Cost-Efficient Algebraic Connectivity Optimisation of Backbone Networks

Mohammed J.F. Alenazi[a,b,*], Egemen K. Çetinkaya[a,d,1], James P.G. Sterbenz[a,c]

[a]*Information and Telecommunication Technology Center, The University of Kansas, Lawrence, KS 66045, USA*
[b]*College of Computer and Information Sciences, Department of Computer Engineering, King Saud University, Riyadh, Saudi Arabia*
[c]*School of Computing and Communications, Lancaster University, Lancaster LA1 4WA, UK*
[d]*Department of Electrical & Computer Engineering, Missouri University of Science and Technology, Rolla, MO 65409, USA*

Abstract

Backbone networks are prone to failures due to targeted attacks or large-scale disasters. Network resilience can be improved by adding new links to increase network connectivity and robustness. However, random link additions without an optimisation objective function can have insignificant connectivity improvement. In this paper, we develop a heuristic algorithm that optimises a network by adding links to achieve a higher network resilience by maximising algebraic connectivity and decreasing total cost via selecting cost-efficient links. We apply our algorithm to five different backbone topologies and measure algebraic connectivity improvement and the cost incurred while adding new links. For evaluation, we apply three centrality node attacks to the non- and optimised networks and show the network flow robustness while nodes are removed. Our results show that optimised graphs with higher algebraic connectivity values are mostly more resilient to centrality-based node attacks.

Keywords:
Network optimisation, heuristic algorithm, resilience, connectivity, robustness, dependability, algebraic connectivity, backbone network

1. Introduction and Motivation

Networks in general, and communication networks in particular, are prone to a variety of challenges and attacks that can have costly consequences. However, network connectivity can be improved with careful planning and optimisation, and the impact of such challenges can be reduced. The design and optimisation of cost-efficient networks that are resilient against challenges and attacks has been studied by many researchers over the past few decades, but the resilient network design problem is NP-hard.

In this paper, we approach resilient network design from a graph theoretic perspective. We develop a *heuristic algorithm* that improves the connectivity of a graph in terms of the *algebraic connectivity* metric by adding links. Algebraic connectivity $a(G)$ is defined as the second smallest eigenvalue of the Laplacian matrix [1] and it is widely used for topological optimisations [2, 3, 4]. A secondary objective of our algorithm is to select the links that improve the algebraic connectivity of the graph in the least costly fashion in which we capture the cost of network as the total link length. Furthermore, we parameterise our optimisation algorithm such that connectivity and cost are weighted depending on a cost-effect parameter γ.

The heuristic to increase algebraic connectivity in a graph is based on adding links to the nodes that have least incident links (i.e. minimal degree nodes) [2, 4]. Our parameterised heuristic algorithm identifies and selects the links that increase the algebraic connectivity of a graph depending on the available budget. Moreover, the search of the best links is computationally less expensive in our algorithm compared to an exhaustive search. We use five commercial service provider physical networks (AT&T, Level 3, Sprint, Internet2, and CORONET) to evaluate our algorithm. Our algorithm provides the cost-efficient new links to improve a network's resilience measured by the algebraic connectivity metric.

The rest of the paper is organised as follows: We present a brief background on network design and optimisation in Section 2. The assumptions, objective functions, and our heuristic algorithm is presented in Section 3. The dataset for the communication networks as well as optimisation evaluation of these topologies using our algorithm is presented in Section 4. The robustness evaluation of the non- and optimised graphs is presented in Section 5. Finally, we summarise our findings as well as propose future work in Section 6.

2. Background and Related Work

Network design and optimisation has been studied in past decades [5, 6, 7, 8, 9, 10, 11, 12, 13, 14] and many problems in this field are considered to be NP-hard [15, 16, 17, 18]. Several monographs provide in-depth coverage of the topic [19, 20, 21, 22]. The design process includes constructing the network from the ground up including placement of nodes [8, 9] and

*Corresponding Author
Email addresses: `malenazi@ittc.ku.edu, mjalenazi@ksu.edu.sa` (Mohammed J.F. Alenazi), `ekc@ittc.ku.edu, ekc@mst.edu` (Egemen K. Çetinkaya), `jpgs@ittc.ku.edu, jpgs@comp.lancs.ac.uk` (James P.G. Sterbenz)

[1]Work performed primarily at The University of Kansas.

providing connectivity among nodes to enable services. The optimisation process includes improvement of the network for one or multiple objectives. Network optimisation can be accomplished by means of rewiring while keeping the number of edges constant [4] or by means of adding new links to improve the connectivity of graphs [2]. Moreover, the design process is different for backbone and access networks, since the topological structure of these networks fundamentally differ [8, 9, 10].

Network design and optimisation objectives are cost, capacity, reliability, and performance [7, 8, 9]. Network cost is incurred by the number of nodes required, capacity of nodes required, and number of links. Previously, we provided a network cost model to add a link between node i and node j as:

$$C_{i,j} = f + v \times d_{i,j} \tag{1}$$

where f is the fixed cost associated with the link (including termination), v is the variable cost per unit distance for the link, and $d_{i,j}$ is the length of the link [23, 24]. Moreover, if we assume that the fibre length dominates *wide-area* network cost and ignore the fixed cost associated with each link, the network cost can be written as:

$$C = \sum_{i} l_i \tag{2}$$

where l_i is the length of the i-th link [25, 26].

Topological connectivity is another objective that can be measured by means of many graph metrics such as average degree, betweenness, closeness, and graph diversity [4, 5, 6, 27, 28, 29]. In this paper, we measure the connectivity of a graph in terms of algebraic connectivity metric. Algebraic connectivity $a(G)$ is defined as the second smallest eigenvalue of the Laplacian matrix [1]. The Laplacian matrix of G is: $L(G) = D(G) - A(G)$ where $D(G)$ is the diagonal matrix of node degrees, $d_{ii} = \deg(v_i)$, and $A(G)$ is the symmetric adjacency matrix with no self-loops. The algebraic connectivity of a complete graph (i.e. full mesh) is n where n is the number of nodes, and it is 0 for a disconnected graph with more than one component.

Topology design using algebraic connectivity has been studied by several researchers [2, 3, 4]. It has been shown that algebraic connectivity is more informative and accurate than average node degree when characterising network resilience [3]. Moreover, we have shown algebraic connectivity [26, 30] and diversity [29] are predictive of flow robustness of graphs. Three synthetically generated topologies (i.e. Watts-Strogatz, Gilbert, Barabási-Albert) have been optimised using edge rewiring in which the objective is to increase the algebraic connectivity [4]. It was shown that algebraic connectivity increases the most if edges are rewired between weakly connected nodes. Another study optimised synthetically generated Erdős-Rényi and Barabási-Albert graphs in terms of adding links to the existing topology [2]. It was concluded that adding links between a low degree node and a random node is computationally less expensive than an exhaustive search. In this paper, we present an algorithm for topological optimisation in terms of adding links, which maximises algebraic connectivity and aims to choose links so that the cost is minimal among given choices.

3. Topology Optimisation Algorithm

In this section, we describe our algorithm that optimises connectivity and cost of a topology. Our heuristic algorithm is implemented using Python. Furthermore, we assume that node locations are given for a graph to apply optimisation algorithm, as would be the case for a deployed service provider.

3.1. Objectives

The objective of this algorithm is to identify the best links to be added to improve the connectivity of the graph. In this paper, we use algebraic connectivity as a measure of connectivity, but we note that any graph connectivity property, such as average node degree, clustering coefficient, or diversity can be used with our heuristic. For example, the clustering coefficient can be used to replace the algebraic connectivity or both the clustering coefficient and the algebraic connectivity can be used with a tuning parameter to control their effect in selecting the links.

3.2. Algorithm

The topology optimisation algorithm has three inputs: an input graph G_i, a number of required links L_r, and a cost-effect parameter γ. The input graph G_i has a number of nodes n_i with a number of links l_i. The number of required links L_r is the number of links that should be added to the graph. The cost-effect parameter γ is a tuning parameter between cost and algebraic connectivity. When $\gamma = 0$, the cost term of the rank function is neglected since it is zeroed. As a result, the algorithm selects the link that maximises the algebraic connectivity. On the other hand, when $\gamma = 1$, the algebraic connectivity is neglected and the least link cost is selected in each iteration. The algorithm adds links to the graph with L_r iterations. To keep track of the selected links in each iteration, the algorithm adds these links to a list. In each iteration, the algorithm starts by adding the selected links from previous iterations to the graph. Then, the rank value is computed for each candidate link and the link with the maximum rank value is selected to be added. A ranking function is used to select the best candidate in each iteration. The rank value r is computed using:

$$r = (1 - \gamma)a(G) + \gamma(1 - C) \tag{3}$$

where C represents the length of the ranked link. This algorithm uses four functions: cost function $\mathsf{cost}(L)$, algebraic connectivity function $\mathsf{algConn}(G)$, $\mathsf{maxLink}(D)$, and $\mathsf{candidate}(G)$. The cost function $\mathsf{cost}(L)$ returns the cost of adding a link L. In this paper, the cost is defined as the Euclidean distance between the two ends of the link. The algebraic connectivity function $\mathsf{algConn}(G)$ takes a graph G and returns the second smallest eigenvalue of its Laplacian matrix. The $\mathsf{maxLink}(D)$ function returns the maximum ranked link. The $\mathsf{candidate}(G)$ takes a graph G as input and returns a set of candidate links to be added to the graph. The candidate links are a set of links that are examined every time a link is added to a graph. One option to use for the candidate links is the set of complement links of a graph is denoted as $\bar{G}$, which can be determined as the set of links in full mesh subtracted from the current links in a graph G. The number of complement links (cf. shown in column 4 Table 2) is computed as:

$$\frac{n_i(n_i-1)}{2} - l_i \tag{4}$$

However, this number is computationally expensive as the number of nodes n_i gets larger, which results in a complexity of $O(L_r n_i^2)$. In an attempt to decrease the number of candidate links, we only examine the links connected to the lowest degree node in the graph. As a result, the algorithm complexity decreases to $O(L_r n_i)$.

Both the $\mathsf{algConn}(G)$ and $\mathsf{cost}(L)$ functions are normalised to have a maximum value of one. Since the theoretical maximum value for the algebraic connectivity of a given graph is the number of its nodes, it is normalised by dividing it by the number of nodes. To normalise the cost function, it is divided by the maximum possible distance between any nodes in the graph. The pseudocode of our algorithm is shown in Algorithm 1.

```
Functions:
cost(L) := cost function
algConn(G) := algebraic connectivity function
candidate(G) := candidate links function
maxLink(D) := max value of a dictionary
Input:
G_i := input graph
L_r := number of required links
γ := cost effect parameter
Output:
an ordered list of the added links
begin
    selectedLinks = []; empty ordered list
    rank = {}; empty dictionary
    while L_r > 0 do
        G = G_i
        G.addlinks(AddedLinks)
        for link in candidate(G) do
            rank[link]=(1−γ)algConn(G)+γ(1−cost(link))
        end
        selectedLinks.add(maxLink(rank))
        L_r = L_r − 1
    end
    return selectedLinks
end
```

Algorithm 1: Topology optimisation algorithm

3.3. Algorithm options

In this paper, we consider physical networks. Hence, we have added an option that removes very long links from the candidate link set. This is because it is not practical to add very long links between cities such as a physical fibre link between Los Angeles and New York City. Therefore this raises the question of what the maximum length should be chosen. In our implementation, we have it as a variable that can be set by the user. We choose the maximum length link in the input graph to be the threshold for long links in the dataset, which gives a good indicator for the maximum link length a provider can afford.

4. Analysis

In this section, first we describe our algorithm on a small size graph. Next, we present the topological dataset we use to apply our algorithm, followed by cost and connectivity analysis of commercial backbone providers.

4.1. Algorithm Evaluation

In this section, we explain how our heuristic algorithm optimises a topology on a small-size graph. Figure 1 shows a sample graph with 8 nodes and 9 links as solid lines. The initial algebraic connectivity of this sample graph is 0.3432 and the initial cost (i.e., total link length in km) of the graph is 8,203. Our heuristic algorithm adds links to the *least* connected nodes, which in the example are nodes 0 and 7. The six candidate links for node 0 are shown as square dots, whereas five candidate links for node 7 are shown as long dashes and dots. Throughout this example, we describe how our algorithm operates if we are going to add *one* link $L_r = 1$ to the sample graph shown in Figure 1.

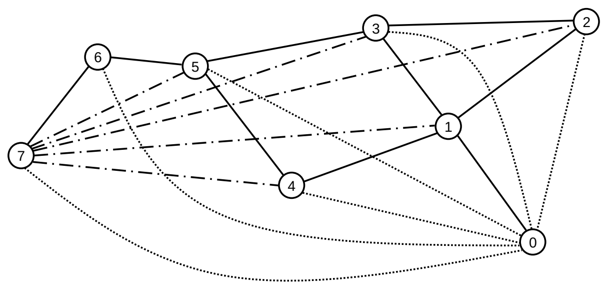


Figure 1: Sample graph

There can be a maximum of 28 links in this 8-node graph (maximum links can be calculated by $\frac{n(n-1)}{2}$). Since there are 9 links in the graph, if we were to do an exhaustive search, there would be $28 - 9 = 19$ candidate links (i.e. the complement links). In the sample graph shown in Figure 1, there are six candidate links that can be added to node 0 and there are five links for node 7 using our heuristic algorithm. Therefore, the candidate link set is reduced to 11, because our algorithm only considers candidate links from the least connected nodes. The algebraic connectivity and cost value of adding each link individually for $\gamma = 0$ and $\gamma = 1$ is shown in Table 1.

When $\gamma = 0$, our algorithm ignores the cost associated with adding a link and selects the additional link that increases the algebraic connectivity of the graph the most. For $\gamma = 0$, the algorithm adds the link between node 1 and 7 in the example graph since it provides the highest algebraic connectivity among the 11 candidate links. When $\gamma = 1$, the cost is the dominant factor

Table 1: $a(G)$ and cost values for the sample graph

Link	**$\gamma = 0$**		**$\gamma = 1$**	
	$a(G)$	**$\Delta a(G)$**	**cost**	**Δ cost**
$0 \leftrightarrow 2$	0.3485	0.0053	**9,275**	**1,072**
$0 \leftrightarrow 3$	0.3588	0.0156	9,405	1,202
$0 \leftrightarrow 4$	0.3659	0.0227	9,848	1,645
$0 \leftrightarrow 5$	0.4079	0.0647	10,624	2,421
$0 \leftrightarrow 6$	0.5908	0.2476	11,228	3,025
$0 \leftrightarrow 7$	0.7713	0.4281	11,843	3,640
$7 \leftrightarrow 1$	**0.8345**	**0.4913**	11,302	3,099
$7 \leftrightarrow 2$	0.7071	0.3639	12,061	3,858
$7 \leftrightarrow 3$	0.6651	0.3219	10,915	2,712
$7 \leftrightarrow 4$	0.5918	0.2486	10,207	2,004
$7 \leftrightarrow 5$	0.5075	0.1643	9,463	1,260

determining the addition of a link. Therefore, our heuristic algorithm selects the link between node 0 and 2, since it incurs the lowest cost among the candidate set of links. The selection of links via our heuristic algorithm is highlighted bold in Table 1. Moreover, we performed an exhaustive search on the sample graph shown in Figure 1, and find that the link between node 1 and 7 has the highest algebraic connectivity among 19 possible links. The result of the exhaustive search for the least incurred cost link indicated that the link between node 3 and 4 is the best option, however, as mentioned above, our algorithm adds links to the minimal degree nodes. Therefore our algorithm selects the link between node 0 and 2 when $\gamma = 1$.

4.2. Topological Dataset

We study physical level topologies of five service provider networks. Among the five provider networks we study, the AT&T [31], Level 3 [32], and Sprint [33] are the commercial backbone providers. The Internet2 [34] is a research network whereas CORONET is a synthetic fiber topology [35, 36]. Selection of five different providers with different network size and order graphs demonstrates the applicability of our algorithm. Physical-level topologies of the five service providers were constructed using a third party map [37]. The details of generating physical-level topologies are presented in our previous work [25, 30].

Table 2: Physical Topological dataset

Network	**Nodes**	**Links**	**Complement links**
AT&T	383	488	72,665
Level 3	99	132	4,719
Sprint	264	313	34,403
Internet2	57	65	1,531
CORONET	75	99	2,676

4.3. Backbone Provider Network Analysis

Our algorithm is applied to five ISPs by adding 100 physical links. We show the graph algebraic connectivity and the cost incurred in terms of meters after adding each link. Moreover, we show the relation of cost and algebraic connectivity and the slope in these figures shows how the cost increases as the graph connectivity improves.

4.3.1. Selection of γ values

The γ parameter that ranges 0 to 1 controls the outcome of the algorithm as described in Section 3.2. In Equation 3, we have two terms: $(1-\gamma)a(G)$ and $\gamma(1-C)$. The $a(G)$ is the *normalised* algebraic connectivity value, which is low for sparse graphs and one for a full mesh graph. The value of C denotes the normalised cost of adding a link and it is low when the maximum possible link length in the input graph is larger than the average link length in the candidate set. Therefore, choosing the value of γ depends on the initial properties of the input graph. For these physical graphs, we choose $\gamma = \{0, 10^{-9}, 10^{-7}, 10^{-5}, 1\}$ because the cost term is larger than the γ term about six order of magnitude for physical level graphs.

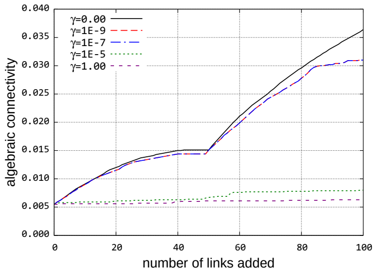
4.3.2. Physical level topology analysis

As explained in Section 3, an option is added in our heuristic algorithm to discard the links that are longer than the actual maximum link of the graph. The connectivity and cost optimisation of physical level provider graphs are shown in Figure 2. Algebraic connectivity improvement of the five physical level topologies after adding 100 links iteratively is depicted in Figures 2a, 2d, 2g, 2j, and 2m. The algebraic connectivity is higher for $\gamma = 0$ than the other values of γ, and for $\gamma = 1$ our algorithm considers minimising the cost, but not improving the algebraic connectivity. Moreover, we observe the occurrence of possible phase transition when $\gamma = 1$ for the physical-level graphs. For example, algebraic connectivity improvement of the AT&T physical topology starts with a moderate increase, and after about 50th link addition, the improvement (i.e., the slope of the curve) gets steeper. Moreover, the algebraic connectivity improvement of CORONET topology has two phase changes at 35th and 79th links. The reasons for the occurrence of this phenomenon will be the subject of future work.

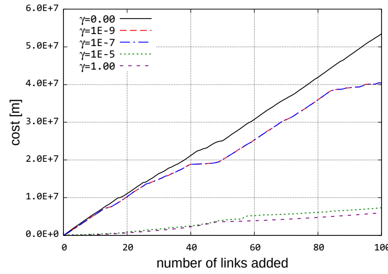
The cost incurred when adding 100 links iteratively to the physical level topologies are shown in Figures 2b, 2e, 2h, 2k, and 2n. The cost in physical topology is the length of links to be laid between nodes, thus, short links are favorable in physical level topology optimisation for $\gamma = 1$. The relationship between connectivity and cost for physical level topologies are shown in Figure 2c, 2f, 2i, 2l, and 2o. For the Level 3 example shown in Figure 2f, if the cost is the constraint (i.e. $\gamma = 1$), the designer can improve the algebraic connectivity to 0.05 by adding 100 links. On the other hand, if there is available budget (i.e. $\gamma = 0$) the algebraic connectivity of the Level 3 topology can be improved more than 0.3.

4.3.3. Optimisation comparison of backbone networks

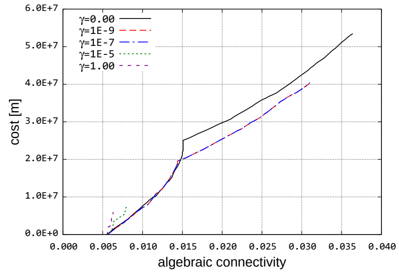
Finally, we compare the optimisation output of the five backbone provider topologies using $\gamma = 0$ and $\gamma = 1$ as shown in



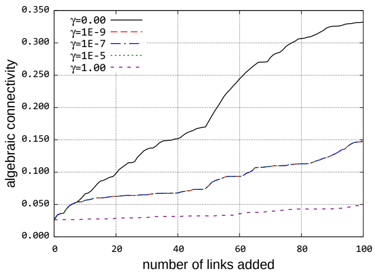
(a) AT&T connectivity improvement



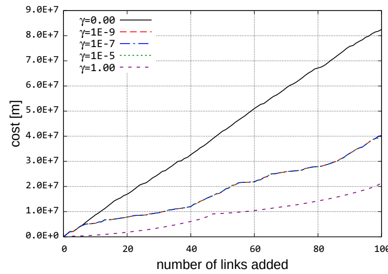
(b) AT&T cost incurred with adding links



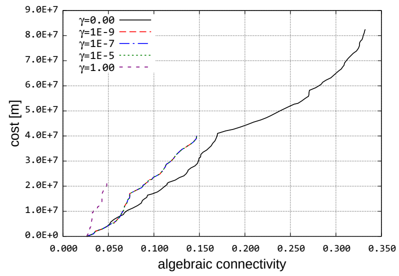
(c) Connectivity and cost for AT&T



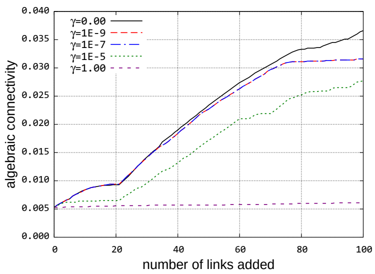
(d) Level 3 connectivity improvement



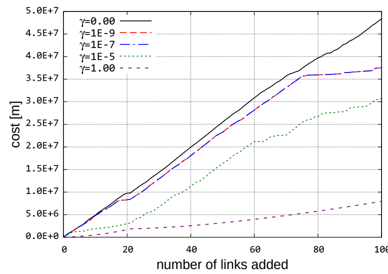
(e) Level 3 cost incurred with adding links



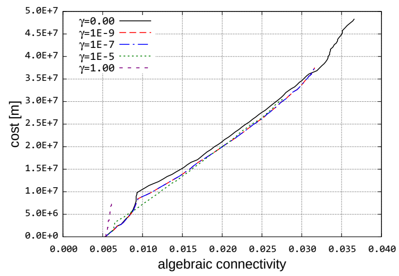
(f) Connectivity and cost for Level 3



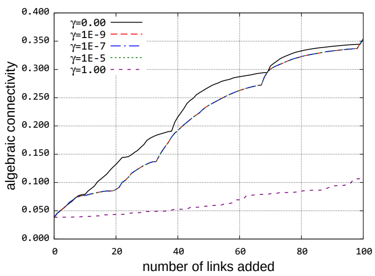
(g) Sprint connectivity improvement



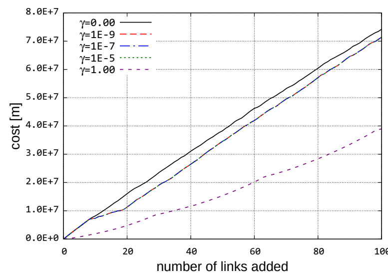
(h) Sprint cost incurred with adding links



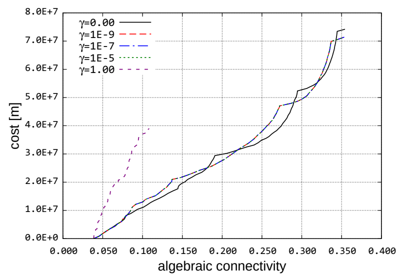
(i) Connectivity and cost for Sprint



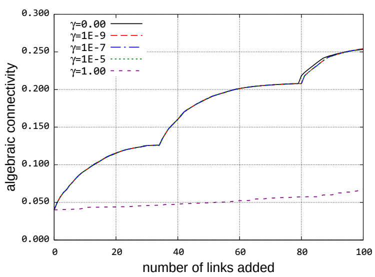
(j) Internet2 connectivity improvement



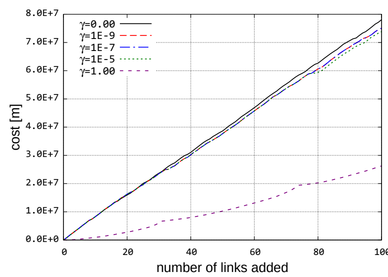
(k) Internet2 cost incurred with adding links



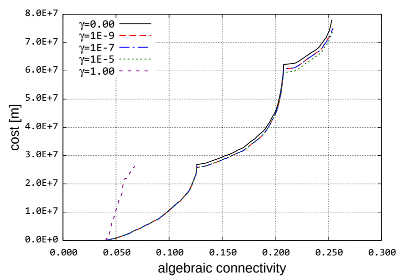
(l) Connectivity and cost for Internet2



(m) CORONET connectivity improvement

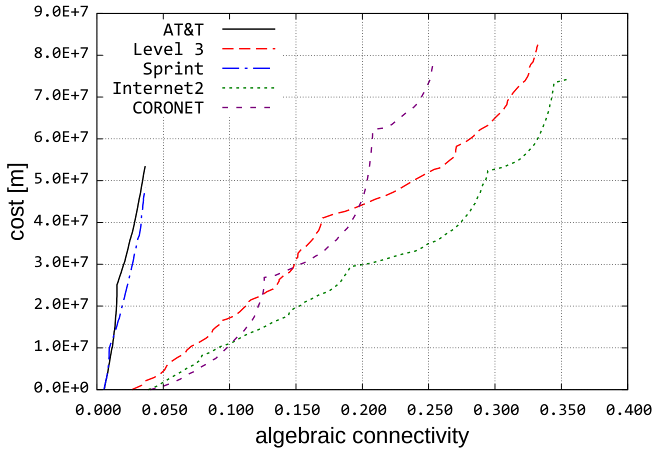


(n) CORONET cost incurred with adding links

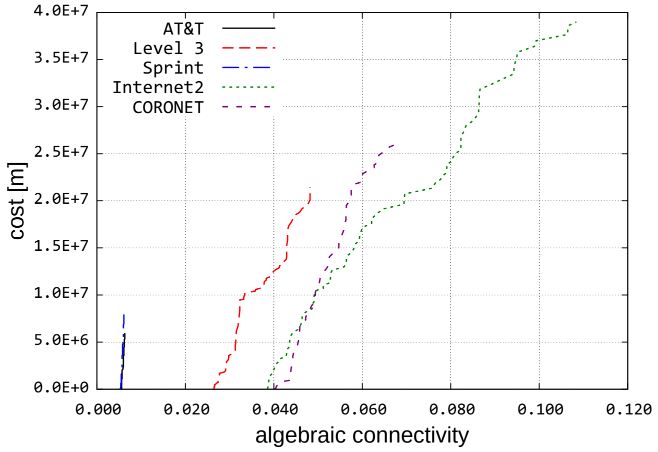


(o) Connectivity and cost for CORONET

Figure 2: Analysis of physical topologies



(a) $\gamma = 0$



(b) for $\gamma = 1$

Figure 3: Algebraic connectivity and cost effect

Figure 3a and Figure 3b, respectively. Sprint and AT&T physical level topologies have similar results using $\gamma = 0$ with about the same algebraic connectivity as when we add 100 links as shown in Figure 3a. This is because both Sprint and AT&T have a relatively large number of nodes compared to the other providers. Hence, they need more links than a smaller graph to achieve the same algebraic connectivity. On the other hand, the Level 3, Internet2, and CORONET physical level topologies start from an even higher initial algebraic connectivity and significantly improve to a higher algebraic connectivity with larger cost than the others as shown in Figure 3a. These three physical topologies have smaller number of nodes, therefore addition of 100 links significantly change the algebraic connectivity of the graph. Similar conclusions can also be drawn when we compare our optimisation algorithm output for $\gamma = 1$ as shown in Figure 3b. The main difference is that when there is no budget constraint (i.e. $\gamma = 0$), the algebraic connectivity and cost is an order of magnitude higher. As a result, we can see that with the same number of link added, smaller graphs gain higher algebraic connectivity improvement.

5. Robustness Evaluation

In this section, we present the set of attacks used to evaluate the robustness of the resulting optimised and non-optimised topologies. Moreover, we study the resilience of optimised and non-optimised topologies in terms of flow robustness graph metric.

5.1. *Flow robustness*

Flow robustness [29, 38] is a graph metric that measures the ratio of possible number of pair-connections, to the maximum number of pair-connections, $n(n-1)$. If the graph is partitioned, the possible number of pair-connections is the sum of $n(n-1)$ connections for each component. The range of flow robustness value is [0,1], and the flow robustness is 1 if the graph is not partitioned and 0 if the graph has no links.

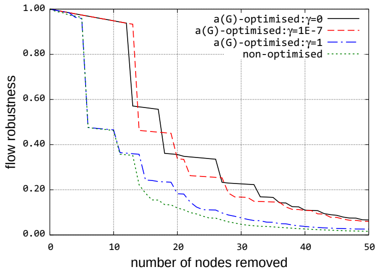
5.2. *Graph centrality attacks*

We use a graph theoretic model to attack a given graph and show how its flow robustness changes after each node removal. In this paper, we use three centrality metrics: node betweenness, node closeness, and node degree [39]. Thus, we have three attack models, in which the node with the highest centrality is removed. The node betweenness attack targets the node through which the highest number of shortest paths pass. The node closeness attack targets the closest node to all the other nodes in terms of hop count. The highest degree node attack targets the node with the highest number of neighbours. The list of removed nodes is determined adaptively for each attack model. This means the node centrality values are calculated after each node is removed and the new highest is selected to be the next node to be removed. This is done repeatedly until all nodes are selected. The adaptive removal of nodes gives a more correct selection for the highest centrality than the non-adaptive removal, in which the highest targeted number of nodes are selected based on a single evaluation [26, 40].

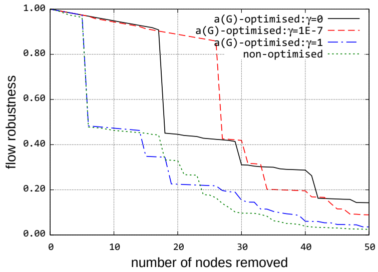
5.3. *Robustness evaluation results*

In this section, we show the results of applying the graph centrality attacks to non- and optimised graphs by removing 50 nodes from each graph. From the optimised graphs, we select the graphs generated using $\gamma = \{0, 10^{-7}, 1\}$ since they represent lowest, middle, and highest γ values. The optimised graphs when $\gamma = 0$ are expected to be the most resilient since new links are selected purely to improve algebraic connectivity with no cost consideration. The optimised graphs when $\gamma = 10^{-7}$ are expected to be the second most resilient graphs since new links are selected to improve algebraic connectivity while favoring least cost links with a threshold related to 10^{-7}. The optimised graphs when $\gamma = 1$ are expected to be the least resilient graphs since new links are selected to purely decrease the total cost.

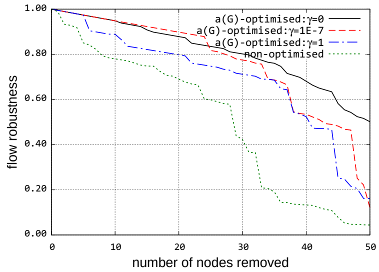
The results of applying the attacks are depicted in Figure 4. The node betweenness attack is consistently the most destructive since flow robustness decreases faster than the other two centrality attacks as shown in Figures 4a, 4d, 4g, 4j, and 4m. The second most destructive centrality attack among the three



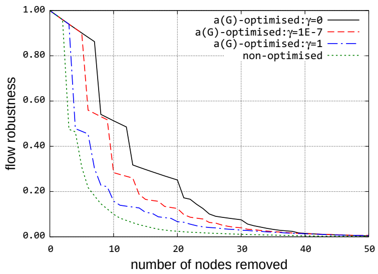
(a) AT&T betweenness-based attack



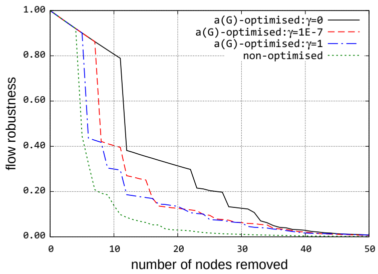
(b) AT&T closeness-based attack



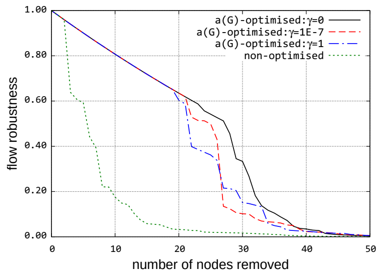
(c) AT&T degree-based attack



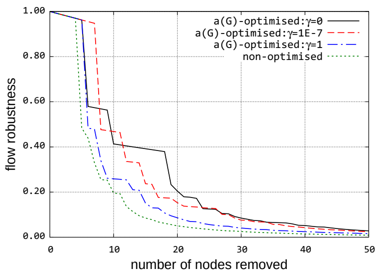
(d) Level 3 betweenness-based attack



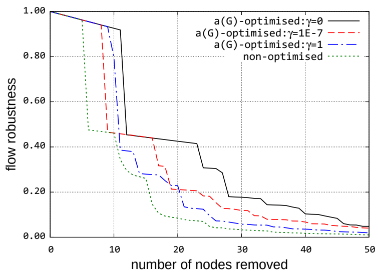
(e) Level 3 closeness-based attack



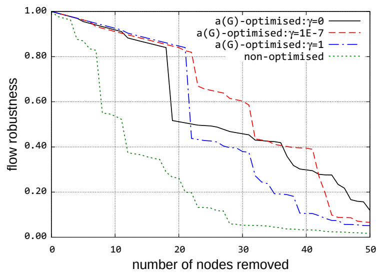
(f) Level 3 degree-based attack



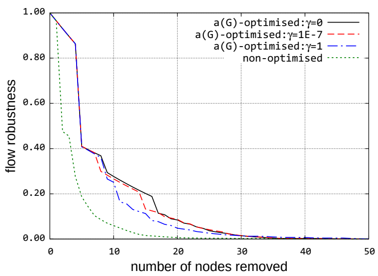
(g) Sprint betweenness-based attack



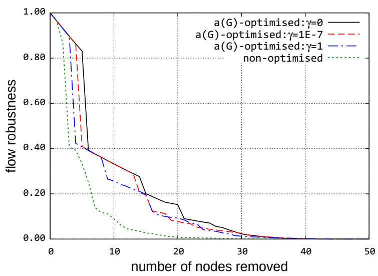
(h) Sprint closeness-based attack



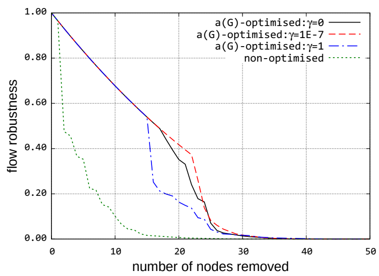
(i) Sprint degree-based attack



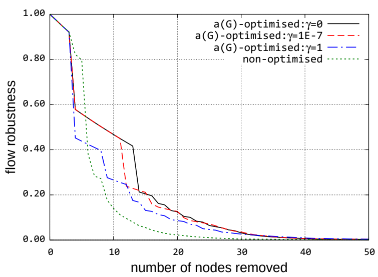
(j) Internet2 betweenness-based attack



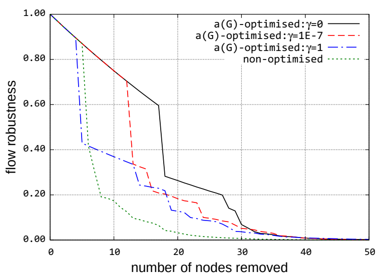
(k) Internet2 closeness-based attack



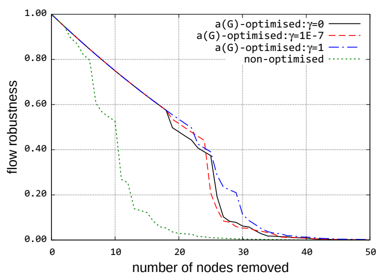
(l) Internet2 degree-based attack



(m) CORONET betweenness-based attack



(n) CORONET closeness-based attack



(o) CORONET degree-based attack

Figure 4: Flow robustness analysis of optimised and non-optimised topologies

Table 3: Sum of flow robustness

Provider	Optimisation Method	Betweenness attack	Closeness attack	Degree attack
AT&T	non-optimised	11.70	15.15	25.66
	$a(G)$-optimised: $\gamma = 0$	21.90	27.32	41.34
	$a(G)$-optimised: $\gamma = 1^{-7}$	21.14	30.31	38.89
	$a(G)$-optimised: $\gamma = 1$	13.01	15.60	35.55
Level 3	non-optimised	5.68	7.15	7.30
	$a(G)$-optimised: $\gamma = 0$	13.81	16.58	23.32
	$a(G)$-optimised: $\gamma = 1^{-7}$	10.96	12.22	21.33
	$a(G)$-optimised: $\gamma = 1$	7.96	10.61	21.05
Sprint	non-optimised	8.46	11.10	14.31
	$a(G)$-optimised: $\gamma = 0$	14.50	20.64	29.39
	$a(G)$-optimised: $\gamma = 1^{-7}$	13.87	16.38	31.68
	$a(G)$-optimised: $\gamma = 1$	10.41	15.10	26.75
Internet2	non-optimised	4.09	5.00	4.71
	$a(G)$-optimised: $\gamma = 0$	8.98	10.20	15.47
	$a(G)$-optimised: $\gamma = 1^{-7}$	8.74	9.23	15.94
	$a(G)$-optimised: $\gamma = 1$	8.12	8.37	13.84
CORONET	non-optimised	7.43	7.84	9.87
	$a(G)$-optimised: $\gamma = 0$	10.82	17.23	18.60
	$a(G)$-optimised: $\gamma = 1^{-7}$	10.39	14.38	18.61
	$a(G)$-optimised: $\gamma = 1$	8.70	10.62	19.60

is the closeness attack since it shows higher impact on the flow robustness. The least destructive attack is the degree attack as it has the lowest impact on flow robustness. Table 3 shows the sum of flow robustness values, which represent the area under the flow robustness curve after the 50 nodes are removed. The $a(G)$-optimised graphs are more resilient than non-optimised graphs because they have 100 additional links.

For AT&T non- and optimised graphs, the results of applying three centrality attacks are shown in Figures 4a, 4b, and 4c. For the betweenness attack on AT&T non- and optimised graphs, we observe that the $a(G)$-optimised graph when $\gamma = 0$ has the highest sum of flow robustness of 21.90 as shown in Table 3. Next, the $a(G)$-optimised graph when $\gamma = 10^{-7}$ graph comes second in terms of flow robustness with insignificant difference of 21.14. Optimised graphs when $\gamma = 1$ and non-optimised graphs have the lowest flow robustness values of 13.01 and 11.70 respectively. For the closeness attack, the $a(G)$-optimised graph when $\gamma = 10^{-7}$ graph has the highest flow robustness value of 30.31 and the second highest is when $\gamma = 0$ with a flow robustness sum of 27.32. Similar to betweenness attack results, optimised graphs when $\gamma = 1$ and non-optimised graphs have the lowest flow robustness values of 15.60 and 15.15 respectively. For the degree attack, the $a(G)$-optimised graph when $\gamma = 0$ graph has the highest flow robustness value of 41.34 and the second highest is when $\gamma = 10^{-7}$ with a flow robustness sum of 38.89. Similar to betweenness attack results, optimised graphs when $\gamma = 1$ and non-optimised graphs have the lowest flow robustness values of 15.60 and 15.15 respectively. Given the previous flow robustness for all the attacks on AT&T non- and optimised graphs, we see that $a(G)$-optimised graph when $\gamma = 0$ is mostly more resilient than the other γ values. However, in some cases, the $\gamma = 10^{-7}$ yields very similar flow robustness results to the graphs optimised using $\gamma = 0$. In these cases, the costs associated adding links should be considered and based on the user optimisation objective, the feasible graphs can be selected. For example, for the betweenness attack on AT&T, the flow robustness sums difference between $\gamma = 0$ and $\gamma = 10^{-7}$ is $21.90 - 21.14 = 0.76$, which is insignificant. On the other hand, the cost difference between the two is about $5.3 \times 10^7 - 4.0 \times 10^7 = 1.3 \times 10^7$ m, which is significantly high cost. At this point, it depends on the user to decide that if this additional flow robustness is worth 1.3×10^7 m.

Using the same method, we study the flow robustness values for the other providers non- and optimised graphs for each centrality attack presented in Table 3. From these results, we can see very clearly the same pattern in AT&T non- and optimised graphs. The graphs optimised using $\gamma = 0$ are the most resilient to any centrality attacks for the examined physical graphs. This is not always the case, we have four cases where $\gamma = 10^{-7}$ yields flow robustness sums than $\gamma = 0$. The first case happens in the

AT&T graphs with closeness attack and the other three cases happen for the degree attack on the three providers: Sprint, Internet2, and CORONET. By looking at the corresponding algebraic connectivity values for each case, we see that the algebraic connectivity values are higher for $\gamma = 0$ even though $\gamma = 10^{-7}$ have higher flow robustness sums for these cases. However, for the other 11 cases the flow robustness sums are higher for graphs with higher algebraic connectivity values.

6. Conclusions and Future Work

Network design and optimisation is a major area of research. In this paper, we present a new heuristic algorithm that optimises the connectivity of a given graph with node locations. We use algebraic connectivity as a measure to improve the connectivity of the graph. This algorithm minimises the cost of adding new links by selecting shorter links with high algebraic connectivity. We introduce a tuning parameter γ to control the effect of the cost function while selecting new links. Furthermore, the candidate links that are being added to improve the connectivity of the graph can be constrained by a length limit in our algorithm. We apply this algorithm to physical-level topologies of the five backbone providers. The results show trade-offs between improving algebraic connectivity and minimising cost, from which a cost-efficient set of link addition can be chosen based on the value of γ. Moreover, we apply centrality attacks on the non- and optimised graphs and study their resilience in terms of flow robustness. We show that graphs with higher algebraic connectivities have mostly higher flow robustness values, which means that they are more resilient.

For our future work, we would like to run our heuristic algorithm using graph properties such as clustering coefficient, betweenness, and graph diversity [29]. For example, we can add clustering coefficient to replace the algebraic connectivity or we can add both with a parameter to weight their effect in ranking the links that need to be added to improve connectivity of the graph. We will also modify our algorithm to achieve a specified graph metric value with a constrained budget. Finally, we will investigate the occurrence of a phase transition phenomenon in physical-level topologies and compare backbone provider graphs with synthetic topologies.

Acknowledgments

We would like to acknowledge Dongsheng Zhang and Yufei Cheng for discussions on this work. This research was supported in part by NSF grant CNS-1219028 (Resilient Network Design for Massive Failures and Attacks) and by NSF grant CNS-1050226 (Multilayer Network Resilience Analysis and Experimentation on GENI). Mohammed J.F. Alenazi is supported by SACM (Saudi Arabian Cultural Mission). This is a significantly extended version and substantial revision of the paper that appeared in the 5th IEEE/IFIP International Workshop on Reliable Networks Design and Modeling (RNDM) 2013 [41]. We would like to acknowledge members of the ResiliNets group for discussions on this work.

References

[1] M. Fiedler, Algebraic connectivity of graphs, Czechoslovak Mathematical Journal 23 (2) (1973) 298–305.

[2] H. Wang, P. Van Mieghem, Algebraic connectivity optimization via link addition, in: Proceedings of the 3rd ICST International Conference on Bio-Inspired Models of Network, Information and Computing Sytems (BIONETICS), Hyogo, Japan, 2008, pp. 22:1–22:8.

[3] W. Liu, H. Sirisena, K. Pawlikowski, A. McInnes, Utility of algebraic connectivity metric in topology design of survivable networks, in: Proceedings of the 7th IEEE International Workshop on Design of Reliable Communication Networks (DRCN), Washington, DC, 2009, pp. 131–138.

[4] A. Sydney, C. Scoglio, D. Gruenbacher, Optimizing algebraic connectivity by edge rewiring, Applied Mathematics and Computation 219 (10) (2013) 5465–5479.

[5] R. S. Wilkov, Analysis and design of reliable computer networks, IEEE Transactions on Communications 20 (3) (1972) 660–678.

[6] J. M. McQuillan, Graph theory applied to optimal connectivity in computer networks, ACM SIGCOMM Comput. Commun. Rev. 7 (2) (1977) 13–41.

[7] M. Gerla, L. Kleinrock, On the topological design of distributed computer networks, IEEE Transactions on Communications 25 (1) (1977) 48–60.

[8] R. R. Boorstyn, H. Frank, Large-scale network topological optimization, IEEE Transactions on Communications 25 (1) (1977) 29–47.

[9] J. G. Klincewicz, Hub location in backbone/tributary network design: a review, Location Science 6 (1–4) (1998) 307–335.

[10] B. Gavish, Topological design of computer communication networks – the overall design problem, European Journal of Operational Research 58 (2) (1992) 149–172.

[11] A. Konak, A. E. Smith, Network reliability optimization, in: M. G. Resende, P. M. Pardalos (Eds.), Handbook of Optimization in Telecommunications, Springer US, 2006, pp. 735–760.

[12] H. C. Cankaya, A. Lardies, G. W. Ester, Network Design Pptimization from an Availability Perspective, in: Proceedings of the 11th International Telecommunications Network Strategy and Planning Symposium (NETWORKS), Vienna, 2004, pp. 359–364.

[13] S. Chamberland, M. St-Hilaire, S. Pierre, On the point-of-presence optimization problem in IP networks, Canadian Journal of Electrical and Computer Engineering 30 (3) (2005) 137–143.

[14] D. Fay, H. Haddadi, A. Thomason, A. Moore, R. Mortier, A. Jamakovic, S. Uhlig, M. Rio, Weighted Spectral Distribution for Internet Topology Analysis: Theory and Applications, IEEE/ACM Transactions on Networking 18 (1) (2010) 164–176.

[15] M. O. Ball, Complexity of network reliability computations, Networks 10 (2) (1980) 153–165.

[16] M. O. Ball, Computational complexity of network reliability analysis: An overview, IEEE Transactions on Reliability 35 (3) (1986) 230–239.

[17] S. Khuller, B. Raghavachari, Graph and Network Algorithms, ACM Comput. Surv. 28 (1) (1996) 43–45.

[18] H. Noltemeier, H.-C. Wirth, S. O. Krumke, Network Design and Improvement, ACM Comput. Surv. 31 (3es) (1999) 1–5.

[19] C. H. Papadimitriou, K. Steiglitz, Combinatorial Optimization: Algorithms and Complexity, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1982.

[20] R. E. Tarjan, Data Structures and Network Algorithms, Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1983.

[21] W. J. Cook, W. H. Cunningham, W. R. Pulleyblank, A. Schrijver, Combinatorial Optimization, John Wiley & Sons, Inc., New York, NY, USA, 1998.

[22] B. Korte, J. Vygen, Combinatorial Optimization: Theory and Algorithms, 4th Edition, Springer Publishing Company, Incorporated, 2007.

[23] M. A. Hameed, A. Jabbar, E. K. Çetinkaya, J. P. Sterbenz, Deriving Network Topologies from Real World Constraints, in: Proceedings of IEEE GLOBECOM Workshop on Complex and Communication Networks (CCNet), Miami, FL, 2010, pp. 400–404.

[24] J. P. Sterbenz, E. K. Çetinkaya, M. A. Hameed, A. Jabbar, Q. Shi, J. P. Rohrer, Evaluation of Network Resilience, Survivability, and Disruption Tolerance: Analysis, Topology Generation, Simulation, and Experimentation (invited paper), Telecommunication Systems 52 (2) (2013) 705–736.

[25] E. K. Çetinkaya, M. J. F. Alenazi, Y. Cheng, A. M. Peck, J. P. G. Sterbenz, On the Fitness of Geographic Graph Generators for Modelling Physical Level Topologies, in: Proceedings of the 5th IEEE/IFIP International Workshop on Reliable Networks Design and Modeling (RNDM), Almaty, 2013, pp. 38–45.

[26] E. K. Çetinkaya, M. J. F. Alenazi, A. M. Peck, J. P. Rohrer, J. P. G. Sterbenz, Multilevel Resilience Analysis of Transportation and Communication Networks, Springer Telecommunication Systems Journal(accepted in July 2013).

[27] L. d. F. Costa, F. A. Rodrigues, G. Travieso, P. R. Villas Boas, Characterization of complex networks: A survey of measurements, Advances in Physics 56 (1) (2007) 167–242.

[28] I. M. Soi, K. K. Aggarwal, Reliability indices for topological design of computer communication networks, IEEE Transactions on Reliability R-30 (5) (1981) 438–443.

[29] J. P. Rohrer, A. Jabbar, J. P. Sterbenz, Path Diversification for Future Internet End-to-End Resilience and Survivability, Springer Telecommunication Systems.

[30] E. K. Çetinkaya, M. J. F. Alenazi, J. P. Rohrer, J. P. G. Sterbenz, Topology Connectivity Analysis of Internet Infrastructure Using Graph Spectra, in: Proceedings of the 4th IEEE/IFIP International Workshop on Reliable Networks Design and Modeling (RNDM), St. Petersburg, 2012, pp. 752–758.

[31] AT&T [online].

[32] Level 3 network map [online].

[33] Sprint [online].

[34] Internet2 [online].

[35] G. Clapp, R. A. Skoog, A. C. Von Lehmen, B. Wilson, Management of Switched Systems at 100 Tbps: the DARPA CORONET Program, in: International Conference on Photonics in Switching (PS), Pisa, 2009, pp. 1–4.

[36] The Next Generation Core Optical Networks (CORONET) [online].

[37] KMI Corporation, North American Fiberoptic Long-haul Routes Planned and in Place (1999).

[38] J. P. Rohrer, A. Jabbar, J. P. G. Sterbenz, Path diversification: A multipath resilience mechanism, in: Proceedings of the IEEE 7th International Workshop on the Design of Reliable Communication Networks (DRCN), Washington, DC, 2009, pp. 343–351.

[39] L. C. Freeman, A Set of Measures of Centrality Based on Betweenness, Sociometry 40 (1) (1977) 35–41.

[40] P. Holme, B. J. Kim, C. N. Yoon, S. K. Han, Attack vulnerability of complex networks, Phys. Rev. E 65 (2002) 056109.

[41] M. J. F. Alenazi, E. K. Çetinkaya, J. P. G. Sterbenz, Network Design and Optimisation Based on Cost and Algebraic Connectivity, in: Proceedings of the 5th IEEE/IFIP International Workshop on Reliable Networks Design and Modeling (RNDM), Almaty, 2013, pp. 193–200.

7. Author Biographies

Mohammed J.F. Alenazi: is a Ph.D. candidate in the department of Electrical Engineering and Computer Science at The University of Kansas. He received his B.S. and M.S. degrees in Computer Engineering from the University of Kansas in 2010 and 2012 respectively. He is a graduate research assistant in the ResiliNets research group at the KU Information & Telecommunication Technology Center (ITTC). His research interests are in resilient networks, particularly network topology optimisation using graph metrics. He is a member of the IEEE, ACM, and Upsilon Pi Epsilon (UPE).

Egemen K. Çetinkaya: is Assistant Professor of Electrical & Computer Engineering at Missouri University of Science and Technology (formerly known as University of Missouri–Rolla). He received the B.S. degree in Electronics Engineering from Uludağ University (Bursa, Turkey) in 1999, the M.S. degree in Electrical Engineering from University of Missouri–Rolla in 2001, and Ph.D. degree in Electrical Engineering from the University of Kansas in 2013. He held various positions at Sprint as a support, system, and design engineer from 2001 until 2008. His research interests are in resilient networks. He is a member of the IEEE Communications Society, ACM SIGCOMM, and Sigma Xi.

James P.G. Sterbenz: is Professor of Electrical Engineering & Computer Science and on staff at the Information & Telecommunication Technology Center at The University of Kansas, and is a Visiting Professor of Computing in InfoLab 21 at Lancaster University in the UK. He received a doctorate in computer science from Washington University in St. Louis in 1991, with undergraduate degrees in electrical engineering, computer science, and economics. He is director of the ResiliNets research group at KU, PI for the NSF-funded FIND Postmodern Internet Architecture project, PI for the NSF Multilayer Network Resilience Analysis and Experimentation on GENI project, lead PI for the GpENI (Great Plains Environment for Network Innovation) international GENI and FIRE testbed, co-I in the EU-funded FIRE ResumeNet project, and PI for the US DoD-funded highly-mobile airborne networking project. He has previously held senior staff and research management positions at BBN Technologies, GTE Laboratories, and IBM Research, where he has lead DARPA- and internally-funded research in mobile, wireless, active, and high-speed networks. He has been program chair for IEEE GI, GBN, and HotI; IFIP IWSOS, PfHSN, and IWAN; and is on the editorial board of IEEE Network. He has been active in Science and Engineering Fair organisation and judging in Massachusetts and Kansas for middle and high-school students. He is principal author of the book High-Speed Networking: A Systematic Approach to High-Bandwidth Low-Latency Communication. He is a member of the IEEE, ACM, IET/IEE, and IEICE. His research interests include resilient, survivable, and disruption tolerant networking, future Internet architectures, active and programmable networks, and high-speed networking and systems.