LA-UR-04-7649

Approved for public release;
distribution is unlimited.

Title:

Critical Energy Infrastructure
Contingency Screening Heuristics
Status Report

Author(s):

M. P. Blue
B. W. Bush

Submitted to:

Jon MacLaren
U.S. Department of Homeland Security

Los Alamos NATIONAL LABORATORY

Critical Energy Infrastructure Contingency Screening Heuristics Status Report

M. Blue
Systems Engineering and Integration Group
Decision Applications Division
Los Alamos National Laboratory
Los Alamos, NM 87545

B. Bush
Energy & Environmental Analysis Group
Decision Applications Division
Los Alamos National Laboratory
Los Alamos, NM 87545

26 May 2004

Contents

1 Overview **2**

2 Element-Ranking Approach:
Critical Components in Energy Supply Networks **4**

2.1 Introduction 4
2.2 Background 4
2.3 Shortest Paths 5
2.4 Edge Betweenness 6
2.5 Shortest Cycle 7
2.6 Examples 7
2.7 Ranking 16

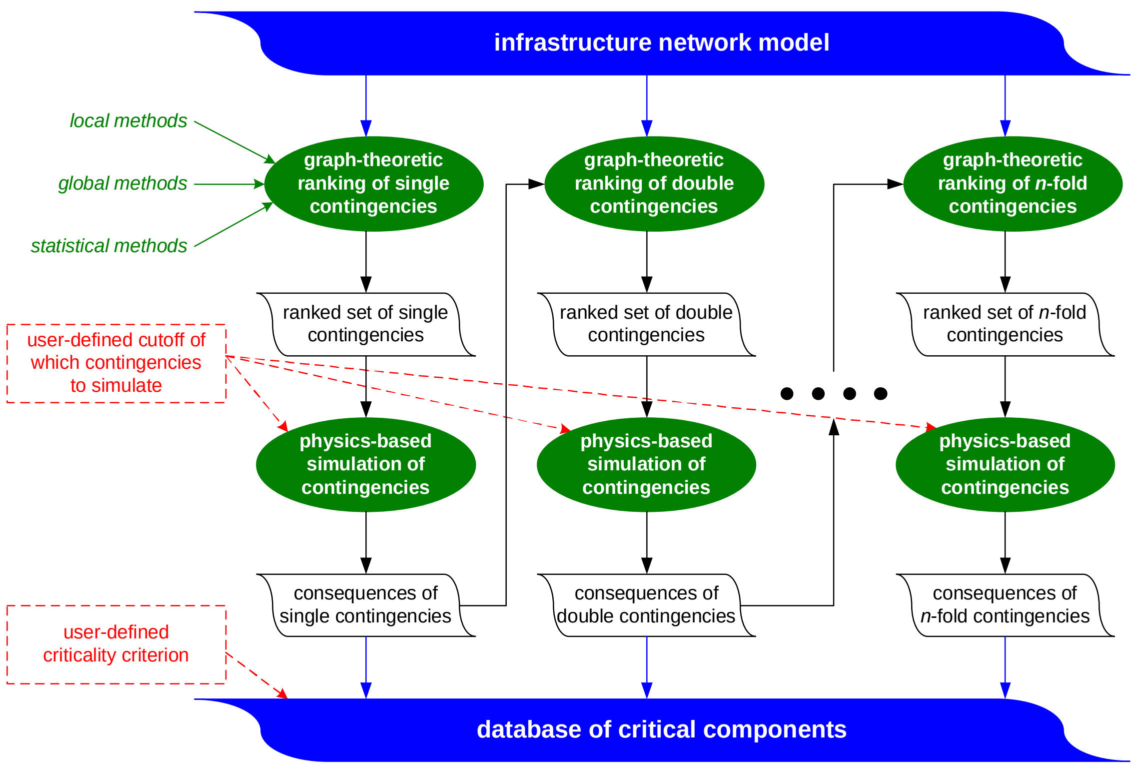
3 Graph-Ranking Approach:
Robustness and Reliability Metrics
for Energy Transmission Networks **17**

3.1 Introduction 17
3.1.1 Definitions 18
3.2 Terminology 22
3.2.1 Graphs & Networks 22
3.2.2 Infrastructure Physics 23
3.3 Connectivity Metrics 24
3.3.1 Stability and Independence numbers 25
3.3.2 Strength 25
3.4 Flow Metrics 25
3.4.1 Performance Index 25
3.4.2 Residual Capacity 26
3.4.3 Minimum Cut 28
3.4.4 Area of Influence 39
3.4.5 Connected Partitions 39
3.5 Temporal Metrics 40

Chapter 1

Overview

This report summarizes the status of our efforts to develop heuristics for identifying the contingencies with the greatest impact in critical energy infrastructure networks. The goal of this work is to rapidly rank prospective contingency sets using graph-theoretic methods so that we can focus computation-intensive physics simulations of infrastructures on the most important contingencies. The flow chart below illustrates the procedure we envision for populating a database of critical components by successively developing ranked lists of contingencies involving the outage of one, two, three, etc. infrastructure elements: at each stage of the process the contingencies are initially ranked using local, global, or statistical graph-theoretic methods and then a subset of these is simulated to determine their consequences; the information about the severity of already-computed contingencies can be used to construct further sets of contingencies.



The following chapters in this report discuss complementary element-based and graph-based approaches to ranking contingencies for prospective simulation.

Element-ranking approaches are local, constructive techniques that assign a score to each element (node or link) in the graph: the elements are ranked higher on the list of contingencies based on their score. Single-element contingencies are simply sorted by score of their outage elements; multiple-element contingencies are sorted by the sum of their outage elements' scores. Alternatively, element-ranking approaches can compute combined scores for small groups of elements, eliminating the need to sum the scores of individual elements.

Graph-ranking approaches are global, evaluative techniques that assign a score to a whole subgraph (the original graph minus the subset of elements in the contingency). These have the drawbacks that they do not naturally generate lists of prospective contingencies (because there are too many possible contingencies) and are generally slower than element based methods (because they involve algorithms that compute functions on a whole graph), but they allow ranking of mixed sets of contingencies (i.e., single-, double-, and triple-element or arbitrary combinations of node and link outages) simultaneously.

Statistical approaches use consequence information from past contingency simulation runs to predict the consequences for contingencies that have not yet been evaluated using physics-based simulation. This allows us to build on the knowledge base of previous simulations to identify new contingencies that would be highly ranked.

Continued future work on heuristics for contingency screening will focus on applying the methods outlined here to real-world networks and evaluating the practical effectiveness of competing approaches.

Chapter 2

Element-Ranking Approach: Critical Components in Energy Supply Networks

2.1 Introduction

Let $G = (V, E)$ be a graph, where V represents the set of vertices and E the set of edges. The goal in this chapter is to develop a ranking of the power set (set of all subsets) of $E \bigcup V$. That is, we wish to find the most critical subsets of nodes and edges. The ranking should take into account all subsets whose removal from the graph results in a reduction of the maximum flow from the super-source (a node representing all energy suppliers) to the super-sink (a node representing all consumers), with greater loss of flow resulting in a higher ranking. In addition we would like to consider the number of graph elements (as it is easier for a terrorist to attack fewer targets), giving a higher ranking to smaller subsets. Also vulnerability of edges and nodes to attack may be considered.

We would also like to develop a "real" or approximate algorithm for finding such critical sets particularly in large networks. This paper provides a ranking for single edges, and pairs of edges which is easily extrapolated to include single nodes and pairs of nodes. The method is easily extendable to larger sets of edges (or nodes) , however, ranking the entire power set of E, V and $E \bigcup V$ proves to be a quite difficult task.

2.2 Background

Let the graph $G = (V, E)$ represent a power supply/demand network. Let d_i represent the demand at vertex i. Let D denote the total demand of the network; $D_G = \sum_{i \in V} d_i$. Given a set S, denote the power set of S by $\wp(S)$. Consider the function $\psi : \wp(E) \to \Re^+$ given by:

$$\psi(\Gamma) = D_G - D_{G/\Gamma} \tag{2.1}$$

In Equation 2.1, G/Γ is the graph obtained by removing the set of edges Γ from the graph G.

Definition 2.2.1 *Any set of edges whose removal from the network results in a loss of total flow through the network is called a flow-reducing set of edges. (Equivalent definition for sets of nodes or sets consisting of both nodes and edges.)*

Definition 2.2.2 *Two edges* e_1 *and* e_2 *in a flow network* $G = (V, E)$ *are called additive if* $\psi(\{e_1, e_2\}) = \psi(\{e_1\}) + \psi(\{e_2\})$.

Definition 2.2.3 *Two edges* e_1 *and* e_2 *in a flow network* $G = (V, E)$ *are called sub-additive if* $\psi(\{e_1, e_2\}) < \psi(\{e_1\}) + \psi(\{e_2\})$.

Definition 2.2.4 *Two edges* e_1 *and* e_2 *in a flow network* $G = (V, E)$ *are called super-additive if* $\psi(\{e_1, e_2\}) > \psi(\{e_1\}) + \psi(\{e_2\})$.

Given an unweighted graph, the *vertex betweenness centrality* is defined as follows (see [HHKY02].) Let $\sigma_{ww'}$ be the number of shortest paths between w and w'. Similarly, let $\sigma_{ww'}(v)$ be the number of shortest paths between w and w' that pass through v. The vertex betweenness centrality for a vertex v is given by:

$$C_B(v) = \sum_{w \neq w' \in V} \frac{\sigma_{ww'}(v)}{\sigma_{ww'}} \tag{2.2}$$

Similarly, the *edge betweenness centrality* for an edge e, is given by:

$$C_B(e) = \sum_{w \neq w' \in V} \frac{\sigma_{ww'}(e)}{\sigma_{ww'}} \tag{2.3}$$

An edge with high betweenness centrality is along many shortest paths. Thus, pairs of edges with high betweenness (where the paths considered are from energy source to energy sink) are good candidates for critical pairs of edges.

2.3 Shortest Paths

Given a power supply graph $G = (V, E)$, augment G by adding in a super-source σ. Edges are added from all generators to the super-source σ. Let $N = G \cup \{\sigma\}$. We develop an algorithm for ranking $\wp_2(E)$ (all subsets of E containing 2 elements) using shortest paths from σ to each individual load i. (We can also group loads together by placing in "semi-super-loads" to represent neighborhoods or cities to reduce computation time.)

The algorithm developed in this section is based on the following observations:

Let $e_1 \in E$ and $e_2 \in E$, $e_2 \neq e_1$

1. If e_1 and e_2 are along the same paths to a load node, then e_1 and e_2 are most likely sub-additive as removal of both edges will affect the same path.

2. If e_1 and e_2 are only along paths to different loads, then e_1 and e_2 are most likely additive. Flow along these edges affects different loads.

3. If e_1 and e_2 are along different paths to the same load, then e_1 and e_2 may be super-additive. That is the removal of both e_1 and e_2 will affect different paths to the same load. Thus, at least 2 routes are affected.

Based upon the above observations, we wish to find pairs of edges along distinct paths from the σ to any load node i (or load set λ).

Before beginning the algorithm, the user should decide which (if any) loads to group together. After the loads are grouped, create the appropriate network N with super-source and semi-super-loads. The user should then rank the loads in order of importance (i.e. hospitals, airports etc), with the highest given rank 1, next rank 2, etc. For each load vertex i, let $n(i)$ be the ranked value. Ties are allowed; if there is no preference let $n(i) = 1$ for all i.

The algorithm, for each load (or semi-super-load), i is as follows:

1. Find the set P_i of all shortest paths from σ to i. Let $\Pi = \{e \in E \mid e \in p_j\}$ for some path $p_j \in P_i$.

2. Consider the set $\bigcap P_i$ of all pairwise intersections of paths $p \in P_i$.

3. For each edge pair e_j, $e_k \in \Pi$, let $\pi_k(e_j)$ denote the number of paths in P that contain e_k but not e_j. Similarly, let $\pi_j(e_k)$ denote the number of paths in P that contain e_j but not e_k.

4. Give the pair e_j, e_k the ranking number:

$$R(\{e_j, e_k\}) = \sum_i n(i)\pi_k(e_j)\pi_j(e_k)(c(e_j) + c(e_k))$$

   where $c(e)$ is the capacity of edge e.

The edge pairs are ranked by R, with the highest R having the highest ranking. It should be noted that edge pairs that are along all the same shortest paths to the same load will have rank 0. Also edge pairs that are not involved in shortest paths to the same load also have rank 0. While we believe this would be an excellent way to find and rank pairs of edges, it is extremely difficult in practice as finding the set of all shortest paths is NP-complete.

2.4 Edge Betweenness

Edges with high betweenness relating to shortest paths from sources to sinks, are excellent candidates for critical edges. These edges are along the most direct paths from the energy source to the consumer. The following algorithm is adapted from the shortest path and vertex betweenness algorithms found in [N01].

Consider an energy graph with set of energy sources Σ and individual loads. Compute the shortest paths from all $s \in \Sigma$ to all other nodes in the network as follows (this is directly from [N01] and is essentially a breadth first search from each source):

1. Pick a source $s \in \Sigma$.

2. Assign distance 0 to s. Set variable $d = 0$.

3. For each vertex k whose assigned distance is d, follow the edges from k to each node l. Declare k a predecessor of l. Assign each vertex l to have a distance of $d + 1$. We note that a vertex may have more than one predecessor, although each node has only one distance label.

4. Set d to $d + 1$ and repeat step 2 until all nodes have distance labels.

5. Let d_{max} denote the distance label of the last node labelled.

To find the source edge betweenness, $b_s(e)$ value for each edge, first label each vertex with $b_s(v) = 1$. We next begin with the vertices furthest from the source. That is we begin with all vertices with distance $d_{\max}$.

1. At each vertex, count the number of edges, n with tail at v. (We speak of heads and tails here as we are only interested in the shortest paths from the super-source). Give each edge with tail at v the value, $b_s(e) = \frac{b_s(v)}{n}$.

2. For each node, w at the head of the each edge, e in step 1 above, relabel the node to be $b_s(w) + b_s(e)$.

3. Repeat steps 1 and 2 until you reach s. Notice that all edges which have v as a head will already be labelled.

4. Repeat the above for all $s \in \Sigma$. The final edge betweenness is given by the sum

$$b(e) = \sum_{s \in \Sigma} b_s(e)$$

The above algorithm gives the betweenness for both edges and vertices. This quantity is not the full betweenness of the graph as we are only counting shortest paths from the set of sources.

2.5 Shortest Cycle

The betweenness will locate edges that occur along many paths to consumers. Since we are interested in pairs of edges, it is interesting to consider the shortest cycle containing an edge of high betweenness. Thus, to find critical pairs of edges, consider the following:

Rank the edge in order of betweenness with the highest first. For each edge of high betweenness (a measure to be determined by the user) compute the shortest cycle containing that edge. All pairs of edges along this cycle should be considered to pair with the original edge. Natural choices to limit the number of pairs are to take the two edges with highest betweenness, take the two edges with highest capacity, and/or take the two edges that occur in more than one shortest cycle (again very difficult to find all shortest cycles in a large energy graph).

2.6 Examples

We consider the following two examples. The first example does not have many cycles, so there are fewer sets of pairs of edges considered critical, while several of the single edge cuts will result in loss of flow. The second example has more cycles so the ranking of pairs is more relevant. Again, the algorithm only picks a single shortest cycle so not all pairs are found.

The edge betweenness for the first example is found in Figure 2.2.

The pairs of edges from highest level of "criticality" to lowest are found in Figure 2.3. Notice here that although these edges appear directed in the table, they are not in the original graph and thus should be considered a bi-directional edge.

Since the example in Figure 2.1 is fairly small, we have computed the paired edge rankings from the shortest path section. We have listed these in Figure 2.6.

Our next example is found in Figure 2.6.

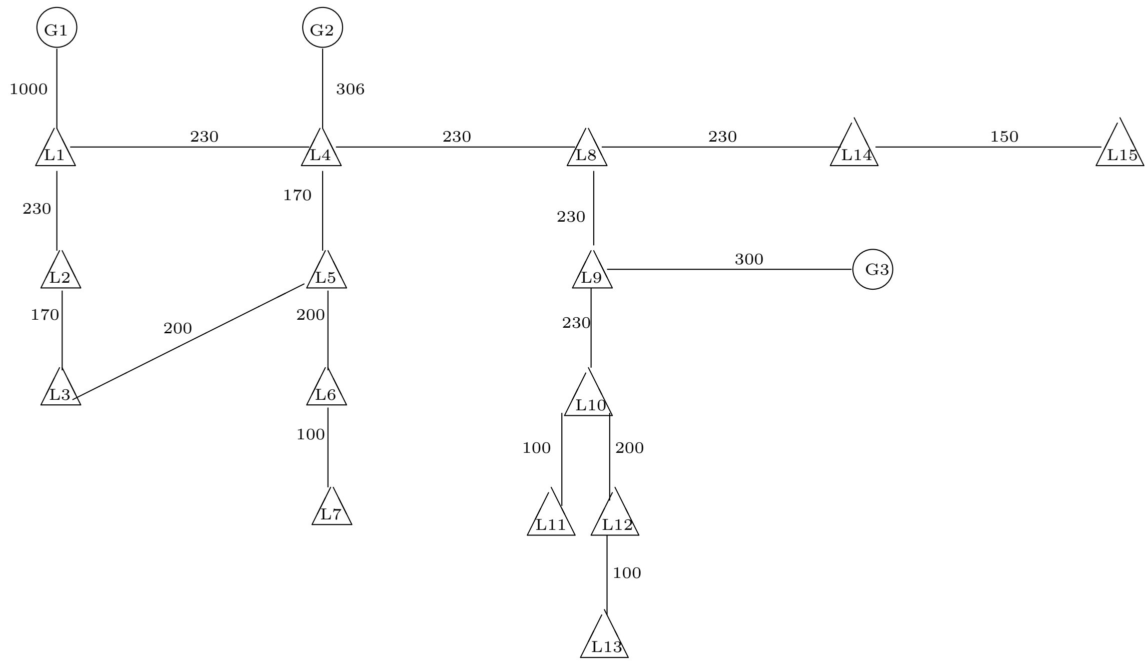


Figure 2.1: A small system network, the circles are producers, the triangles consumers, and the numbers represent line capacity.

Tail	Head	Betweenness
L4	L8	27.0
L8	L9	24.0
G3	L9	19.0
L4	G2	19.0
L1	G1	19.0
L5	L4	9.5
L2	L1	5.5
L10	L9	12.0
L4	L1	22.5
L5	L3	3.5
L2	L3	3.5
L12	L10	6.0
L5	L6	6.0
L14	L8	6.0
L13	L12	3.0
L6	L7	3.0
L11	L10	3.0
L14	L15	3.0

Figure 2.2: Edge Betweenness for the Graph in Figure 2.1

Edge 1	Edge 2
$L1 \to L4$	$L1 \to L2$
$L1 \to L4$	$L4 \to L5$
$L1 \to L4$	$L3 \to L2$
$L1 \to L4$	$L5 \to L3$
$L1 \to L2$	$L4 \to L5$
$L1 \to L2$	$L3 \to L2$
$L1 \to L2$	$L5 \to L3$
$L4 \to L5$	$L3 \to L2$
$L4 \to L5$	$L5 \to L3$
$L3 \to L2$	$L5 \to L3$

Figure 2.3: Pairs of Edges from Figure 1 listed in order from highest level of criticality to lowest

Edge 1	Edge 2	Shortest Path Ranking
$G2 \to L4$	$G3 \to L9$	1818
$G2 \to L4$	$L8 \to L9$	1608
$L4 \to L8$	$G3 \to L9$	1590
$L4 \to L8$	$L8 \to L9$	1380
$G1 \to L1$	$G2 \to L4$	1306
$G1 \to L1$	$L3 \to L5$	1200
$G1 \to L1$	$L4 \to L5$	1170
$L1 \to L2$	$G2 \to L4$	536
$L1 \to L2$	$L4 \to L5$	500
$L2 \to L3$	$G2 \to L4$	476
$L1 \to L2$	$L3 \to L5$	430
$L2 \to L3$	$L3 \to L5$	370
$L2 \to L3$	$L4 \to L5$	340

Figure 2.4: Shortest Path pairs ranking for Figure 2.1

Edge 1	Edge 2
$L4 \to L8$	$L8 \to L9$
$L4 \to L8$	$L9 \to G3$
$L4 \to L8$	$G2 \to L4$
$L8 \to L9$	$L9 \to G3$
$L8 \to L9$	$G2 \to L4$
$L4 \to L1$	$G1 \to L1$
$L4 \to L1$	$G2 \to L4$
$G3 \to L9$	$G2 \to L4$
$G2 \to L4$	$G1 \to L1$
$L1 \to L4$	$L1 \to L2$
$L1 \to L4$	$L4 \to L5$
$L1 \to L4$	$L3 \to L2$
$L1 \to L4$	$L5 \to L3$
$L1 \to L2$	$L4 \to L5$
$L1 \to L2$	$L3 \to L2$
$L1 \to L2$	$L5 \to L3$
$L4 \to L5$	$L3 \to L2$
$L4 \to L5$	$L5 \to L3$
$L3 \to L2$	$L5 \to L3$

Figure 2.5: For the first example - pairs obtained by using the betweenness then computing the shortest cycles using the super-source. Any edges from the super-source are not included. Also, there are now 2 shortest cycles containing edge $L1 \to L4$, so I picked the different one. Also, all pairs following the second horizontal line are in the shortest cycle ranking

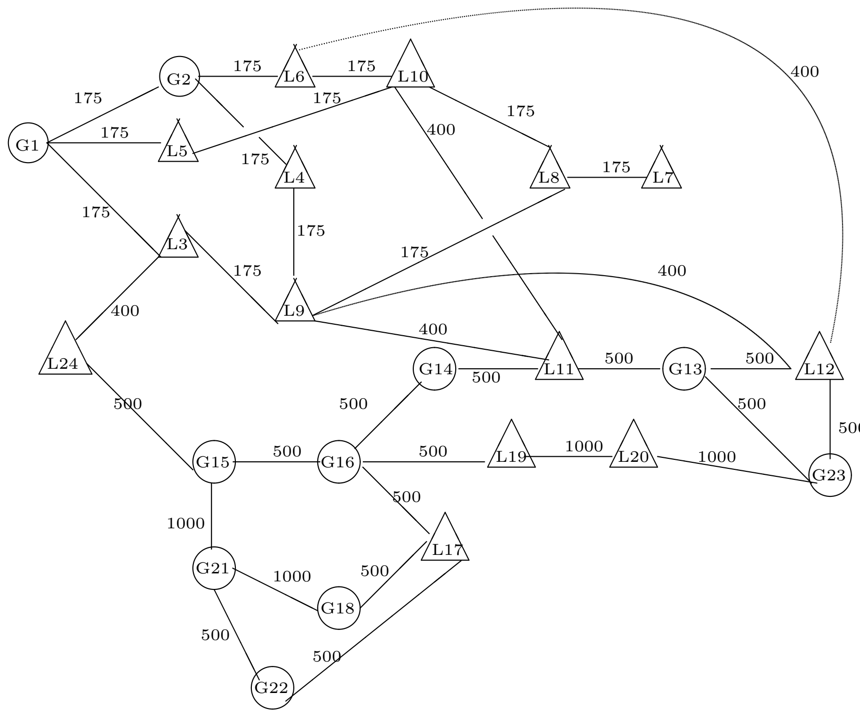


Figure 2.6: Another small electrical network

The table of betweenness values for Figure 2.6 is found in Figure 2.7, and the ranked edge pairs in Figures 2.8, 2.9, and 2.10.

Tail	Head	Betweenness
G16	G14	67.02430555555556
G16	G15	43.697916666666664
G21	G15	39.53124999999999
G16	L17	50.80208333333333
G15	L24	43.89583333333333
G14	L11	53.579861111111114
L24	L3	42.89583333333333
G16	L19	40.19097222222222
G23	L20	36.19097222222222
G21	G18	26.25
G13	L11	25.215277777777775
L19	L20	36.19097222222222
G22	G21	17.947916666666664
G23	L12	28.31597222222222
L11	L10	31.656249999999996
G1	L3	23.854166666666668
G22	L17	30.135416666666664
G13	G23	16.895833333333332
L6	G2	18.208333333333332
L17	G18	23.333333333333336
L9	L11	24.13888888888889
L6	L10	21.135416666666668
L9	L3	26.375000000000004
L5	L10	32.0
L5	G1	32.0
G2	G1	18.020833333333336
L6	L12	15.09375
L8	L10	22.020833333333332
L9	L12	24.34722222222222
G13	L12	18.72222222222222
L9	L8	22.020833333333332
L9	L4	18.520833333333332
G2	L4	17.520833333333332
L8	L7	10.0

Figure 2.7: Betweenness values for the edges in Figure 2.6

Edge 1	Edge 2
$L11 \rightarrow G14$	$G16 \rightarrow G14$
$L11 \rightarrow G14$	$L19 \rightarrow G16$
$L11 \rightarrow G14$	$G23 \rightarrow L20$
$L11 \rightarrow G14$	$L20 \rightarrow L19$
$L11 \rightarrow G14$	$L11 \rightarrow G13$
$L11 \rightarrow G14$	$G13 \rightarrow G23$
$G16 \rightarrow G14$	$L19 \rightarrow G16$
$G16 \rightarrow G14$	$G23 \rightarrow L20$
$G16 \rightarrow G14$	$L20 \rightarrow L19$
$G16 \rightarrow G14$	$L11 \rightarrow G13$
$G16 \rightarrow G14$	$G13 \rightarrow G23$
$L19 \rightarrow G16$	$G23 \rightarrow L20$
$L19 \rightarrow G16$	$L20 \rightarrow L19$
$L19 \rightarrow G16$	$L11 \rightarrow G13$
$L19 \rightarrow G16$	$G13 \rightarrow G23$
$G23 \rightarrow L20$	$L20 \rightarrow L19$
$G23 \rightarrow L20$	$L11 \rightarrow G13$
$G23 \rightarrow L20$	$G13 \rightarrow G23$
$L20 \rightarrow L19$	$L11 \rightarrow G13$
$L20 \rightarrow L19$	$G13 \rightarrow G23$
$L11 \rightarrow G13$	$G13 \rightarrow G23$
$G1 \rightarrow G2$	$L5 \rightarrow L10$
$G1 \rightarrow G2$	$G1 \rightarrow L5$
$G1 \rightarrow G2$	$L6 \rightarrow G2$
$G1 \rightarrow G2$	$L10 \rightarrow L6$
$L5 \rightarrow L10$	$G1 \rightarrow L5$
$L5 \rightarrow L10$	$L6 \rightarrow G2$
$L5 \rightarrow L10$	$L10 \rightarrow L6$
$G1 \rightarrow L5$	$L6 \rightarrow G2$
$G1 \rightarrow L5$	$L10 \rightarrow L6$
$G15 \rightarrow G21$	$G16 \rightarrow L17$
$G15 \rightarrow G21$	$G18 \rightarrow G21$
$G15 \rightarrow G21$	$L17 \rightarrow G18$
$G15 \rightarrow G21$	$G15 \rightarrow G16$

Figure 2.8: Edge Pairs listed from most to least "critical" for the network in Figure 2.6 - continued in the next table

Edge 1	Edge 2
$G16 \rightarrow L17$	$G18 \rightarrow G21$
$G16 \rightarrow L17$	$L17 \rightarrow G18$
$G16 \rightarrow L17$	$G15 \rightarrow G16$
$G18 \rightarrow G21$	$L17 \rightarrow G18$
$G18 \rightarrow G21$	$G15 \rightarrow G16$
$L17 \rightarrow G18$	$G15 \rightarrow G16$
$L12 \rightarrow L9$	$L12 \rightarrow L6$
$L12 \rightarrow L9$	$L9 \rightarrow L8$
$L12 \rightarrow L9$	$L10 \rightarrow L6$
$L12 \rightarrow L9$	$L8 \rightarrow L10$
$L12 \rightarrow L6$	$L9 \rightarrow L8$
$L12 \rightarrow L6$	$L10 \rightarrow L6$
$L12 \rightarrow L6$	$L8 \rightarrow L10$
$L9 \rightarrow L8$	$L10 \rightarrow L6$
$L9 \rightarrow L8$	$L8 \rightarrow L10$
$L10 \rightarrow L6$	$L8 \rightarrow L10$
$G21 \rightarrow G22$	$G21 \rightarrow G18$
$G21 \rightarrow G22$	$G18 \rightarrow L17$
$G21 \rightarrow G22$	$L17 \rightarrow G22$
$L11 \rightarrow L9$	$L11 \rightarrow L10$
$L11 \rightarrow L9$	$L8 \rightarrow L9$
$L11 \rightarrow L9$	$L10 \rightarrow L8$
$L11 \rightarrow L10$	$L8 \rightarrow L9$
$L11 \rightarrow L10$	$L10 \rightarrow L8$

Figure 2.9: Edge Pairs listed from most to least "critical" for the network in Figure 2.6 continued from the previous table and continued in the next table

Edge 1	Edge 2
$L12 \rightarrow G23$	$L12 \rightarrow G13$
$L12 \rightarrow G23$	$G13 \rightarrow G23$
$L12 \rightarrow G13$	$G13 \rightarrow G23$
$L3 \rightarrow G1$	$L3 \rightarrow L9$
$L3 \rightarrow G1$	$L5 \rightarrow L10$
$L3 \rightarrow G1$	$G1 \rightarrow L5$
$L3 \rightarrow G1$	$L8 \rightarrow L9$
$L3 \rightarrow G1$	$L10 \rightarrow L8$
$L3 \rightarrow L9$	$L5 \rightarrow L10$
$L3 \rightarrow L9$	$G1 \rightarrow L5$
$L3 \rightarrow L9$	$L8 \rightarrow L9$
$L3 \rightarrow L9$	$L10 \rightarrow L8$
$L11 \rightarrow G13$	$L11 \rightarrow L10$
$L11 \rightarrow G13$	$L12 \rightarrow G13$
$L11 \rightarrow G13$	$L6 \rightarrow L12$
$L11 \rightarrow G13$	$L10 \rightarrow L6$
$L11 \rightarrow L10$	$L12 \rightarrow G13$
$L11 \rightarrow L10$	$L6 \rightarrow L12$
$L11 \rightarrow L10$	$L10 \rightarrow L6$
$L12 \rightarrow G13$	$L6 \rightarrow L12$
$L12 \rightarrow G13$	$L10 \rightarrow L6$
$L4 \rightarrow L9$	$L4 \rightarrow G2$
$L4 \rightarrow L9$	$L9 \rightarrow L8$
$L4 \rightarrow L9$	$L6 \rightarrow G2$
$L4 \rightarrow L9$	$L10 \rightarrow L6$
$L4 \rightarrow L9$	$L8 \rightarrow L10$
$L4 \rightarrow G2$	$L9 \rightarrow L8$
$L4 \rightarrow G2$	$L6 \rightarrow G2$
$L4 \rightarrow G2$	$L10 \rightarrow L6$
$L4 \rightarrow G2$	$L8 \rightarrow L10$
$L9 \rightarrow L8$	$L6 \rightarrow G2$

Figure 2.10: Edge Pairs listed from most to least ”critical” for the network in Figure 2.6 - continued from the previous table

2.7 Ranking

The reduction of flow caused by removal of a subset $S \subset E$ can be bounded above by the sum of the line capacities as well as the amount of flow along those lines given a solution to the power flow. In addition the vulnerability of the lines should be considered. Computing the vulnerability may be difficult - can consider the number of times it has malfunctioned in the past. Also possibly the time required to repair a given line.

The set of individual lines is ranked based upon betweenness. To obtain a ranking of pairs of edges, first rank by betweenness. Then starting with the edge of highest betweenness compute the shortest cycle containing that edge. Pick all pairs in that cycle, beginning with the edge of highest betweenness. Repeat for the edge with next highest betweenness, being sure not to repeat pairs of edges. This method can easily be adapted to triples of edges as well as higher order sets of edges.

We also note that while we have concentrated on sets of edges here, the same methods can be applied to sets of nodes, as the betweenness values for both nodes and edges are computed. To find the appropriate nodes within the shortest cycle, one only needs to look at the endpoints of the edges in the shortest cycle.

Chapter 3

Graph-Ranking Approach: Robustness and Reliability Metrics for Energy Transmission Networks

3.1 Introduction

We begin to develop infrastructure robustness measures that account for the dynamical properties and physics of the infrastructures. These robustness measures help to describe the ability of the network to serve its customers. We are attempting to develop a general set of terminology which does not apply equally to different physical infrastructures. In particular, we want to answer these sorts of questions (which again have different interpretations depending on the physics of the particular infrastructure):

- How close is the system to its consumption, production, and/or capacity limit?
- How much can production limits and/or capacity limits be reduced and still serve consumption?
- How much can consumption be increased given the production and capacity limits?
- How much production or capacity must be lost to cause some consumption site to lose service?
- How much production or capacity must be lost to cause all consumption sites to lose service?
- Where can additional production capability or line capacity be added to the network to make it more robust?
- In what localized area does a production site supply consumption sites?
- In what localized area does a consumption site draw on production sites?
- How much does random degradation of the system affect its ability to serve consumption?
- How much does an attack on one consumption site affect others?
- Are a few transmission lines or production sites responsible for most of the system capacity or are many?
- What is the structure of minimum cuts and how are they correlated?

- How many pieces can the graph be broken into and still have the consumption satisfied (or $x\%$ of it satisfied)?
- How much time does it take to restore the system after contingencies occur?

Answers to the above questions can be used for:

- assessment of system in contingency screening without solving full equations
- contingency selection
- system modification (i.e., adding production)

The fundamental premise behind this work is to use the static and steady-state properties of the network to make rough predictions of its response to perturbations and contingencies without having to solve the full nonlinear or differential equations describing the system. In general we have 3 approaches: (1) data-driven (machine learning); (2) physics-driven (reduction methods); and (3) graph- theoretic. We would like to develop several types of robustness metrics: topological (purely graph-theoretic), flow, and temporal. In each case, we are interested in both global and local measurements.

The physics of energy transmission depends heavily on the type of energy. For example, natural gas flow depends on pressure, pipe diameter and pipe length, whereas electric power depends on voltage and phase angles. However, despite the energy type, flow conservation must occur at each node of the network. this is described by the ”mass balance” equation:

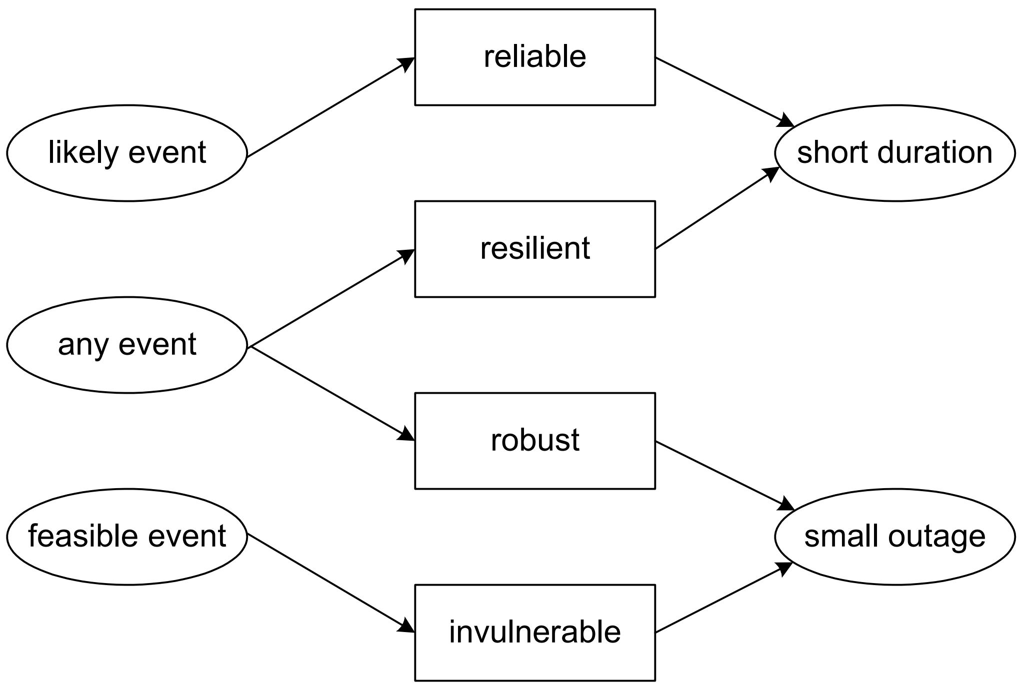
$$\sum_{i=1}^{n_i} m_{ij,pipe} + \sum_{k=1}^{n_{sink}} m_{k,sink} - \sum_{l=1}^{n_{source}} m_{l,source} = 0, \quad i = 1, \ldots, N \tag{3.1}$$

where
N = number of nodes in the network (including the reference node)
n_i = number of edge connections to node i
m_{ij} = flow rate from node i to node j
n_{source} = number of power source nodes
n_{sink} = number of power sink nodes

For flow related robustness measures, the mass balance equation will be the most important. For topological measures, the mass balance or flow equation is also important, as we are interested in how the shape of the network affects the power flow. We note here that flow and topological measures will work for all fluid networks. The purely topological metrics will work for communication and transportation networks as well.

3.1.1 Definitions

We wish to measure the ability of the network to satisfy consumption demands. In particular, disruptions in service should be minimal and quickly repaired. The following graphic represents the temporal definitions of reliable, resilient, robust and invulnerable.



metric	likelihood of event	size of event	duration of outage	severity of outage
reliability	likely	small	short	minor
robustness	any	any	any	minor
fragility	any	any	any	severe
resilience	any	small	short	minor
performance	any	any	any	any
invulnerability	rare	any	long	severe

Definition 3.1.1 *A* contingency *is any event that affects the transmission network in a negative manner with regards to power transmission. A contingency may affect one, many or all of the components of the network. Usually a contingency involves at least one component failure, but it may also simply be a reduction in production ability or a reduction in capacity along a flow line.*

Reliability

Traditionally, reliability metrics are mean time to failure, mean failure duration, and fraction of outage hours per consumer. Reliability measures how the network responds to naturally-occurring component outages and how long it takes to repair the components and restore the system to full operation. The analysis of reliability requires a model for the probability that particular components may fail and the distribution of repair times for them. Reliability is not such a useful notion when natural disasters, intentional interference with a network, or other extreme events are considered.

Reliability and performance also tend to reference flow; how close you are to maximum flow, or the probability that the network can meet a given demand. Quantities used to measure reliability in the past are: expected demand not served, demand not served, and forced outage rate (see

[SL 99]). Where, demand not served is the difference between the demand and the power generated by the network in a given state; the expected demand not served is an average of the demand not served over a set of simulations of likely events. In addition, reliability has been measured by the probability that the network will be functioning at time t. This is dependent upon the probabilities that each component is functioning and dependent upon the probability that the system was functioning for times $s < t$. (See [LSS 98].) In general, a network is reliable at a given time if there is a d-capacity path from source node to sink node, where the sink node has demand d. We propose the following definition of reliability:

Definition 3.1.2 *The* reliability *of a network is the expected fraction of the time that the network is functional to supply all demand (or sufficiently functional to supply a particular location) under "normal" operating conditions.*

In [LJY 95] and [DJ 72], a method for computing the reliability involves decomposing the state space into disjoint sets of accepted states, unaccepted states and unspecified states. The unspecified states are further decomposed until all states are classified as accepted or unaccepted. A state is acceptable if the demand met is d for a given d. The probability of each accepted state is computed and the reliability is the sum of the probabilities of all the accepted states.

The North American Electric Reliability Council defines reliability in terms of two criteria: adequacy and security (see [NERC 02]) . Adequacy is the ability to serve customers under normal or expected operating conditions and hence is similar to our definition of reliability. Security is the ability of the network to perform under sudden or unexpected occurrences. This is related to our definitions of robustness and vulnerability which we give later.

Robustness

Topologically speaking robustness and resilience are usually a reference to connectivity and/or connectivity after the deletion of edges and/or vertices. For example, assume that nodes are removed from the network randomly. A measure of robustness is p_c, where p_c is the smallest fraction of nodes that need to be removed from the network so that the network disintegrates into smaller pieces (see [CEA 00]).

With respect to energy infrastructures, we are interested in demand not met, or unsatisfied consumers. Demand of course fluctuates depending upon the season, the time of day and the price of the energy source.

We suggest the following definition of robustness:

Definition 3.1.3 *The* robustness *of a network is how much damage it can sustain before failing to serve customers (or to serve them at a particular partial level), regardless of the likelihood of the damage or the amount of time it would take to repair the damage and restore service.*

Since demand fluctuates, the robustness measure is a function of total demand. To obtain a single measure of robustness, we would need to consider all seasonal changes in demand and combine the results.

Definition 3.1.4 Fragility *is the opposite of robustness. (Here "damage" is a general term that may be defined in terms of the number of failed components, total capacity lost, etc.)*

Not all reliable systems are robust, because they may have critical components (see below) that have low probability of failure or are easily repaired under normal circumstances, but whose loss would cause severe degradation of the system (in terms of providing service). Not all robust systems are reliable, because they may have frequently-failing components that take a long time to repair.

Resilience

As mentioned above, resilience usually refers to a measure of connectivity or structure of a network after node or arc failures. In [SHL 96], resilience is measured in terms of capacity reliability, the probability that the carrying capacity meets the flow demand. In [BOS 01], resilience against arc failure is described as follows: given (s,t) and demand T, on failure of k arcs, there is sufficient reserved capacity in the remainder of the network to support an (s,t) flow of value T.

We propose the following definition of resilience to distinguish resilience and robustness:

Definition 3.1.5 *The* resilience *of a network is how quickly (i.e., in how little time) it can recover after a contingency that affects customers (either all or specific ones).*

A resilient network may or may not be reliable or robust, but reliable networks are typically resilient.

Invulnerability

In general, invulnerability has been any measure of graph integrity, strength or toughness. Vulnerability of a network has been measured in terms of the tendency for the network to become disconnected, and the size of the largest component after a disconnection (edge or vertex integrity). Alternatively, vulnerability has been measured in terms of the diameter of the network and the average distance between node pairs (or betweenness). (See [HKY 02] and [LKKK 98]). In [HKY 02], attack vulnerability is a measure of decrease in network performance where network performance is measured in terms of functionality and betweenness. In terms of power transmission networks, disconnection of the network is only a problem if the subnetworks cannot meet the demand. Also, the diameter of a power supply network might be large, but the distance between consumer pairs is not as important as the distance between producer/consumer pairs.

We would also like to include the susceptibility of the network to attack, for even though some components might be critical to the network, they may not contribute to its vulnerability if they are somehow protected from damage. We think of the attack vulnerability as a decrease of network performance due to a selected removal of vertices or edges (as in [HKY 02]), but we would also like to take into account the probability of attack success.

Definition 3.1.6 *A network is* vulnerable *if a small number of components can be targeted for damage to cause a severe affect on customers (or on a particular customer).*

Definition 3.1.7 *A network is* invulnerable *if it is not vulnerable.*

Flexibility

An energy system is flexible if it can respond to all different types of demand. Demand types change during the time of day (more power needed in downtown areas during the work week for example).

Definition 3.1.8 *Consider a network with current demand level δ A network is* δ-flexible *if the reliability of the network does not change regardless of the demand distribution over the network.*

Performance

Every power system has a limit to the amount of power that can be supplied. Each component has a limit, lines have capacity limits, producers have production limits, etc. In general, a power system can withstand demand growth or damage to the system if under normal conditions, the system is operating well under its limits on all components. Performance Indices measure "closeness" of a power system is to its limit.

In, [WD 97], a performance index for AC power flow is described as a weighted average of individual performance Criteria. The performance criteria are computed on individual components affected by the criterion (e.g. brownouts, line overloading). The performance of each component is compared to a theoretical limit taking into account relevance of geometric location.

Definition 3.1.9 *The* performance *of a network is how close it is operating to its capacity; networks operating in high performance conditions have little available resources for additional service, nor for adjusting to handle contingencies. Such systems are brittle and break easily; hence, they are neither reliable, resilient, robust nor invulnerable.*

Criticality

Definition 3.1.10 *A component in a network is* critical *if the network fails to function or is significantly degraded when the component is damaged.*

A component of the power system is critical if damage to it causes loss of demand met by some customer. A component may be extremely critical, critical, moderately critical, or not critical depending upon the loss of demand met when the component is removed from the network.

Criticality can be measured by the effects of removal of the component on its neighbors; the region of influence (how many customers will be affected); as well as electrical or topological distance from customers or producers.

3.2 Terminology

3.2.1 Graphs & Networks

We represent an energy transmission network N as a directed graph with vertices $V = \{v_1, \ldots, v_n\}$ and edges $E = \{e_1, \ldots, e_m\}$. Write $\mathbf{in}(v)$ for the set of edges entering vertex v and $\mathbf{out}(v)$ for the set of edges leaving vertex v. An edge e which points from vertex a to vertex b can be written as an ordered pair, $e = (a, b)$; we call a the *tail*, $\mathbf{tail}(e) = a$ and b the *head*, $\mathbf{head}(e) = b$.

Some of the vertices $P \subseteq V$ have the capacity to produce flow. The *maximum production* limit is expressed as a function $\bar{p} : V \to \mathbb{R}^+$ where $\bar{p}(v) > 0, \forall v \in P$. The *actual production* is written as another function $p : V \to \mathbb{R}^+$; it is *feasible* if $0 \le p(v) \le \bar{p}(v), \forall v \in V$. The *total production* is written $\mathbf{val}(p) = \sum_{v \in P} p(v)$.

Likewise, some of the vertices $C \subseteq V$ have the capacity to consume flow. The *maximum consumption* limit is also expressed as a function $\bar{c} : V \to \mathbb{R}^+$ where $\bar{c}(v) > 0, \forall v \in C$. The *actual consumption* is written as $c : V \to \mathbb{R}^+$; it is *feasible* if $0 \le c(v) \le \bar{c}(v), \forall v \in V$. The *total consumption* is written $\mathbf{val}(c) = \sum_{v \in C} c(v)$.

The *capacity* (i.e., maximum flow limit) for edges is expressed as a function $\bar{f} : E \to \mathbb{R}^+$. The actual *flow* on edge is written as $f : E \to \mathbb{R}$; it is *feasible* if $0 \le f(e) \le \bar{f}(e), \forall e \in E$ and

$$p(v) + \sum_{e \in \mathbf{in}(v)} f(e) = c(v) + \sum_{e \in \mathbf{out}(v)} f(e), \quad \forall v \in V. \tag{3.2}$$

It is sometimes convenient to augment a network $N = (V, E)$ with an additional vertex σ connected to each vertex in the set of producers P: $N_\sigma = (V_\sigma, E_\sigma)$ where $V' = V \cup \{\sigma\}$ and $E_\sigma = E \cup \bigcup_{v \in P} (\sigma, v)$. We call the vertex σ a *super-producer*. Likewise, we sometimes augment a network with an additional vertex τ connected to each vertex in the set of consumers C: $V_\tau = V \cup \{\tau\}$ and $E_\tau = E \cup \bigcup_{v \in C} (v, \tau)$. In other cases, we add both σ and τ to the network: $N_{\sigma\tau} = (V_{\sigma\tau}, E_{\sigma\tau})$. Whenever edges are added to the network, we need to expand the definition of $\bar{f}$ to the enlarged domain.

Given a production capability $\bar{p}$, consumption demand c, and edge capacity $\bar{f}$, the *flow problem* determines a feasible flow f, production p, and consumption c' that maximizes $\mathbf{val}(c')$ subject to the condition $c' \leq c$. We write this $(f, p, c') = F(V, E, \bar{f}, \bar{p}, c)$. Note that the solution might not be unique, so we write the set of all such solutions as $\mathcal{F}(V, E, \bar{f}, \bar{p}, c)$. We say the flow problem is *unconstrained* if there are no additional conditions on the flow other than it be feasible; the flow problem is *constrained* if additional conditions such as pressure (for gas or fluid transport) or voltage (for electric power transmission) relationships are imposed. Furthermore, we call $s = c - c'$ the amount of consumption *shed*.

Closely related to the maximum flow is the minimum cut. We write $Q(V, E, w, s, t)$ for a set of edges with minimum total weight, $q(V, E, w, s, t)$, that separates the vertex s from the vertex t, where $w(e)$ is the weight of an edge:

$$Q(V, E, w, s, t) = \arg\min_{S \subseteq E} \left\{ \sum_{e \in S} w(e) \,\middle|\, \text{there is no path from } s \text{ to } t \text{ in } (V, E - S) \right\}, \tag{3.3}$$

$$q(V, E, w, s, t) = \min_{S \subseteq E} \left\{ \sum_{e \in S} w(e) \,\middle|\, \text{there is no path from } s \text{ to } t \text{ in } (V, E - S) \right\}. \tag{3.4}$$

$$\tag{3.5}$$

Since Q might not be unique, we write the set of all such minimum cuts as $\mathcal{Q}(V, E, w, s, t)$.

3.2.2 Infrastructure Physics

Natural Gas

The physics of the natural gas infrastructure is given by the following equation, where all variables are real numbers:

$$\sum_j Q_{ij} = L_i - G_i \tag{3.6}$$

and

$$Q_{ij} = FC_1 \left(\frac{T_b}{P_b} \right)^{C_2} D_{ij}^{C_5} \varepsilon_{ij} \left(\frac{P_i^2 - P_j^2}{G^{C_3} d_{ij} T_{avg} Z_{avg}} \right)^{C_4} \tag{3.7}$$

where:

P_i	=	pressure at i
Q_{ij}	=	flow from node i to node j
G_i	=	production at i
L_i	=	consumption at i
D_{ij}	=	diameter of pipe from i to j
ε_{ij}	=	efficiency of pipe from i to j
d_{ij}	=	length of pipe from i to j
T_b	=	base temperature
P_b	=	base pressure
G	=	specific gravity
Z_{avg}	=	gas compressibility
F	=	transmission factors, including friction inside the pipe
C_1, C_2, C_3, C_4, C_5	=	constants

The constants, C_1, C_2, C_3, C_4 and F depend on the diameter of the pipeline, and the turbulence in the pipeline.

Electric Power

The electric power infrastructure is represented by the following. We define the following variables which are complex numbers:

V_i	=	voltage at node i
Q_{ij}	=	flow from i to j
G_{ij}	=	production at node i
L_i	=	consumption at i

The equations are:

$$\frac{G_i - L_i}{V_i^*} = \sum_{j \neq i} Y_{ij} V_j + Y_{ii} V_i \tag{3.8}$$

$$Y_{ij} = -y_{ij} \tag{3.9}$$

$$Y_{ii} = \sum_{j} y_{ij} + y'_i \tag{3.10}$$

$$Q_{ij} = V_i^* Y_{ij} V_j \tag{3.11}$$

where:

y_{ij}	=	line admittance from i to j
y'_i	=	bus admittance at i

3.3 Connectivity Metrics

The static topology of a transmission network is represented by an unweighted graph $G = (V, E)$. Some interesting questions about robustness can be answered by examining this graph. Reference [BFT 01] contains a detailed discussion of unweighted infrastructure graphs, along with detailed measurements of actual networks. Here we only focus on metrics related to robustness.

Several types of minimum cuts on unweighted graphs can be used to form robustness metrics:

- $q(V, E, 1, s, t)$, where $s \in P$ and $t \in C$, is the minimum number of edges that must be removed to separate a consumer t from a producer s.

- $q(V_\sigma, E_\sigma, 1, \sigma, t)$, where $t \in C$, is the minimum number of edges that must be removed to separate a consumer t from all producers.
- $q(V_{\sigma\tau}, E_{\sigma\tau}, 1, \sigma, \tau)$ is the minimum number of edges that must be removed to separate all of the consumers from all of the producers.

For the first two items, we have distributions of cut sizes, which may be combined into a single metric by some statistical measure such as the minimum, mean, or a quantile. These sets of cuts can also be organized into *cut trees*, which we discuss in 3.4.3 below.

3.3.1 Stability and Independence numbers

Given a network $N = (V, E)$, The *independence number* is the size of the largest independent set of vertices. A set, $S \subseteq V$, of vertices is independent if there are no edges between $v, w \in S, \forall v, w \in S$. The *stability number* is the cardinality of a maximum stable set of vertices. A set $S \subseteq V$ is stable if it consists of pairwise non-adjacent vertices. Both quantities thus reference the same number and are denoted $\alpha(G)$. See [Fo 92] and [Ra 01].

3.3.2 Strength

The strength, sometimes called the invulnerability, of a graph is given by:

$$\sigma(G) = \min_{A \subseteq E} \left(\frac{\sum_{e_i \in A} \bar{f}_i}{c(g-a) - 1} \right)$$

where $c(H)$ is the number of connected components of any graph H.

The toughness of a graph is given by:

$$t(G) = \min \left\{ \frac{|S|}{c(G-s)} | S \text{ is a separator of } G \right\}$$

where $S \subseteq V$ is a separator of G if $c(G - S) > 1$.

The scattering number of a graph is given by:

$$sc(G) = \max\{c(G-s) - |S| : S \text{ is a separator of } G\}$$

The scattering number of a complete graph is $-\infty$.

3.4 Flow Metrics

3.4.1 Performance Index

The first of these is traditional:

$$I_\delta(f, \bar{f}) = \left(\frac{1}{|E|} \sum_{e \in E} \left| \frac{f(e)}{\bar{f}(e)} \right|^\delta \right)^{1/\delta} \tag{3.12}$$

$$I_\delta(f, \bar{f}) = \left(\frac{1}{|E|} \sum_{e \in E} \left| \frac{\bar{f}(e) - f(e)}{\bar{f}(e)} \right|^\delta \right)^{1/\delta} \tag{3.13}$$

$$I_\delta(p,\bar{p}) = \left(\frac{1}{|P|}\sum_{v\in P}\left|\frac{p(v)}{\bar{p}(v)}\right|^\delta\right)^{1/\delta} \tag{3.14}$$

$$I_\delta(p,\bar{p}) = \left(\frac{1}{|P|}\sum_{v\in P}\left|\frac{\bar{p}(v)-p(v)}{\bar{p}(v)}\right|^\delta\right)^{1/\delta} \tag{3.15}$$

$$I_{\text{sys}} = \frac{\sum_{v\in P}\left(\bar{p}(v)-p(v)\right)}{\sum_{v\in C}\left(\bar{c}(v)-c(v)\right)} \tag{3.16}$$

3.4.2 Residual Capacity

Increased Consumption

Consider a solution $(f, p, c') = F(V, E, \bar{f}, \bar{p}, c)$ to the flow problem and let the amount of consumption be $x = \mathbf{val}(c)$. Now consider how much consumption can increase and still be satisfied:

$$\Delta x = \max_{\Delta c \geq 0}\left\{\|\Delta c\| \,\middle|\, (f', p', c'') = F(V, E, \bar{f}, \bar{p}, c' + \Delta c) \text{ provided } \begin{array}{c} c'' = c' + \Delta c \\ c' + \Delta c \leq \bar{c} \end{array}\right\}, \tag{3.17}$$

where $\|\cdot\|$ is any norm function suitable for measuring the importance of an increase in consumption. If $\|\Delta c\| = \mathbf{val}(\Delta c)$, for instance, then Δx is just the amount of increased consumption. A variation of this computation includes the constraint that production may only increase:

$$\Delta x = \max_{\Delta c \geq 0}\left\{\|\Delta c\| \,\middle|\, (f', p', c'') = F(V, E, \bar{f}, \bar{p}, c' + \Delta c) \text{ provided } \begin{array}{c} c'' = c' + \Delta c \\ c' + \Delta c \leq \bar{c} \\ p' \geq p \end{array}\right\}. \tag{3.18}$$

With either of these computation, we may define a metric

$$I = \frac{\Delta x}{x}. \tag{3.19}$$

Note that out-of-kilter maximum-flow algorithms can be used to solve this problem in the case of unconstrained flow.

Decreased Production

Consider a solution $(f, p, c') = F(V, E, \bar{f}, \bar{p}, c)$ to the flow problem and let the amount of production be $y = \mathbf{val}(p)$. Now consider, how much the production limits can decrease and still satisfy consumption:

$$\Delta y = \max_{\Delta \bar{p} \geq 0}\left\{\|\Delta \bar{p}\| \,\middle|\, (f', p', c'') = F(V, E, \bar{f}, \bar{p} - \Delta\bar{p}, c') \text{ provided } \begin{array}{c} c'' = c' \\ \bar{p} + \Delta\bar{p} \geq 0 \end{array}\right\}. \tag{3.20}$$

We just have $\Delta y = \mathbf{val}(\bar{p} - p)$ if $\|\cdot\| = \mathbf{val}$, which is rather too trivial to be of interest unless we consider all multiple solutions. In any case, we may again define a metric

$$I' = \frac{\Delta y}{y}. \tag{3.21}$$

Decreased Capacity

Consider a solution $(f, p, c') = F(V, E, \bar{f}, \bar{p}, c)$ to the flow problem and let the amount of capacity be $z = \mathbf{val}(f)$. Now consider how much the capacity limits can decrease and still satisfy consumption:

$$\Delta z = \max_{\Delta \bar{f} \geq 0} \left\{ \|\Delta \bar{f}\| \middle| (f', p', c'') = F(V, E, \bar{f} - \Delta \bar{f}, \bar{p}, c') \text{ provided } \begin{matrix} c'' = c' \\ \bar{f} + \Delta \bar{f} \geq 0 \end{matrix} \right\}. \tag{3.22}$$

We just have $\Delta z = \mathbf{val}(\bar{f} - f)$ if $\| \cdot \| = \mathbf{val}$, which again is rather too trivial to be of interest unless we consider all multiple solutions. Again, we may define a metric:

$$I'' = \frac{\Delta z}{z}. \tag{3.23}$$

Algorithm for computing Δz where $\| \cdot \| = \mathbf{val}$ for a given network N:

1. Find the shortest path from the Super-producer to the Super-consumer.
2. Saturate that shortest path (i.e. send as much flow through as possible).
3. Replace N with N' = the residual graph obtained after saturating the path in 2.
4. Go to Step 1 and repeat with N'.
5. Continue until the consumer is met. Let the residual network after the consumer is met be denoted M.
6. $\Delta z = \sum \bar{f}(e_i)$ for $\bar{f}(e_i)$ in network M.

Thus, Δz = sum of all remaining capacity after the consumer is met via the shortest paths.

It should be noted that if you do not saturate the shortest paths first, you may not get the maximum value for I. For, suppose that you can send an additional flow of 4 units along a path, P, from the super-producer to the super-consumer containing n edges. Then the residual capacity is at least $4n$. Thus, to maximize this sum, you wish to keep the longer paths open.

Algorithm to find $\Delta z = \max\{\max \Delta \bar{f} | (f', p', c'') = F(V, E, \bar{f} - \Delta \bar{f}, \bar{p}, c')$ provided $c'' = c'$ and $\bar{f} - \Delta \bar{f} \geq 0\}$

1. List edges in order of decreasing capacity - $e_1, e_2, ..., e_m$, with corresponding capacities $\bar{f}(e_j)$.
2. Set $i = 1$
3. For $i := 1$ to m, do:

   Look at the graph $G - e_i$, where G is the original directed graph representing the energy network.

   Solve the maximum flow problem for $G - e_i$, obtaining (f_i, p_i, c_i)

   If $c_i = c'$, then output $I = \bar{f}(e_i)$. Else, set $r = c' - c_i$ if $\bar{f}(e_i) - r >= \bar{f}(e_{i+1})$ Replace G with G, except change $\bar{f}(e_i)$ to $\bar{f}(e_i) - r$ and solve the maximum flow problem obtaining (f_i', p_i', c_i'). If $c_i' = c'$, then output $\Delta z = \bar{f}(e_i) - r$.

   Else, let $i := i + 1$ and continue
4. If $i = m + 1$, then $\Delta z = 0$.

   If $I = 0$, then any damage to any of the system's lines will result in decreased consumption.

3.4.3 Minimum Cut

Cut-Trees

In 1961 Gomory and Hu discovered an elegant way to represent $s-t$ cuts for all $s,\ t \in V$ for a given network. All cuts can be represented by a single tree graph, hence called the *cut tree*. See [GH 61, CCP 98].

The traditional cut-tree is a spanning graph of all the nodes of the original graph. The weight of an edge (s,t) in the cut-tree is the value of the minimum $s-t$ cut. For any pair of nodes a, b in the original graph, the value of the minimum $a-b$ cut is also represented in the cut-tree as follows. Since the cut-tree is a tree graph there is only one path from a to b. The value of the minimum cut is the smallest edge weight along the path from a to b.

We would like to extend the cut-tree notion to the to a "production" or "consumption" tree, where we are only interested in cutting producers from consumers, and not necessarily separating consumers from consumers. Since a consumer may be met by several producers we use the super-producer, σ, and individual consumers. If we follow the algorithm in [GH 61, CCP 98], we do get a cut tree, again with all minimum cuts between σ and any individual consumer represented. However, as minimum cuts are not unique, the cut-trees are not unique. In particular, the algorithm calls for a choice of a consumer node at each step. The choice affects the paths in the tree although the minimum value along any *s*-*t* path remains constant.

We would like to use the cut-tree to form a measure of robustness for the graph. Certainly the minimum cut is important it is a measure of how much damage the lines could take before the consumer is cut off completely. However, the number of cuts between the super-producer and the consumer is also important. Each edge along the path from σ to a consumer l represents a cut in the original graph which separates l from all sources of power. Thus, the length of the path from σ to l could be important in the robustness measure. There is of course the problem with the non-uniqueness of the cut-tree.

Cactus Representation of Minimum Cuts

A graph, N, is a cactus if no two cycles in C share more than one vertex. In particular, every edge only lies in one cycle. Cactus graphs can be used to represent all minimum cuts of a network (see [Fl 98]). This differs from the cut-tree in that it represents all the minimum cuts of the graph. We would like to extend the cactus, however we are not interested in the minimum cuts of the graph but $\sigma - l$ minimum cuts for the super-producer and individual consumers. Since a cactus can be used to represent all minimum cuts, an extension of the cactus graph should alleviate the non-uniqueness problem.

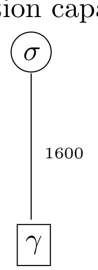
We can eliminate the non-uniqueness problem by enumerating all the minimum cuts, and adding in some nodes. Since there may be many minimum cuts, we do not always get a cactus graph. It is similar in appearance and unique, hence we call this unique graph the *production cut-cactus.* In this case, each node in the cut-cactus is a collection of consumers or producers. There is a node for σ, a node for all the producers, and nodes containing collections of consumers. The edges again represent the minimum cuts; however it is the minimum cut that separates the consumer collection from the producers. As we are not interested in separating consumers from consumers, not all minimum cuts of this type are necessarily represented. The minimum cut to separate a consumer l from σ is the smallest weight along the paths from σ to the node(s) containing l in the tree. However, the lengths of the paths between the node(s) containing l and σ can be used to find the number of distinct proper "minimal" cuts between σ and l. A cut $C \subseteq E$ is *proper* if no proper subset $C' \subset C$ is also a cut. A $\sigma - l$ cut, (A, B), with $l \in A$ is "minimal" if it is the minimum cut for the set A.

To describe the formation of the cut-cactus, we use the example in Figure 3.2 and Table 3.1. (Positive total power represents power generated, negative total power represents power consumed).

Node number	power generated	power consumed	total power
G1	700	0	700
G2	500	0	500
G3	400	0	400
L1	0	1000	-1000
L2	0	10	-10
L3	0	10	-10
L4	0	10	-10
L5	0	10	-10
L6	0	10	-10
L7	0	50	-50
L8	0	10	-10
L9	0	10	-10
L10	0	10	-10
L11	0	50	-50
L12	0	10	-10
L13	0	50	-50
L14	0	10	-10
L15	0	75	-75

Figure 3.1: The table of power generated and consumed for Figure 3.2.

We form the production cut-cactus as follows. At the top (or the root) we have σ. Then there is an edge from σ to a node γ containing all the producers. The weight of this edge is the sum of the production capacities of all the producers.



We wish now to find all minimum cuts between consumers and σ. We also want these to correspond to cuts in the original graph, so we consider the graph which has ∞ as the weight of all edges between σ and each producer found in Figure 3.3

We next find the minimum cut that separates all producers from all the consumers (note that this is just cutting all the lines leading from a producer). We draw an edge from γ to a node, λ that contains all the consumers. The weight of this edge is the value of the cut.

In our example our cut-cactus (which is still a tree) appears in Figure 3.4.

Next we look at the set of consumers, K, that are connected by an edge to a producer. We wish to enumerate all minimal k- σ cuts. For each consumer $k \in K$, we enumerate the minimum cuts as follows (inspired by the method in [MR 95]):

Step 1 - Find a minimum σ-k cut, C_1.

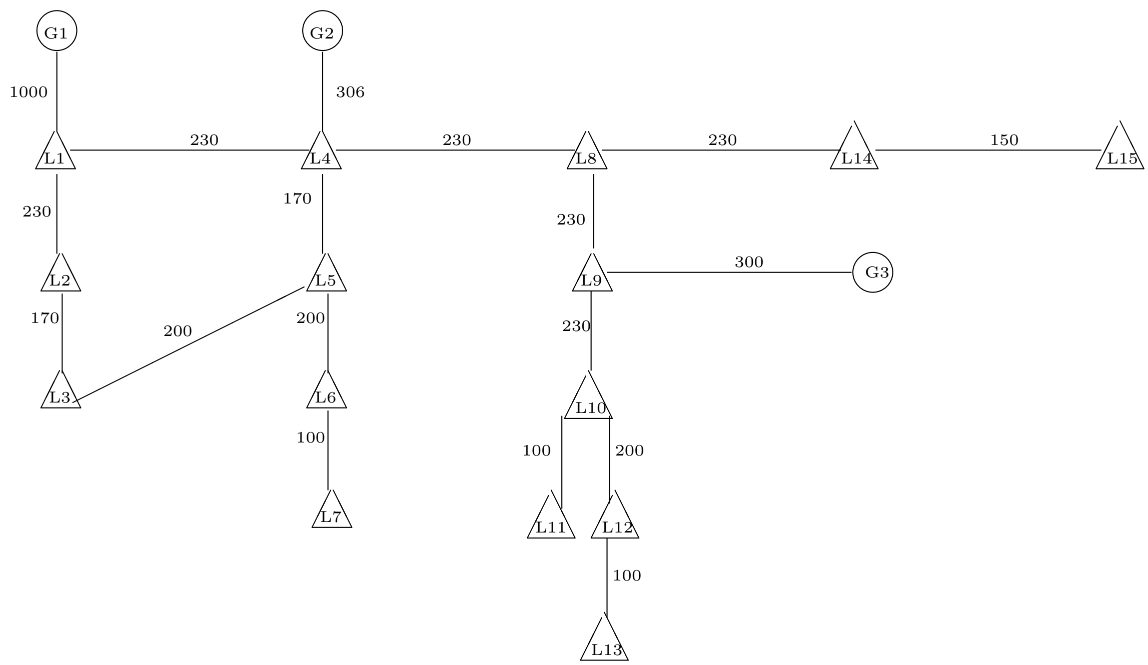


Figure 3.2: A small system network, the circles are producers, the triangles consumers, and the numbers represent line capacity.

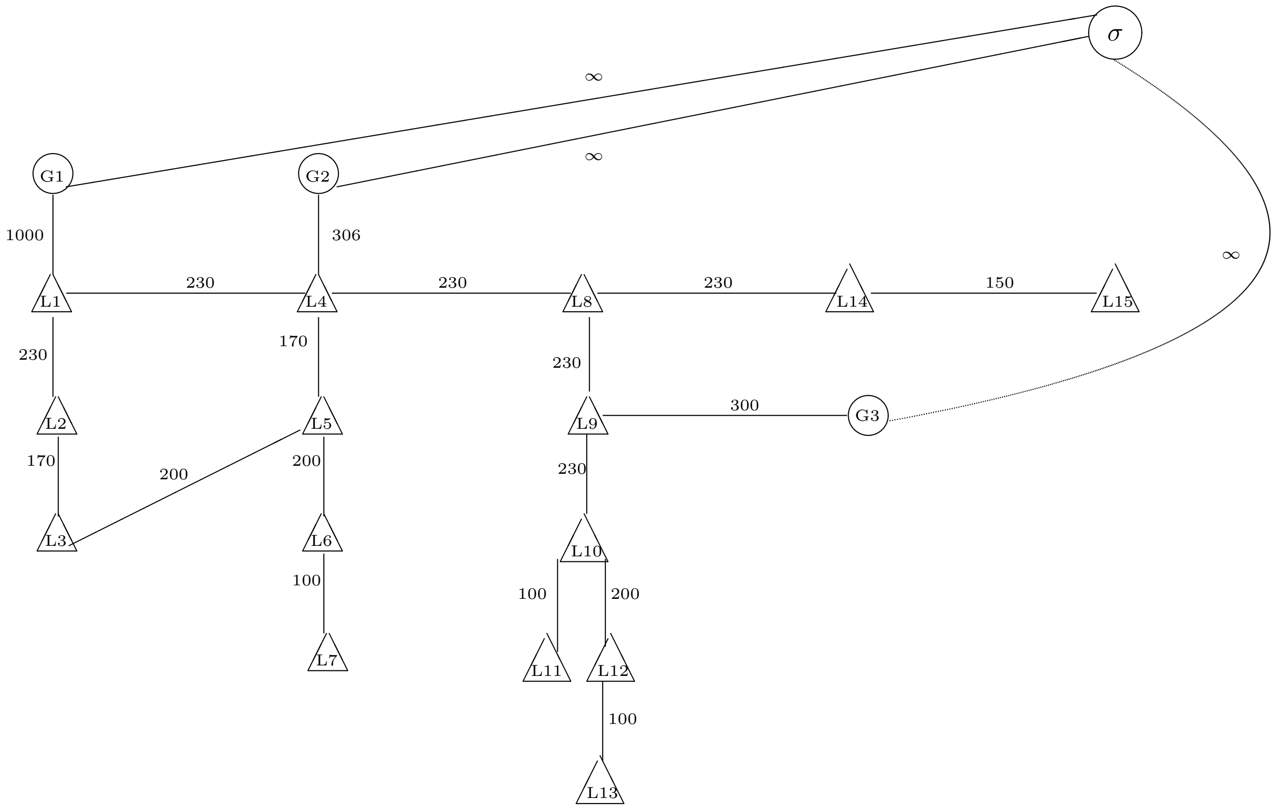


Figure 3.3: Electric network in Figure 3.2 with super-producer, σ

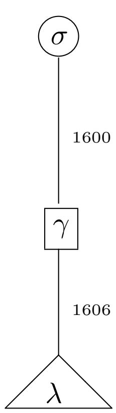


Figure 3.4: Second step in the formation of the cut-cactus

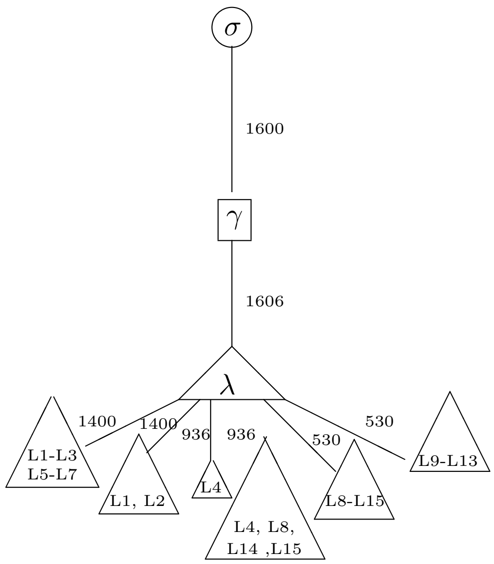


Figure 3.5: Another step in the formation of the cut-cactus

Step 2 - Collapse an edge e_1 in C_1, by creating a new graph, G_1 which has the ends of e_1 as a single node, and all edges that connected to either end is connected to the single node.

Step 3 - Find a minimum σ-k cut, C_2 in the new graph.

Step 4 - If $\mathbf{val}(C_1) = \mathbf{val}(C_2)$, then both C_1 and C_2 are minimum cuts.

Step 5 - Repeat steps 2-4 for every edge in cut C_1

Step 6 - Repeat for each minimum cut found, except do not repeat any of the collapses (i.e. if an edge is in both C_1 and C_2 it is not necessary to collapse that edge twice.)

It should also be noted that enumeration of all minimum cuts is very time consuming for large networks.

For each minimum k-σ cut, $Q = (A, B)$, with $k \in A$ we find, we place an edge (with weight $\mathbf{val}(Q)$) in the cut-cactus. One end of the edge is a node containing A. The other end is the node at the previous step containing nodes with A as a subset. If there is more than one node already in the cactus that contains A, we add a dummy node, draw an edge from the dummy node to A with weight $\mathbf{val}(Q)$ and connect the dummy node to both superset nodes. We now continue this process with the nodes that are a distance of 2 lines from a producer (that is the shortest length between the consumer and any producer is 2 in the unweighted original graph.) We wish to make sure that only minimum cuts are represented, so some consumers may not appear alone in the cactus.

The next step in our cut-cactus formation is shown in Figure 3.5.

Continuing, the final cut-cactus is presented in Figure 3.6.

Note that we could also use a tree representation by just using one edge to represent each cut. These would not be unique, as there would be a choice of representing edge. For example, two trees we could obtain from Figure 3.6 are represented in Figures 3.7 and 3.8.

However, the same information could be obtained from either the tree or cut-cactus version. Another example of a network with a super-producer and its production cut-cactus (which in this

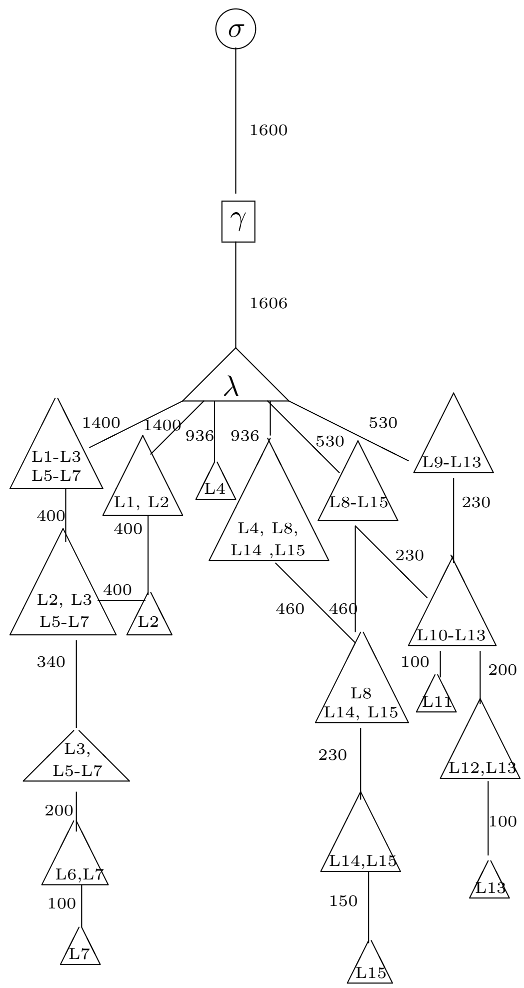


Figure 3.6: The cut-cactus for the electric network in Figure 3.2

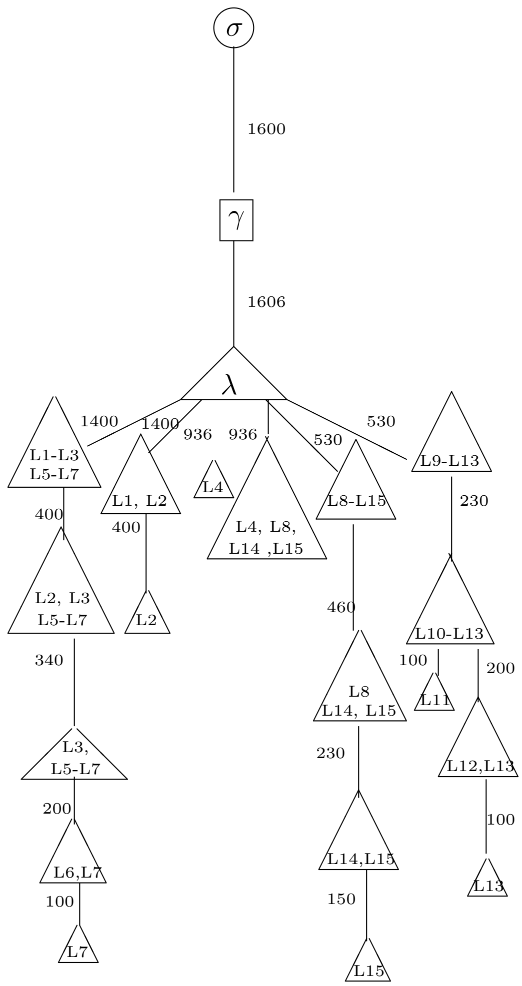


Figure 3.7: One cut tree for the electric network in Figure 3.2.

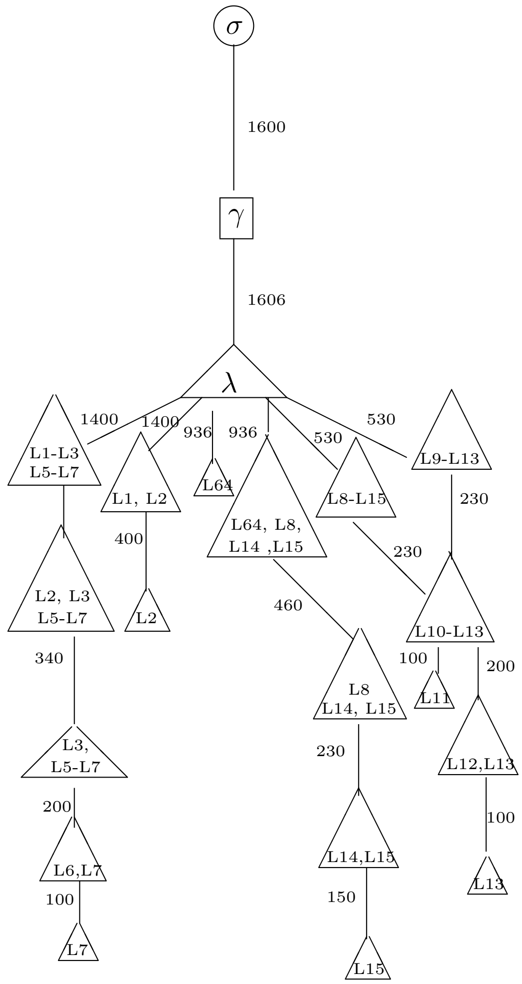


Figure 3.8: Another cut-tree for the electric network in Figure 3.2.

case is neither a tree nor a cactus) are presented in Table 3.9 (total power is positive if power is generated and negative if power is consumed) and Figure 3.10.

Node number	power generated	power consumed	total power
1	700	108	592
2	500	97	403
3	0	180	-180
4	0	74	-74
5	0	71	-71
6	0	136	-136
7	0	125	-125
8	0	171	-171
9	0	175	-175
10	0	195	-195
11	0	0	0
12	0	0	0
13	300	265	45
14	100	50	50
15	600	317	283
16	100	100	0
17	0	0	0
18	400	333	77
19	0	181	-181
20	0	128	-128
21	100	0	100
22	300	0	300
23	660	0	660
24	0	0	0

Figure 3.9: Power generated and consumed by the nodes in Figure 3.10.

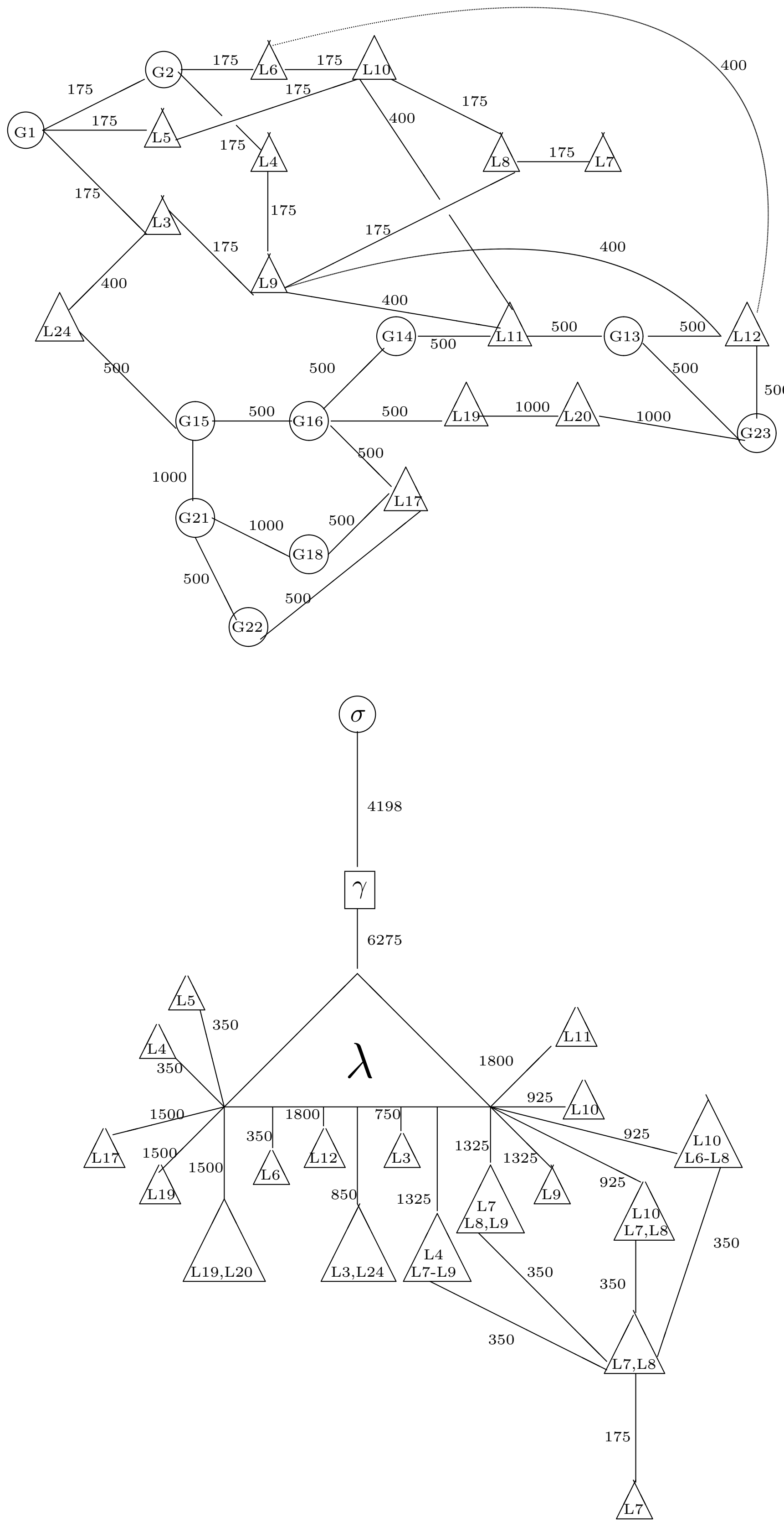


Figure 3.10: Another small electrical network and its cut-cactus

We suggest the following four metrics (κ_1, κ_2, κ_3 and κ_4) as measures of robustness based on the production cut-cactus. Recall that $C \subset V$ is the set of vertices that consume energy. Let $\mathbf{Q} = \bigcup_{l \in C} \mathcal{Q}(V, E, w, \sigma, l)$ be the set of all minimum $\sigma - l$ (for $l \in C$) cuts. Let $k = (A, B) \in \mathbf{Q}$, with $\sigma \in A$. Let $N_k = |B|$. Our first metric, κ_1 is the average of the ratios of the minimum cut to the number of loads separated from the generator for that cut

In equation form, κ_1 is:

$$\kappa_1 = \frac{1}{|\mathcal{Q}|} \sum_{k \in \mathcal{Q}} \left(\frac{\mathbf{val}(k)}{N_k} \right) \tag{3.24}$$

In our two examples, $\kappa_1 = 207.014$ for Figure 3.2 and $\kappa_1 = 772.917$ for Figure 3.10.

For our second metric, κ_2 we consider the ratio of the average minimum cut to the average demand not met, which in equation form is as follows:

$$\kappa_2 = \frac{\frac{1}{|\mathcal{Q}|} \sum_{k \in \mathcal{Q}} \mathbf{val}(k)}{\frac{1}{|\mathcal{Q}|} \sum_{k \in \mathcal{Q}} \left(\sum_{v \in B} c(v) \right)} = \frac{\sum_{k \in \mathcal{Q}} \mathbf{val}(k)}{\sum_{k \in \mathcal{Q}} \sum_{v \in B} c(v)} \tag{3.25}$$

For Figure 3.2, $\kappa_2 = 2.546$, whereas for Figure 3.10 $\kappa_2 = 4.444$. Thus, it takes a more powerful cut to cause loss of demand to a customer in Figure 3.10, which implies that Figure 3.10 is more robust than the network in Figure 3.2.

The third metric, κ_3, is the average amount of extra consumption shed for minimum cut of a consumption site. This is given by the equation:

$$\kappa_3 = \frac{1}{|\mathcal{Q}|} \sum_{v \in B} c(v) \tag{3.26}$$

In Figure 3.2, $\kappa_3 = 188.611$, and in Figure 3.10, $\kappa_3 = 225.333$. Thus, any minimum cut will cause more loss of power for Figure 3.10 than for Figure 3.2.

The final metric κ_4 is described as follows. For each consumer l let $n(l)$ be the number of $\sigma - \ell$ minimum cuts that also separate l from σ.

$$\kappa_4 = \frac{|C|}{\sum n(l)} \tag{3.27}$$

The graph is most robust if $\kappa_4 = 1$. In the example in Figure 3.2 $\kappa_4 = \frac{15}{44}$. In the example in Figure 3.10, $\kappa_4 = \frac{15}{29}$.Using metric κ_4, the second network is more robust than the first.

Robustness and Resilience

The metrics mentioned above are all related to damage sustained or amount of power lost, which would give a measure of robustness. To expand this to include measures of resilience, we would like to incorporate a temporal factor. For each line, generator and consumer node, we can associate a time for repair. This would enable us to measure not only the energy lost for the cut, but also the time it would take to repair the particular cut, giving a measure of resilience. For minimum cuts, we are interested in the smallest cut that requires the longest time to repair (or to restore power).

3.4.4 Area of Influence

We wish to find a single number to measure the area of influence. Let N be a production network (electric or gas). Consider the graph N' where all the edge weights have been replaced by distance rather than capacity. For a given producer G, we wish to find the furthest consumer that can gain power from G, given that G supplies all the nearest customers first. This is easily computed by satisfying the full demand of the closet consumer, then using the residual power to supply the next closest etc. When you reach a consumer, L, where the demand cannot fully be met, the distance from G to L is the radius, R of the area of influence for G. Note that this is unique: if L_1 and L_2 are the same distance, d, from G, then either both demands are met, or they are not. In the first case, we move on so that $R > d$, in the second $R = d$ (assuming we have satisfied all consumers whose distance from G is less than d.

It is interesting to consider how the capacity of the producer will affect the radius of the area of influence. It is possible that a large producer is close to large consumers, or a large producer may be the main power source for many smaller consumers. There may be some insight available from the shape of the scatter plots "distance of influence vs. production capacity".

A robust network will have large radii of influence as this implies that there are many different power solutions. A possible metric for measuring the robustness is: the average radius of area of influence for producers divided by the average production capacity, or the average of the radius of area of influence for producers divided by the production capacity.

In addition, we may consider the area of influence for a consumer, L, rather than a producer. That is, we assume that power comes first from the closest producer, but that the demand for all consumers between L and the producer has been met. A producer is in the area of influence for the consumer L if all consumers along the shortest path from L to G can be satisfied by G and G has some residual capacity to supply power to L.

A possible metric here is: the average radius of area of influence for consumers divided by the average demand, or the average of the radius of area of influence for consumers divided by the demand in that area.

3.4.5 Connected Partitions

Another idea for measuring robustness is considering a partition of the network. A robust network would be able to be partitioned into many subnetworks and still meet the demand. Thus, we now turn to the problem of finding the size, M, of the largest partition of the network where the demand can be satisfied separately for each subnetwork.

We begin with a simple upper bound. Demand in a subnetwork cannot be met if there is no producer in the subnetwork. Hence, the number of producers is an upper bound for M.

To find a lower bound, we again consider the Gomory-Hu cut tree. In this case we are mainly interested in separating producers from producers, so we do not make the complete $s - t$ cut tree, but restrict ourselves to the cuts that only separate producers from producers. Let N be a network.

Step 1: Create the Gomory-Hu cut tree for the producers in N.

Step 2: Consider the subgraphs that arise from the partitions found in Step 1. If all the demands are met within each subnetwork, we are done and M equals the number of producers. If not continue.

Step 3: Consider the subnetwork with the largest unsatisfied demand. Say the producer in this subnetwork is G_1. Find the closest producer, G_2 (in terms of shortest weighted path in the original network). Change the original network by collapsing G_1 and G_2 to obtain N_2

Step 4: Repeat Steps 1-3 with N_2.

Continue until all the demand is met. The number of vertices in the final tree is a lower bound for the number of partitions. Again, since cut trees are not unique, this does not guarantee the same lower bound each time. Of course we could enumerate all the minimum producer cuts and obtain a possibly higher lower bound.

A possible metric related to partitions is $M/|G|$ where M is the size of the largest partition and $|G|$ is the number of producers. The closer this metric is to 1 the more robust the system is.

3.5 Temporal Metrics

A natural reliability metric is the size of cut (or size of impact) vs. time it takes to restore the system. The minimum cut that takes at least time T_0 to restore is given by

$$r(T_0, s, t) = \min_{S \subseteq E} \left\{ \sum_{e \in S} w(e) \middle| \sum_{e \in S} T(e) \geq T_0 \text{ and there is no path from } s \text{ to } t \text{ in } (V, E - S) \right\} \quad (3.28)$$

Here is an algorithm for computing $r(T_0, s, t)$: enumerate all minimum cuts, and then compute their T value, to construct the function r. This procedure would not be efficient, but there may be a generalized network algorithm that can be adapted to solve this problem.

Bibliography

[AKMR97] R. Ahuja, M. Kodialam, A.K. Mishra, and J.B. Orlin. "Computational investigations of maximum flow algorithms" *European Journal of Operational Reseach.* **97** (1997) 509-542.

[HHKY02] S.K. Han, P. Holme, B.J. Kim, and C.N. Yoon. "Attack vulnerability of complex networks" *Physical Review E* **65** (2002) 056109-1 - 056109-14.

[N01] M.E. Newman. "Scientific collaboration networks, II. Shortest paths, weighted networks, and centrality." *Physical Review E.* **64** (2001) 016132-1 - 016132-7.

[MR95] R. Motwani, and P. Raghavan. *Randomized Algorithms* (Cambridge: Cambridge University Press, 1995)

[BBL 94] K. S. Bagga, L. W. Beineke, M. J. Lipman, and R. E. Pippert, "Edge Integrity: a Survey." *Discrete Mathematics* **124** (1994) 3–12.

[BFT 01] B. W. Bush, C. R. Files, and D. R. Thompson, "Empirical Characterization of Infrastructure Networks," Los Alamos National Laboratory, LA-UR-01-5784, 22 October 2001.

[BOS 01] G. Brightwell, G. Oriolo, F. B. Shepherd, "Reserving Resilient Capacity in a Network," *SIAM J. on Discrete Math.* **14** (2001) 524–539.

[CCP 98] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, and A. Schrijver, *Combinatorial Optimization* (New York: Jonh Wiley & Sons, 1998).

[CEA 00] R. Cohen, K. Erez, D. ben-Avraham, and S. Havlin, "Resilience of the Internet to Random Breakdowns." *Physical Review Letters* **85** (2000) 4626–4628.

[DJ 72] P. Doulliez and J. Jamoulle "Transportation networks with random arc capacities," *RAIRO, Rech. Oper. Oper. Res.* **3** (1972) 45–60.

[Fl 98] L. Fleischer, "Building Chain and Cactus Representations of All Minimum Cuts from Hao-Orlin in the Same Asymptotic Run Time," *Lecture Notes in Computer Science* **1412** (1998) 294ff.

[Fo 92] L. R. Foulds, *Graph Theory Applications* (New York: Springer-Verlag, 1992).

[GH 61] R. E. Gomory and T. C. Hu, "Multi-terminal network flows," *SIAM J. Appl. Math.* **9** (1961) 551–570.

[HKY 02] P. Holme, B. J. Kim, C. N. Yoon, and S. K. Han, "Attack Vulnerability of Complex Networks," *Physical Review E* **65** (2002) 6109–6109.

[LKKK 98] V. Lakamraju, Z. Koren, I. Koren, and C. M. Krishna, "Measuring the Vulnerability of Interconnection Networks in Embedded Systems," *Lecture Notes in Computer Science* **1388**(1998) 919-922.

[LJY 95] J. S. Lin, C. C. Jane, and J. X. Yuan, "On Reliability Evaluation of a Capacitated-Flow network in terms of minimal pathsets," *Networks* **25** (1995) 131–138.

[LSS 98] N. Lynn, N. Singpurwalla, and A. Smith, "Bayseian Assessment of Network Reliability," *SIAM Review* **40** (1998) 202–227.

[MR 95] R. Motwani and P. Raghavan. *Ransomized Algorithms* (New York: Cambridge University Press, 1995).

[NERC 02] North American Electric Reliability Council. "The Reliability of Bulk Electric Systems in North America." see www.nerc.com (October 2002).

[Ra 01] D. Rautenbach. "On Vertex Orderings and the Stability Number in Triangle Free Graphs," Discrete Mathematics **231** (2001) 411–420.

[SHL 96] C. R. Schneiter, Y. Y. Haimes, D. Li, and J. H. Lambert,. "Capacity reliability of water distribution networks and optimum rehabilitation decision making," *Water Resources Research* **32** (1996) 2271–2278.

[SL 99] C. T. Su and G. R. Lii, "Reliability planning for composite electric power systems," *Electric Power Systems Research* **51** (1999) 23–31.

[WD 97] K. A. Werley and L. J. Dowell, "A Performance Index for AC Power Flow in Electric Power Transmission Networks," Los Alamos National Laboratory, LANL-97-0087, 4 October 1997.

[ZW 01] S. Zhang and Z. Wang, "Scattering Number in Graphs," *Networks* **37** (2001) 102–106.