Complex Networks Project Report

Modelling power grid network failures

Mateusz Iwo Dubaniowski*[[1]](#footnote-1)*

PhD Researcher

Future Resilient Systems – Singapore-ETH Centre

9 May 2016

The aim of this document is to present the project completed as a part of the Complex Networks (CN) course. The project forming a part of the course incorporated the complex networks simulation framework described throughout the course. We used Python numpy’s library implementation of the complex networks concepts to develop and visualize the solution. Furthermore, we outline the solution developed throughout the course of the project. Finally, we provide some insights into results, and discuss possible future work that could be completed on the project.

## The Problem

Electrical grids consist of various elements such as power stations, transformation stations, substations, and transmission lines. These elements of the power grid can potentially be an object of a hostile attact, or can suffer from a critical failure due to various factors. These factors include atmospheric conditions, external power gird network influences, or systemic failures. Such failures can have significant consequences on the overall power grid of a country. An example of such failure would be the blackout around Orchard Road arean in Singapore on the 13th of December 2015. This failure had a negative impact on the economy of the area, forcing several shops to close early. Through development of a simulation of such failures, ande identifying the most critical aspects of the power grid network, we will be able to devise a contingency plan for such failures as well as attempt to focus on mitigating these.

## The Task

We aim to develop a simulation of response of a power grid network to failures or downgrades in the system. This will include devising a network of electrical supply grid elements. Such network will be developed with the use of numpy library. We will emulate transmission lines by assigning a capacity to each transmission line. This would allow us to see how impact of a node outage propagates to other elements of the network. And, subsequently, what impact it has on the overall system.

## The Solution

To implement the network, and solve the issue we will implement the following strategy.

We use the available power grid network of the US[[2]](#footnote-2). We assign a capacity to each edge of the network to represent its capacity to transport electricity. Similarly, we assign a capacity to each node (vertice) representing this node’s production or consumption of electricity. To analyze netowork’s response to failueres or unfavourable events, we then remove nodes from the network and/or decrease their capacities. Finally, we observe how it impacts the system. Particularly, we look at whether the network could be load balanced to ensure sufficient energy supply to each node after the failure event. The process described allows us to infer how the power grid network behaves under such unfavourable conditions. In Table 1, we show a summary of this process. In subsequent sections, we focus on presenting the details of the method of implementation of this solution, and what results we present.

Table 1: High level overview of the solution. Steps performed to complete the simulation.

|  |
| --- |
| 1. Set up a network based on the US power grid topology |
| 1. Populate the network with attributes for edges and vertices    * Capacities of vertices- energy production (positive) or consumption (negative)    * Capacities of edges - transmission lines' bandwidth |
| 1. Load balance the network to ensure all capacities of vertices are non-negative |
| 1. Remove a node or an edge (or decrease capacity) - introduce a failure |
| 1. Attempt load balancing again to see if the failure is critical |
| 1. Present results and statistics about the network |

## Method

We use the US power grid network topology compiled by Watts and Strogatz throughout the project. Although the topology is reasonably extensive, it contains 6594 edges and almost 4941 vertices, it does not include any numerical values signifying electricity being produced or consumed by each of these vertices. Similarly, it lacks bandwidth capacities for all edges contained in the topology. Consequently, before attempting to balance the network, or test if it is balancable, we had to populate the network accordingly. Only then, we could perform meaningful operations on the network, such as balancing or removing nodes and observing the response.

### Populating the network

We decided to populate the network with a random number generator in line with a particular pattern, which in a general framework could be easily adjusted accordingly to fit the particular case. Since, we do not know the granularity and the level of detail and of coverage of each node of the used topology, we used the following parameters for the generation of supply and demand of electricty. The parameters are presented in Table 2. Similarly, in case of edges, we ensured that each can carry twice the average load of an edge, where average load is defined as total produced electricty divided by total number of edges. However, we also ensure that each node can be theoretically discharged through the neighbouring edges, so if the neighbouring edges combined cannot satisfy a node, their capacity is increased.

Table 2: Parameters of network nodes population model.

|  |  |
| --- | --- |
| **Parameter** | **Values** |
| Fraction of nodes that are producers | 1/30 |
| Producers capacity | 40\*U(100, 500) |
| Consumers capacity | 5\*U(-100, 5) |

### Balancing the network

To ensure that all nodes can be satisfied, we perform network load balancing. If the network can be balanced after a failure – removing a node, that means it can survive the failure. To perform load balancing, we iteratively try to average all neighboring nodes until all of them are satisfied. In this context, a node is considered satisfied, when its current capacity i.e. net energy available to that node is greater than or equal to 0. However, while we perform balancing, the energy exchanged between neighboring nodes cannot exceed the connecting edge’s capacity. If there is no change at all in subsequent steps or the maximum change is very small, and not all nodes are satisfied yet, we determine that the network cannot be balanced.

Having completed initial balancing, we randomly remove a node from the network, and then attempt the balancing again. If the balancing can be achieved that means the network can survive the failure, otherwise, it is prone to failure when the given node is removed.

## Results

After completing the balancing, we compile a set of results regarding the network. Primarily, we state whether removing a particular node in a the power grid network would cause that network to crash. Moreover, we present the foll

## Summary and conclusions

## Future work

1. iwo.dubaniowski@frs.ethz.ch [↑](#footnote-ref-1)
2. US power grid topology from D. J. Watts and S. H. Strogatz, Nature 393, 440-442 (1998) [↑](#footnote-ref-2)