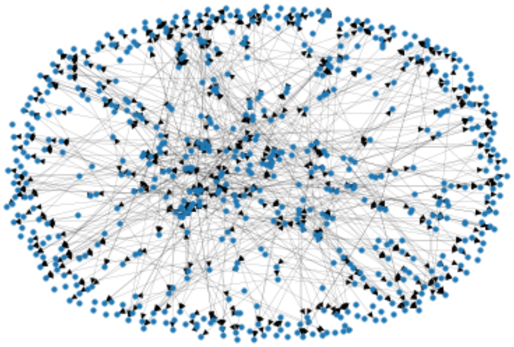

Fig. 3. Visualization of the P2P network at the initial stage of $T = 0$. For ease of visualization, only the top 1% of nodes (measured by degree) are shown in the picture.

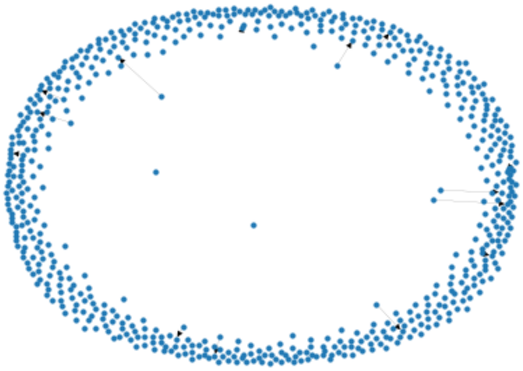

Fig. 4. Visualization of the P2P network at $T = 1,000$. The edges are removed and defended using the random algorithms. Comparing to the initial state, the top ranking nodes have many fewer edges.

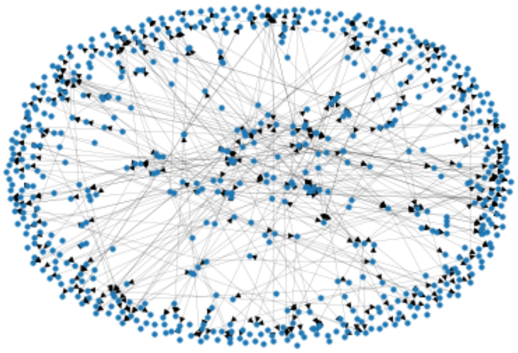

Fig. 5. Visualization of the P2P network at $T = 1,000$. The edges are removed and defended using the smart algorithms. Comparing to the random algorithm, the top ranking nodes are able to retain many of its edges.

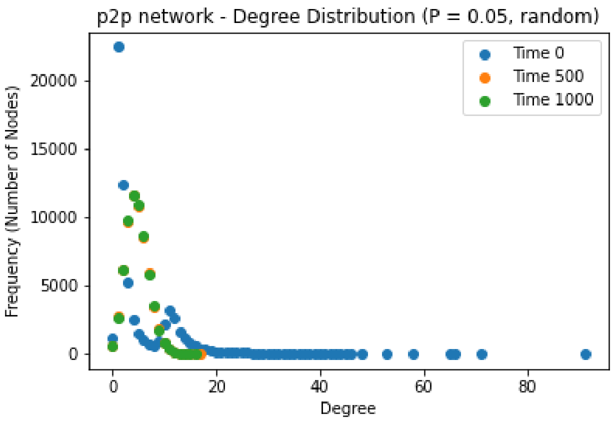

Fig. 6. The degree distribution under random scheme, for P2P network at different times. As can be observed, time is enemy to a network without a strong defense mechanism, as nodes with high degree will be destroyed over time.

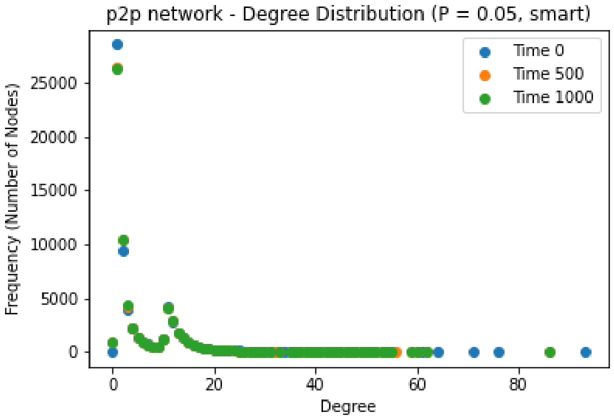

Fig. 7. The degree distribution under smart scheme, for P2P network at different times. Compared to the random scheme, as time progresses, a number of nodes with high degree are still retained.

D. *Significance and limitation of our results*

The experiment results provide insights into the strategy of attacking and defending a network, and its significance is derived from its potential to further analyze other real-world networks. For instance, for our paper presentation, we were exposed to the concept of cascading failures and interdependent cyber-physical network. Our approach may potentially be extended to analyze how to prevent or delay the occurrence of cascading failures in interdependent networks.

On the other hand, we recognize certain limitations of our results. First, in the real-world, the attacker and defender may adopt different strategies. Second, in our experiment setting, the same number of edges are destroyed and repaired, and this may not be the case in the real-world. Furthermore, we did not take into consideration the computational resources involved. For instance, in the case of one nation attacking another nation's internet network, the availability of computational resources can be a major determinant of the outcome. Overall, many questions remain unanswered and could serve as a starting point for future research direction.

V. Conclusion and Short-term Plans

This project is motivated to investigate real-world networks and their robustness to withstand attacks. For milestone 1 and 2, we have selected the network data sets, analyzed the network's fundamental attributes, designed two algorithms, and executed the experiments. Although we only demonstrated the P2P networks here, our experimental results show that across the four selected networks, the networks is indeed more robust under smart algorithm than random algorithm. This makes intuitive sense, as the smart algorithm is intentionally fixing the network by connecting the nodes with low degree to the central "hubs" in the networks.

A. *Contribution of group members*

Each team member has contributed equally to the project. In the initial data set selection and subsequent analysis, each member is responsible for two networks. For the algorithm implementation phase, Runqi (Tony) Huang designed the random and smart attack algorithm, while Vikas Kashyap designed the random and smart defense algorithm. The algorithms are then merged into an environment for the experiments.

B. *Next step for milestone 3*

For the next milestone, we plan on improving the smart attack and defense algorithm. For example, we notice that under our current smart attack algorithm in Algorithm 2, step A2 is still stochastic. Alternatively, we could rank the opposing node by betweenness, and remove edges that connect to nodes with high betweenness. Similarly measure can also be adopted in the smart defense to make the network more resilient to outside attack. We hypothesize that this scheme will further improve the network robustness, comparing to our current smart algorithm. Please stay tuned.

References

[1] A.-L. Barabasi. Network Science. Cambridge University Press. 2016. http://barabasi.com/f/619.pdf

[2] J. Leskovec and A. Krevl. SNAP Datasets: Stanford Large Network Dataset Collection. 2014. https://snap.stanford.edu/data/

[3] M. Molloy and B. Reed. A criticial point for random graphs with a given degree sequence. Random Structures and Algorithms, 6: 161, 1995.

[4] H. Dai, H. Li, T. Tian, X. Huang, L. Wang, J. Zhu, and L. Song. Adversarial attack on graph structured data. arXiv preprint arXiv:1806.02371, 2018.

[5] L. Sun and Y. Dou and C. Yang and K. Zhang and J. Wang and P. S. Yu and L. He and B. Li. Adversarial Attack and Defense on Graph Data: A Survey. IEEE. 2022.

[6] W. Ellens and R. E. Kooij. Graph measures and network robustness. arXiv. 2013.

[7] J. Wu, Y. Tan, H Deng, Y. Li, B. Liu, X. Lv. Spectral Measure of Robustness in Complex Networks. arXiv. 2008.

[8] Gu, Y., Li, C. The robustness of urban rail transit network based on complex network theory. 3rd International Conference on Materials Engineering, Manufacturing Technology and Control. Atlantis Press. 2016.