

Optimizing the Study of Cascading Failures in Synthetic Electrical Grids

Thesis Submitted in Partial Fulfillment
of the Requirements
of the Jay and Jeanie Schottenstein Honors Program

Yeshiva College

Yeshiva University

September 2020

Daniel Feldan

Mentor: Professor Sergey Buldyrev, Physics

I: Introduction:

Power grids can be modeled as a network of nodes and the electrical lines that connect them. Removal of a single node or a group of nodes, either by natural systemic failure or human attack, causes a redistribution of power in the network which can overload any number of nodes^{1,2,3,4}. When these nodes overload, they fail and cause a further redistribution of power within the network. This process can repeat and lead to massive cascading failures in the network, causing major blackouts or a complete failure of the network^{2,3}.

However, while these blackouts can be devastating, it is found that during the initial stages of a cascading failure that leads to a blackout there is a latent period where the damage is mostly localized and the decrease in yield of the system (the fraction of demand satisfied after the cascade) is insignificant¹. This is important because if a real-world cascading failure begins in a network, a fast simulation of the same failure can be executed during this latent period and be used to help prevent major losses in the network's yield. The goal of this project is to produce accurate and rapid simulations of cascading failures in order to study how they behave and the properties of the electrical networks that cause them.

Many people studying failures in power grids use a direct current power flow model^{1,3}. For the purposes of our research we will be utilizing a similar DC power flow model which is equivalent to the flows in a resistor network^{1,3}. This allows us to utilize Kirchhoff's Equations to solve the flow of the network at any point during a failure.

However, in actuality it is very difficult to study the properties of electrical networks and the cascading failures that occur within them because there is very little publicly available data on electrical networks^{6,7,8}. This is due to the fact that by releasing the structure of an electrical grid, the vulnerabilities of the grid would also be revealed^{6,7,8}. Therefore, this project will focus not on real world data, but will utilize a data set produced by researchers at Columbia University that resembles a synthetic electrical grid^{6,7,8,9}.

The rest of the paper is organized as follows. In Section II we describe the synthetic power grid and perform some simple data exploration on the grid itself. Then in Section III we will discuss our initial goals and methods for creating fast simulations of cascading failures in the power grid. Finally, in Section IV we will discuss and summarize our results.

II: Model and Data Exploration:

A. Definitions:

An electric network is composed of two elements, nodes, denoted as N , and edges connecting nodes, denoted as E . Each edge between nodes i and j , $e_{i,j}$, has a resistance of $r_{i,j}$. With regard to our electrical grid, there are three types of nodes: generators (g), loads (l), and transmitters (t). The total number of nodes in the network is as follows: $N = g + l + t$. In our dataset, there are a total of 14,430 nodes: 3,095 generators, 3,058 loads, and 8,277 transmitters. Each generator node supplies current to the network ($I_i^+ > 0$), each load node uses the energy supplied by the generator nodes ($I_i^- < 0$), and each transmitter node helps facilitate the traversal of electricity throughout the network but does not

supply or use a significant amount of energy ($I_i^0 = 0$). Additionally, The law of charge conservation states that the net amount of charge flowing through a node should be conserved, therefore, the amount of charge that a node consumes or produces should be equivalent to the amount of charge flowing in and out of the node through its edges ($\sum I_i^+ = \sum I_i^-$).

Furthermore, edges are categorized as g_g, l_l, t_t, g_l, g_t, or l_t. For each of these categorizations, if a category contains a g it means that one of the nodes in the edge is a generator. It does not imply which node in the edge it is, only that it is part of the edge. The same applies for loads and transmitters respectively. That means if there is an edge between a generator and a transmitter it would be categorized as g_t or t_g, as both categorizations are equivalent. In total, there are 18,884 edges in the network. A brief breakdown of the number of each type of edge is listed in Table 1 below.

	Total	g_g	l_l	t_t	g_l	g_t	l_t
Count	18884	1620	1297	9844	1234	1795	3094

Table 1: The number of each edge type in the network

B. Model:

The network is specified in 3 separate csv files – Lines, Loads, and Generators. Lines contains a list of edges in the network and the resistance of each edge. The Generators file list all of the nodes in the network, as well as the amount of current they supply. The Loads file lists all of the nodes in the network as well as the amount of current they consume. It is important to note that while all the values in Loads are recorded as positive the values translate to a net loss amount of current passing through that node in the network.

C. Data Exploration:

Again, the data is made up of 14,430 nodes and 18,884 edges. In Fig. 1 and Fig. 2 we list the percent breakdown of generators, loads, and transmitters in both the nodes and edges.

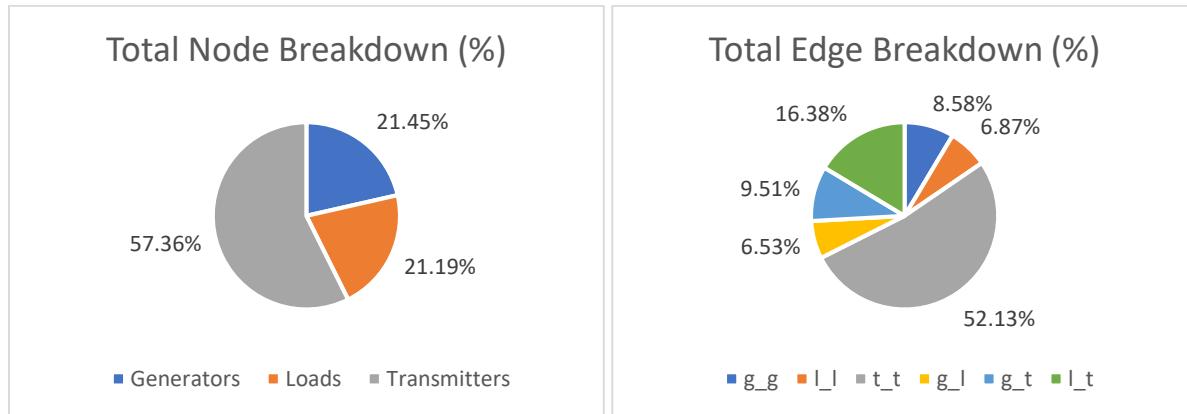


Figure 1 (left) and Figure 2 (right): Figure 1 shows the percentage breakdown of Generators, Loads, and Transmitters in the network. Figure 2 show the percent breakdown of each type of edge in the network.

From a quick glance it is easy to notice that transmitters make up a large part of the network both in terms of nodes and edges. They consist of over half the number of nodes and over 75% of the edges. Furthermore, the difference between the amount of load and generator nodes is very small. However, in regard to the edges, the amount of edges that have a load (30%) significantly exceeds the number of edges that have a generator (24%). This is due to the fact that the percentage of edges between a load and transmitter is much larger than the percentage of edges between a generator and transmitter.

When exploring the distribution of the resistances of the edges, the edges follow a normal distribution in the center with power law tails. The power law tails are manifested by the approximately straight-line behavior of the left and the right tails on a double logarithmic plot (Fig. 4)

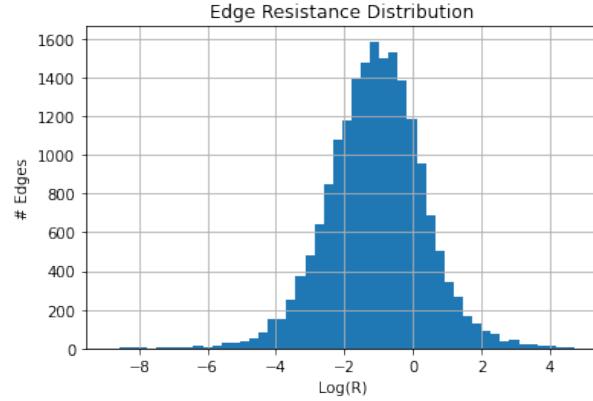


Figure 3: A histogram of the edges in the network binned by the natural logarithm of their resistance

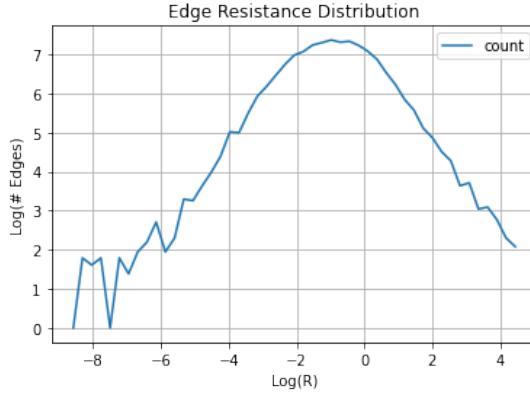
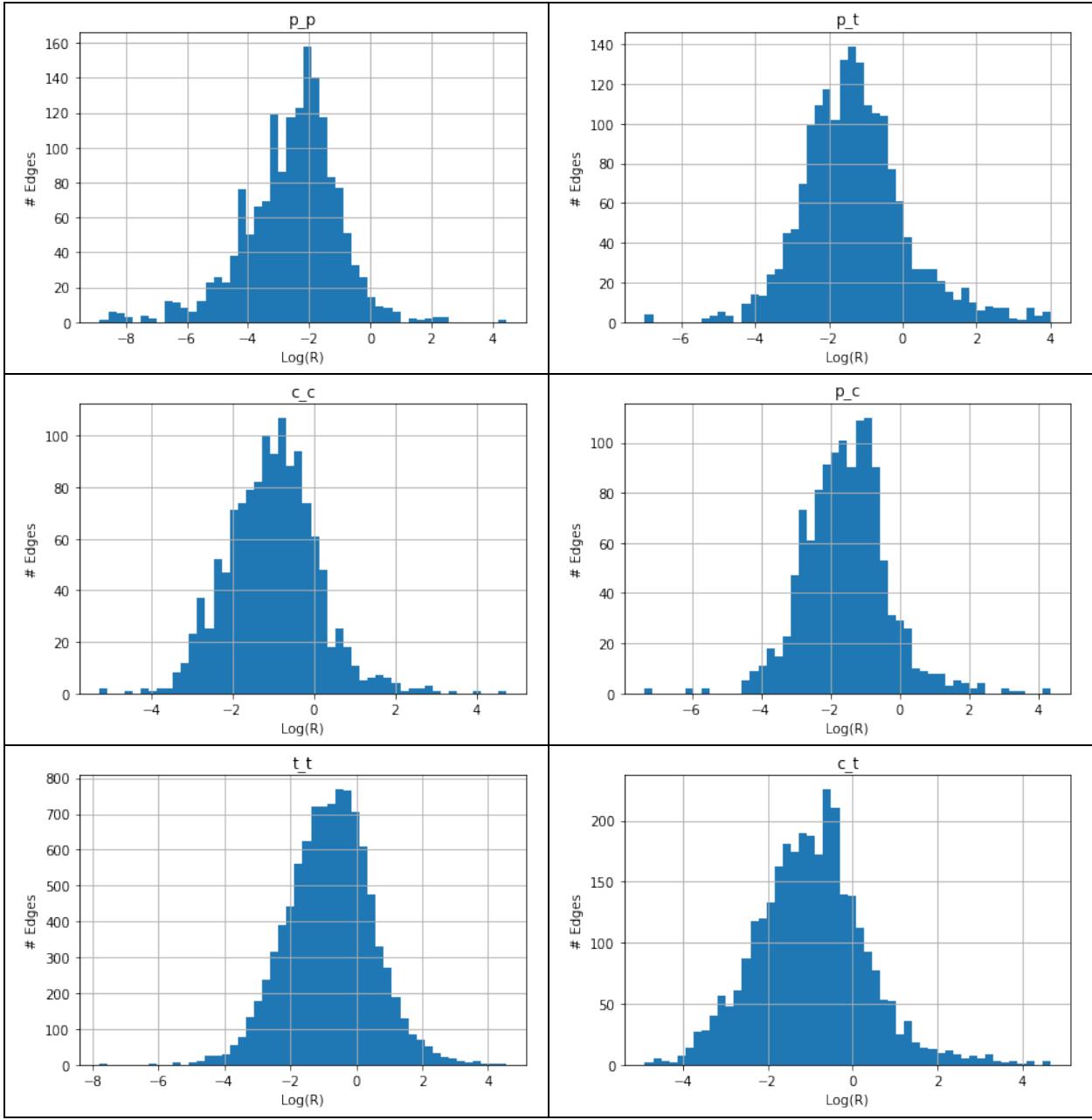


Figure 4: The natural logarithm of the size of each bin from Figure 3 plotted against the natural logarithm of the resistance of that bin

When only analyzing edges between specific types of nodes, a similar normal distribution in the center follows (Figures 5-10). The distribution might be shifted to the left or right slightly, but this is to be expected. A complete statistical breakdown of each of the histograms is shown in Table 2 below.



Figures 5-10: Histograms of each edge type in the network binned by the logarithm of their resistance

Edges	Resistance		
	Mean	Standard Deviation	Median
All Edges	-1.09	1.41	-1.06
g_g	-2.57	1.55	-2.35
g_t	-1.29	1.42	-1.38
g_l	-1.55	1.22	-1.57
l_l	-1.01	1.12	-1.01
l_t	-0.97	1.30	-0.97
t_t	-0.79	1.30	-0.75

Table 2: Statistical data on the distribution of each edge type in the network

III: Goals and Research:

We already have a program which simulates cascading failures in a network. The program works as follows: It incites failure in a single edge, which then causes a redistribution of power within the network. Using Kirchhoff's Equations, we solve for the amount of current flowing through each edge (a process referred to as solving the network) and find which lines have a current capacity above their maximum capacity. We remove those lines causing another redistribution of power within the network. This process is repeated until the network converges onto a state of equilibrium³.

We can then apply this program to our synthetic dataset to study the effects of a cascading blackout. However, while the current program is functional, it is far too slow to be used in a real time setting because it has to solve the network at every iteration. Therefore, what we aimed to achieve was to speed up the program, specifically the stage of the program that solves the network after the power within has been redistributed.

We sped up the solving of the network by reducing the network to a simpler form, solving this simpler network, and then using the partially solved network to quickly solve the original network. In order to reduce the network into a simpler version, we grouped specific nodes or edges together (reducing multiple nodes or edges into a single node or edge) in order to reduce the number of nodes and edges within the network. This means that when the program needs to solve the network again, the network it is solving is a much smaller version than the original network and requires less computation.

We grouped many types of nodes with the following criteria:

A. Low resistance

Two or more nodes connected by a line with very little resistance can be reduced to a single node. This is due to the fact that since the resistance between the nodes is very low, the total flow of energy between the two nodes is equivalent to the total amount of energy flowing into and out of these nodes. Therefore, the two nodes can be considered a single node whose inputs and outputs of electricity are the same as each component nodes that it represents.

Since there was no clear minimum threshold of resistance to choose, we used three possible options: $e^{-6.7}$, e^{-5} , $e^{-2.5}$. $e^{-6.7}$ was chosen as it is the end of the left tail in the distribution of resistances. The other two thresholds were chosen arbitrarily.

B. Dangling Edges

At the fringes of the network, there are singly connected components (SCC) which do not have generator or load nodes in them. These grouping of nodes are a byproduct of either the way real-world electrical grids are constructed, or a cascading failure knocking out the consumers and loads in the SCC. Since there are no nodes which generate or consume power in these SCCs, we know that the current between them is 0. Therefore, we do not need to solve for their currents using Kirchhoff's Equations and can remove them from the dataset.

C. Multi-Edges

Some pairs of nodes have multiples edges that connect them. These types of edges are referred to as multi-edges. For nodes which have multi-edges between them, we combined these edges together to create a single edge with resistance R_n between the two nodes. This is due to the fact the total

resistance between two nodes is independent of the number of edges. R_n has a resistance which is the sum of the reciprocal of the current of all the multi-edges:

$$R_n = \frac{1}{\sum \frac{1}{R_i}}$$

D. Strongly Connected Components

We planned on reducing large strongly connected components into single nodes. Because SCCs have defined current intake and outputs it would theoretically be easier to manage and recompute their current flow with each iteration. Theoretically, it makes more sense to overload an entire SCC by analyzing its input and output and therefore reduce the amount of iterations the program needs to take considerably. However, due to collaboration challenges from COVID-19 and the semester moving online, we were unable to accomplish this goal.

IV: Results:

After simplifying the network, we reduced the network from having 18,884 edges and 14,430 nodes to having the number of edges and nodes recorded in Table 3.

	# Nodes	# Edges
Unsimplified	14430 (-0%)	18884 (-0%)
$e^{-6.7}$	14401 (-0.002%)	16755 (-11.274%)
e^{-5}	14287 (-0.100%)	16633 (-11.920%)
$e^{-2.5}$	12185 (-15.579%)	14271 (-22.428%)

Table 3: The number of nodes and edges in the network after combining nodes with different thresholds of minimum resistance

It is important to note that for the lower two thresholds, the change in nodes is much smaller than the change in edges. This is due to the fact that as the threshold changes, only the number of nodes combined due to criteria A changes. The number of edges combined due to criteria B and C remain constant at 2,110 edges.

In Table 4 we list the number of nodes combined for each of the three thresholds.

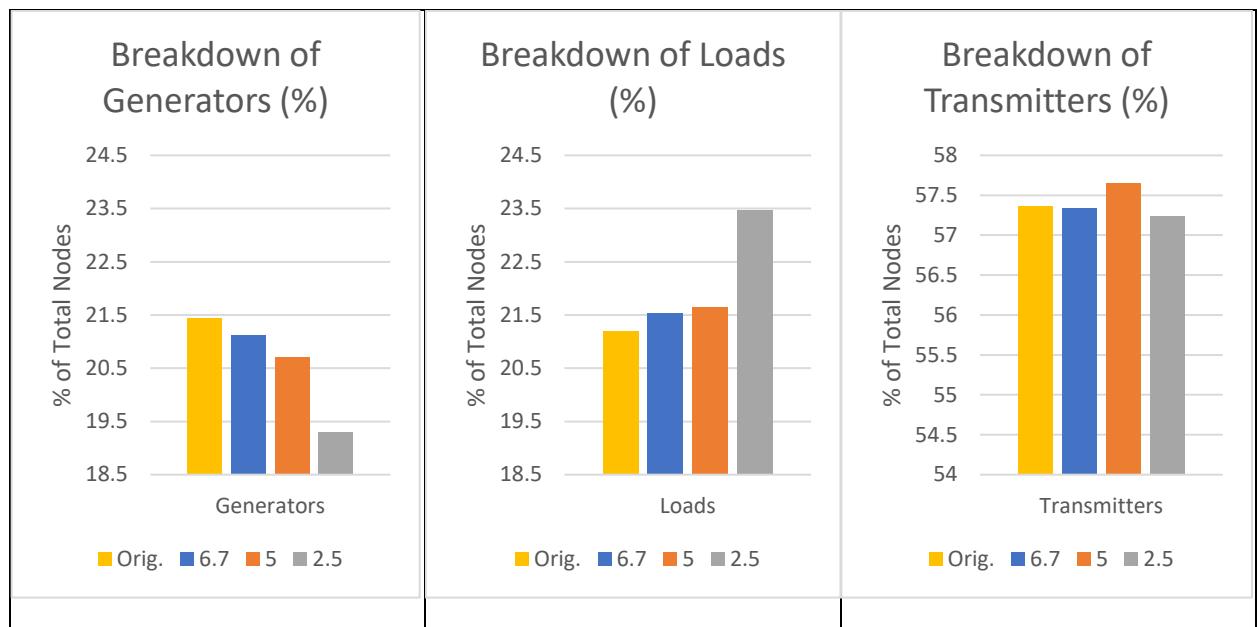
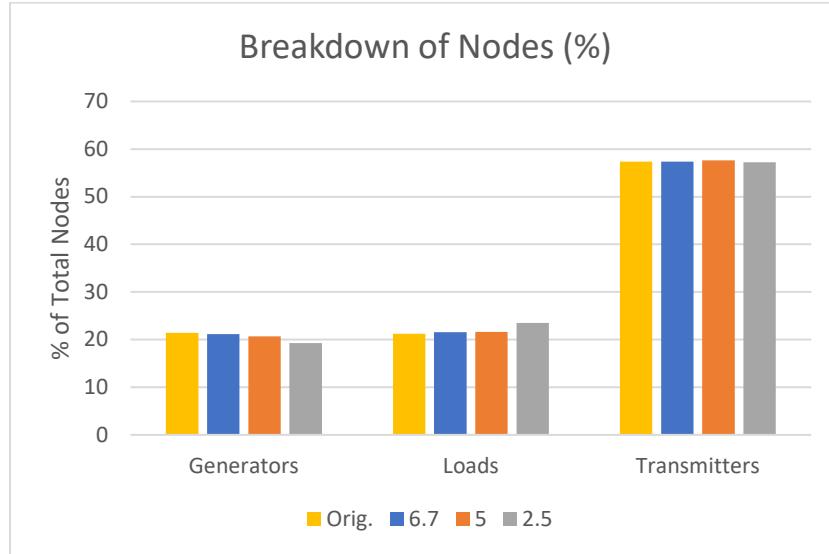
	Nodes Combined
$e^{-6.7}$	29
e^{-5}	143
$e^{-2.5}$	2245

Table 4: The number of nodes combined for each threshold of minimum resistance

As the threshold increases, the number of nodes that are combined drastically increases. This is in line with our data, as the distribution of the natural logarithm of the edge resistance has power law tails. As the threshold increases, the number of nodes should increase by an exponential amount, which it does.

Below we show the differences in distributions between each of the node types for the original graph and each threshold. In all four cases the percentage of overall transmitter nodes stays roughly the same. While the percentage of generators and consumers are very close before any processing is done to the graph, the percentage of generators decrease by a substantial amount while the number of consumers increases by a similar amount.

Figure 11: The difference of percentages of each node type in the network as we change the threshold of minimum resistance.



Figures 12-14(L-R): The difference of percentages of each node type in the network as we change the threshold of minimum resistance, but only for only a single node type

We can explain this discrepancy by analyzing the percentage breakdown of the combined edges, as shown in Fig. 15. It is important to note for the threshold of $e^{-6.7}$, while a very large percentage of the edges that were combined were g_g edges the total number of edges is actually very small (<27 edges out of 18,000)

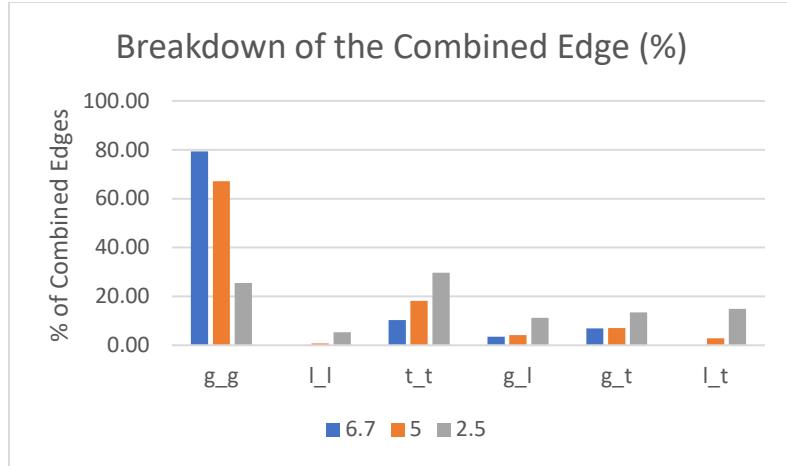


Figure 15: The percent of each edge type that was combined for different thresholds of minimum resistance

From these graphs we see that a large portion of the combined edges are between two generator nodes, which is in line with the fact that the overall percentage of generators decreased in the network when simplified. Furthermore, the graph shows that very few of the combined edges contained a load. This implies that that the number of loads did not actually increase. However, since the number of generators decreased, the relative percentage of the remaining node types, i.e loads and transmitters, increased. But since a large portion of the combined edges also had a transmitter, this increase in percentage of the loads and transmitters is offset, explaining why the percentage of transmitters stayed relatively constant.

Lastly, one can see that while for minimum resistance thresholds of $e^{-6.7}$ and e^{-5} , the number of edges combined are largely edges between two generator nodes. However, with a threshold of $e^{-2.5}$, the difference in types of edges that were combined seem to be much less jarring.

An interesting point is that since the total amount of production of the combined node is the production of each node that was combined, if a transmitter would be combined with a producer or consumer, the new node would be a producer or consumer respectively. Therefore, the number of generators or producers could actually increase in the network. However, that did not happen, as a vast majority of the edges that were combined were between nodes of the same type.

After simplifying the network, we timed how long it took to solve the network ten times and found the average speed of each solution. We list our results below in Table 5.

	Time (s)	% Speedup
Un-simplified	256.406	N/A
$e^{-6.7}$	166.781	34.954%
e^{-5}	134.063	47.714%
$e^{-2.5}$	50.343	80.3366%

Table 5: The amount of time it takes to solve the network when processed with different thresholds of minimum resistance.

These results look very promising. Even for the smallest threshold of minimum resistance, we achieved a speedup of almost 35%. This large speedup is most likely due to the large number of edges combined from criteria B and C, but even so it is a large speedup to be gained.

V: Conclusion:

In this paper, we analyzed the synthetic power grid dataset created by researchers at Columbia University⁹. We showed that the resistances of this data set follow a normal distribution in the center with power law tails. Furthermore, we analyzed the breakdown of node and edge types in the network.

When we attempted to solve for the currents of each line in the synthetic network as is, the network converged too slowly to be used in an applicable setting. To alleviate this issue, we simplified the network before solving it by combining nodes and edges together based on specific criteria. These criteria included: combining multi-edges into single edges, removing dangling edges, and combining pairs of nodes with low edge resistance between them. After processing the network, we reduced the amount of time it takes to solve the network by a considerable amount. Furthermore, we varied the minimum resistance threshold needed to combine nodes based on the third criteria and studied which threshold is best to use.

Bibliography:

1. Y. Kornbluth, G. Cwilich, and S. V. Buldyrev “Distribution of blackouts in the power grid and the Motter and Lai model”, December 2019
2. Y. Kornbluth, G. Barach, Y. Tuchman, B. Kadish, G. Cwilich, and S. V. Buldyrev “Network overload due to massive attacks”, in *Phys. Rev. E* vol. 97 Iss. 5m May 2018
3. Spiewak, R., Soltan, S., Forman, Y., Buldyrev, S., & Zussman, G. (2018). A study of cascading failures in real and synthetic power grid topologies. *Network Science*, 6(4), 448-468. doi:10.1017/nws.2018.14
4. B. A. Carreras, I. Dobson, Hui Ren, “Long-Term Effect of the n-1 Criterion on Cascading Line Outages in an Evolving Power Transmission Grid”, *IEEE Systems Journal*, vol. 23, no. 3, August 2008
5. B. A. Carreras, V. E. Lynch, I. Dobson, and D. E. Newman, “Complex Dynamics of Blackouts in Power Transmission Systems,” *Chaos Journal*, September 2004
6. S. Soltan, A. Loh, and G. Zussman, “A learning-based method for generating synthetic power grids,” *IEEE Systems Journal*, vol. 13, no. 1, pp. 625–634, Mar. 2019
7. S. Soltan and G. Zussman, “Generation of synthetic spatially embedded power grid networks,” in *Proc. IEEE PES-GM’16*, July 2016.
8. S. Soltan and G. Zussman, “Generation of synthetic spatially embedded power grid networks,” in *arXiv:1508.04447 [cs.SY]*, Aug. 2015
9. S. Soltan, A. Loh, and G. Zussman, “Columbia University Synthetic Power Grid with Geographical Coordinates,” <https://doi.org/10.17041/drpl/1471682>, Jan. 2018
10. B. A. Carreras, I. Dobson, and D. E. Newman, “North American blackout time series statistics and implications for blackout risk”, *IEEE Systems Journal*, vol. 31, no. 6, pp. 4406-4414, November 2016