

An Efficient Spectral Bound for Link Vulnerability Assessment in Large-scale Networks

Thang N. Dinh, Yilin Shen, and My T. Thai

Dept. of Comp. & Info. Sci. & Eng.

University of Florida

Gainesville, FL, US, 32611

{tdinh, yshen, mythai}@cise.ufl.edu

Abstract—Simultaneous attacks can cause devastating damage, breaking down communication networks into small fragments. To mitigate the risk and develop proactive responses, it is essential to assess the robustness of network in the worst-case scenarios. In this paper, we propose a spectral lower-bound on the number of removed links to incur a certain level of disruption in terms of *pairwise connectivity*. Our lower-bound explores the latent structural information in the *network Laplacian spectrum*, the set of eigenvalues of the Laplacian matrix, to provide guarantees on the robustness of the network against intentional attacks. Such guarantees often cannot be found in heuristic methods for identifying critical infrastructures. For the first time, the attack-resistant proofs of large scale communication networks against link attacks are presented.

Index Terms—Lower-bound method; vulnerability assessment; pairwise connectivity; spectral algorithm;

I. INTRODUCTION

Connectivity plays a vital role in network performance and is fundamental to vulnerability assessment. The number of connected node pairs in the network, (a.k.a pairwise connectivity), lends itself as an effective measure to account for the effect of the attacks [1], [2], [3], [4], [5], [6], [7].

Vulnerability assessment has been recently formulated as an connectivity optimization problem called *β -edge disruptor*, which finds a *minimum cost links* whose removal causes a *significant level (β) of network pairwise degradation* [6]. The *β -edge disruptor* reflects the common sense that when breaking the network by removing links, the more links required to be removed, the less vulnerable the network is. The *β -edge disruptor* approach enables the exploration of different network disruption levels which can be used to gain the deeper insight into network structure and robustness in various operating environments.

Unfortunately, the *β -edge disruptor* problem is NP-hard [6] i.e. there is no efficient algorithm to solve the problem, unless $P=NP$. A pseudo-approximation algorithm and mathematical approaches for the *β -edge disruptor* problems are introduced in [6] and [5], respectively. Although those methods can provide performance guarantees, they are only applicable for small and medium networks of few thousand nodes. For larger networks, we have to rely on heuristics which can have arbitrary bad worst-case performance. Hence, there is a lack of methods to provide robustness proofs against intentional attacks for large networks.

In this paper, we analyze the network spectrum, the eigenvalues of the Laplacian matrix, to give a *lower-bound* for the minimum size of a *β -edge disruptor*, thus, give a certificate on the robustness of the network. Our spectral bound is formulated as an optimization problem of the Laplacian eigenvalues, which are known to contain rich information about the topological structure [8].

Since exact measurement for the *β -edge disruptor* is not available in general, our lower-bound can be coupled with upper bound methods¹ to narrow down the range for actual vulnerability/robustness of the network. We emphasize that while upper bounds for *β -edge disruptor* (or any other minimization problem) can be designed easily, techniques for deriving lower-bound is much scattered in literature. Our contributions are summarized as follows.

- We introduce a new spectral lower-bound for the *β -edge disruptor* problem in form of an eigenvalue optimization problems. At the same time, we enrich the literature on lower-bound techniques.
- We present two efficient methods to compute the proposed lower-bound: 1) the Lagrange multiplier method and 2) the dynamic programming algorithm. Moreover, the Lagrange multiplier method can derive the lower-bound with only a small number of smallest eigenvalues. This is important for large networks where computing the whole network spectrum is both time and memory consuming.
- We perform experiments on different network types and real large-scale networks to demonstrate the quality of the proposed lower-bound and quantify the robustness of the studied networks against intentional attacks.

Related work. Many existing works on network vulnerability assessment mainly focus on the local centrality measurements to differentiate between critical links and nodes and the others, see [9], [6]. Other global graph measures have also been proposed to assess network vulnerability. These measures are mainly functions of graph properties, such as the diameter, global clustering coefficient, etc. [3], [10]. Matisziw and Murray [9] first proposed the pairwise connectivity as an effective measurement and use mathematical programming to solve for exact solutions. Arulselman et al. later define

¹Each heuristic to find *β -edge disruptor* is an upper bound for the problem

the Critical Node/Edge problem problems, which the main objective is to identify top k nodes/links whose removal minimize the pairwise connectivity in the residual network, and provide NP-completeness proofs and integer programming formulations. However, the run-time for exact solutions scale exponentially with the network size.

We first proposed the assessing vulnerability methods in form of optimization problems β -edge/vertex disruptor in [6]. The paper presents NP-hardness of β -edge/vertex disruptor problems, an $O(\log^{1.5} n)$ pseudo-approximation algorithm for β -edge disruptor, and an $O(\log n \log \log n)$ pseudo-approximation algorithm for β -vertex disruptor. The proposed pseudo-approximation algorithms are of theoretical interests as they provide performance guarantees.

Bissias et al. [1], [2] study the problem of bounding the damage under link attacks. However, the provided methods either require solving costly semidefinite programming problem [1] or involving weak bounds due to the presence of partitions with negative sizes [2].

Organization. We briefly present terminologies and problem definitions in Section II. In Section III, we introduce the spectral lower-bound for the the β -disruptor problem together with two methods to compute the lower-bound. Experimental results on different network models and real network instances are obtained in Section IV. Finally, we conclude the paper in Section V.

II. MODEL AND DEFINITIONS

We abstract our general network model as a graph $G = (V, E)$, where $V = \{v_1, v_2, \dots, v_n\}$ refers to a set of nodes and E refers to a set of links. Each edge $(v_i, v_j) \in E$ has a removal cost $c_{ij} \geq 0$ (and $c_{ij} = 0$ if $(v_i, v_j) \notin E$). For convenience, we also denote the number of nodes and links by n and m , respectively.

Since the main purpose of network lies in connecting all the interacting elements in the network, we study on the *overall pairwise connectivity*, which is defined as the number of connected vertex pairs in G . If G is an undirected graph, a vertex pair $(u, v) \in V \times V$ is connected iff there exists a path between u and v . We denote the pairwise connectivity of a graph G by $\mathcal{P}(G)$. Apparently, the pairwise connectivity is maximized at $\binom{n}{2}$ when G is a (strongly) connected graph.

The β -edge disruptor is defined in [6] as follows.

β -edge disruptor. Given $0 \leq \beta \leq 1$, a subset $E_\beta \subset E$ in $G = (V, E)$ is a β -edge disruptor if the overall pairwise connectivity of $G[E \setminus E_\beta]$, obtained by removing E_β from G , is no more than $\beta \binom{n}{2}$. In the β -edge disruptor problem, we aim to find a minimum cost β -edge disruptor.

Laplacian Matrix and Its Eigenvalues. Let $\mathbf{A} = \{c_{ij}\}$ be the weighted adjacency matrix and \mathbf{D} be the degree matrix, defined as the diagonal matrix with the weighted degrees d_1, d_2, \dots, d_n on the diagonal, where $d_i = \sum_j c_{ij}$. The unnormalized graph Laplacian matrix [11] is defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{A}$$

The matrix \mathbf{L} is symmetric and positive semi-definite, since

for every vector $x \in \mathbb{R}^n$ we can verify that

$$x^T \mathbf{L} x = \frac{1}{2} \sum_{i,j=1}^n c_{ij} (x_i - x_j)^2 \geq 0. \quad (1)$$

A direct consequence is that \mathbf{L} has n non-negative, real-valued eigenvalues $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$. In addition, the smallest eigenvalue of λ_1 is zero and the corresponding eigenvector is the constant one vector $\mathbf{1}$ [11].

The second smallest eigenvector λ_2 is known as the algebraic connectivity of the graph and can be used to describe many properties of graphs [11]. For example, the graph G is connected if and only if $\lambda_2 > 0$. For β -edge disruptor problem, the following lower-bound can be derived from λ_2 .

Lemma 1: For any connected graph G , we have

$$\text{OPT}_\beta \geq \frac{1-\beta}{2} \lambda_2 (n-1) \quad (2)$$

where OPT_β denotes the minimum size of a β -edge disruptor.

However, the bound provided in Eq. 2 is rather loose, as the value of λ_2 is often very close to zero (for example when bridges, edges whose deletion increases the number of connected components, are presented in the networks.) This motivates us to study higher eigenvalues beyond λ_2 to design stronger bound for the β -edge disruptor problem.

III. SPECTRAL LOWER-BOUND FOR LINK ASSESSMENT

In this section, we derive a lower-bound on size of β -edge disruptor using higher eigenvalues of the Laplacian matrix \mathbf{L} . We first formulate the lower-bound as an eigenvalue optimization problem. Then two methods with different trade-off between time and quality are introduced to compute the lower-bound.

Let E_β^* be an optimal β -edge disruptor and $s_1^* \geq s_2^* \geq \dots \geq s_n^*$ be the sizes of the connected components after removing E_β^* from the network. Then we can relate OPT_β to the size of the components via the following lemma.

Lemma 2: [12] Let a k -partition of a graph be a division of the vertices into k disjoint subsets containing $s_1 \geq s_2 \geq \dots \geq s_k$ vertices. Let E_{cut} be the set of edges whose two vertices belong to different subsets. Let $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_k$, be the k smallest eigenvalues of the Laplacian matrix plus any diagonal matrix \mathbf{U} such that the sum of all the elements of \mathbf{U} is zero. Then

$$|E_{\text{cut}}| \geq \frac{1}{2} \sum_{i=1}^k s_i \lambda_i.$$

Thus, we have $\text{OPT}_\beta = |E_\beta^*| \geq \frac{1}{2} \sum_{i=1}^n s_i^* \lambda_i$. Here we allow imaginary subsets of size zero and assume w.l.o.g. that $k = n$. Note that s_1^*, \dots, s_n^* are not known without finding E_β^* . Thus, we consider all possible values of $\{s_1, \dots, s_n\}$ which infer network partitions of pairwise connectivity at most $\beta \binom{n}{2}$, and get the minimum of the sum $\frac{1}{2} \sum_{i=1}^n s_i \lambda_i$ as a lower-bound on OPT_β .

Formally, our spectral lower-bound on OPT_β is given by solving the following quadratic programming (QP) optimization problem.

$$\text{minimize } \frac{1}{2} \sum_{i=1}^n s_i \lambda_i \quad (3)$$

$$\text{subject to } \sum_{i=1}^n s_i = n \quad (4)$$

$$\sum_{i=1}^n \binom{s_i}{2} \leq \beta \binom{n}{2} \quad (5)$$

$$s_i \in \{0, 1, \dots, n\} \quad (6)$$

Theorem 1: Let Q_β be the optimal objective of the QP problem (8-11) and OPT_β be the minimum β -edge disruptor of graph $G = (V, E)$. Then, $Q_\beta \leq \text{OPT}_\beta$ for $\beta \in [0, 1]$. Moreover, the equality holds when $\beta = 0$ or $\beta = 1$.

Proof: As discussed in the previous paragraph, the sizes of connected components after removing optimal β -edge disruptor satisfy all constraints (4-6). Hence, $Q_\beta \leq \text{OPT}_\beta$ for all $\beta \in [0, 1]$. We continue with the tightness of the bound at extreme cases when $\beta = 0$ and $\beta = 1$.

Case $\beta = 0$: all subsets are of size one. Hence, $Q_1 = \frac{1}{2} \sum_{i=1}^n \lambda_i = \frac{1}{2} \text{Trace}(\mathbf{L}) = \frac{1}{2} (2|E|) = |E|$. The only way to cut all pairs in the network is to cut all edges. In other words, $Q_0 = \text{OPT}_0 = |E|$.

Case $\beta = 1$: in order to achieve the maximum connectivity $\binom{n}{2}$, there must be a single partition in the network and the optimal disruptor cutting no edges. That is $s_1 = n$ and $s_i = 0 \forall i > 1$. Since $\lambda_1 = 0$, it follows that $Q_1 = 0 = \text{OPT}_1$. ■

Since s_i are integral values, we propose a dynamic programming algorithm to compute the spectral bound in next subsection.

A. Dynamic Programming Method

We first describe the optimal solution structure for the optimization problem in (8-11).

Lemma 3: There exists an optimal solution \mathbf{s}^* of QP(3-6) such that $s_1^* \geq s_2^* \geq \dots \geq s_n^*$.

Proof: Let $\mathbf{s}^* = \{s_1^*, s_2^*, \dots, s_n^*\}$ be an optimal solution of QP(8-11). Denote $\text{inv}(\mathbf{s}^*)$ the number of inversions of \mathbf{s}^* i.e. such pairs of indices (i, j) that $i < j$ such that $s_i^* > s_j^*$. If $\text{inv}(\mathbf{s}^*) = 0$, then $s_1^* \geq s_2^* \geq \dots \geq s_n^*$, otherwise there exists a pair $i < j$ and $s_i^* > s_j^*$. Construct \mathbf{s}' by swapping s_i^* and s_j^* inside \mathbf{s}^* . Then, \mathbf{s}' is a feasible solution of QP(8-11) and the objective increases an amount $s_i^* \lambda_j + s_j^* \lambda_i - (s_i^* \lambda_i + s_j^* \lambda_j) = (s_i^* - s_j^*)(\lambda_j - \lambda_i) \geq 0$. Thus, we obtain a new optimal solution with less the number of inversions. Repeat the process at most $\binom{n}{2}$, that is the maximum number of inversions in \mathbf{s}^* , we finally obtain an optimal solution with no inversions. That optimal solution shall satisfy the lemma's condition. ■

For $k \leq l \leq n$ and $p \leq \beta \binom{n}{2}$, define $\mathcal{L}_k(l, p)$ to be the minimum spectral bound obtained by first k subsets that the total sizes is l and the total pairwise connectivity is at most p . That is

$$\mathcal{L}_k(l, p) = \min_{\mathbf{s}^{(k)} \in \mathbb{N}^k} \left\{ \mathbf{s}^{(k)T} \boldsymbol{\lambda}^{(k)} : \|\mathbf{s}^{(k)}\|_1 = l, \sum_{i=1}^k \binom{s_i}{2} \leq p \right\},$$

Algorithm 1 $\text{ILB}(G, \beta)$

- 1: Compute $\lambda_1, \dots, \lambda_n$
 - 2: $\mathcal{L}_k(l, p) = \begin{cases} +\infty, & \text{if } p < p_{\min}(l, k) \\ \lambda_1 l = 0, & \text{if } p \geq p_{\max}(l, k) \end{cases}$
 - 3: **for** $k = 1$ **to** n
 - 4: **for** $l = 1$ **to** n
 - 5: **for** $p = p_{\min}(l, k)$ **to** $\min\{\beta \binom{n}{2}, p_{\max}(l, k)\}$
 - 6: $\mathcal{L}_k(l, p) = \min \left\{ \begin{array}{l} \mathcal{L}_{k-1}(l, p), \\ \mathcal{L}_k(l - k, p - l + k) + \sum_{i=1}^k \lambda_i \end{array} \right\}$
 - 7: **if** $\mathcal{L}_{k-1}(n, \beta \binom{n}{2}) = \mathcal{L}_k(n, \beta \binom{n}{2})$
 - 8: **return** $\lceil \mathcal{L}_k(n, \beta \binom{n}{2}) \rceil$
 - 9: **return** $\lceil \mathcal{L}_n(n, \beta \binom{n}{2}) \rceil$
-

Then the optimal objective value QP(3-6) shall be given by $Q_\beta = \mathcal{L}_n(n, \beta \binom{n}{2})$.

By Lemma 3, we pay attention only to partitions satisfying $s_1 \geq s_2 \geq \dots \geq s_n$. We now derive the recursive formula for $\mathcal{L}_p(l, k)$ based on the sub-optimal structure of the QP problem. Consider two possible cases of s_k

- $s_k = 0$: There are at most $k - 1$ partitions whose sizes sum up to l . Hence, for this case $\mathcal{L}(l, k) = \mathcal{L}_{k-1}(l, p)$.
- $s_k > 0$: Since $s_1 \geq s_2 \geq \dots \geq s_k > 0$. Let $\tilde{s}_i = s_i - 1 \geq 0$, the vector $\tilde{\mathbf{s}} = \{\tilde{s}_1, \tilde{s}_2, \dots, \tilde{s}_k\}$ satisfies simultaneously the following

$$\begin{aligned} \sum_{i=1}^k \lambda_i \tilde{s}_i &= \sum_{i=1}^k \lambda_i s_i - \sum_{i=1}^k \lambda_i \\ \sum_{i=1}^k \tilde{s}_i &= \sum_{i=1}^k s_i - k = l - k \\ \sum_{i=1}^k \binom{\tilde{s}_i}{2} &= \sum_{i=1}^k \left[\binom{s_i}{2} - s_i + 1 \right] \\ &= \sum_{i=1}^k \binom{s_i}{2} - l + k \leq p - l + k \end{aligned}$$

Therefore, in this case $\mathcal{L}_k(l, p) = \mathcal{L}_k(l - k, p - l + k) + \sum_{i=1}^k \lambda_i$

In summary, we have

$$\mathcal{L}_k(l, p) = \min \left\{ \begin{array}{l} \mathcal{L}_{k-1}(l, p), \\ \mathcal{L}_k(l - k, p - l + k) + \sum_{i=1}^k \lambda_i \end{array} \right\}$$

We compute value of $\mathcal{L}_p(l, k)$ in increasing order of p and l but in decreasing order of k . The base cases for $\mathcal{L}_p(l, k)$ are as follow.

$$\mathcal{L}_k(l, p) = \begin{cases} +\infty, & \text{if } p < p_{\min}(l, k) \\ \lambda_1 l = 0, & \text{if } p \geq p_{\max}(l, k) \end{cases} \quad (7)$$

where $p_{\min}(l, k) = \binom{l/k}{2} (l \bmod k) + \binom{l/k+1}{2} (k - l \bmod k)$ and $p_{\max}(l, k) = \binom{l}{2}$ that are the minimum and maximum pairwise connectivity of a graph with l vertices and k connected components, respectively.

Theorem 2: Optimal solutions of QP(3-6) can be found in $O(n^4)$ time and $O(n^3)$ space.

Thus, the spectral bound can be computed in polynomial time. However, the high time complexity of the dynamic programming algorithm prevents the method from being applied to large networks. Moreover, the dynamic programming algorithm requires computing the whole set of eigenvalues of the networks, which is both time and memory consuming. We continue with an approximation of the spectral bound that achieves (almost) the same lower-bound quality in significantly less time.

B. Lagrange Multipliers Method

We relax the integral conditions on s_i to obtain the following relaxation of the QP, rewritten in vector notation.

$$\text{minimize } \frac{1}{2} \mathbf{s}^T \boldsymbol{\lambda} \quad (8)$$

$$\text{subject to } \|\mathbf{s}\|_1 - n = 0, \quad (9)$$

$$\|\mathbf{s}\|_2^2 - \Delta_\beta \leq 0, \quad (10)$$

$$\mathbf{s} \geq 0, \quad (11)$$

where $\Delta_\beta = \beta n(n-1) + n$ and $\|\cdot\|$ denotes the Euclidean norm.

The Lagrange multiplier is then

$$\mathcal{L}(\mathbf{s}, \chi, \psi, \boldsymbol{\omega}) = \frac{1}{2} \mathbf{s}^T \boldsymbol{\lambda} + \chi(\|\mathbf{s}\|_1 - n) + \psi(\|\mathbf{s}\|_2^2 - \Delta_\beta) - \boldsymbol{\omega}^T \mathbf{s}$$

where $\boldsymbol{\omega} = (\omega_1, \dots, \omega_n) \geq 0$ is a positive multiplier vector.

Notice that the problem is a convex optimization problem with differentiable objective and constraint functions and it satisfies the Slater's condition with $\mathbf{s} = (1, 1, \dots, 1)^T$ [13]. Hence, the following Karush–Kuhn–Tucker (KKT) conditions provide the necessary and sufficient conditions for optimality

$$\begin{aligned} \nabla_{\mathbf{s}} \mathcal{L} &= \frac{1}{2} \boldsymbol{\lambda} + \chi + 2\psi \mathbf{s} - \boldsymbol{\omega} &= 0 \\ \nabla_{\chi} \mathcal{L} &= \|\mathbf{s}\|_1 - n &= 0 \\ \nabla_{\psi} \mathcal{L} &= \|\mathbf{s}\|_2^2 - \Delta_\beta &= 0 \\ \boldsymbol{\omega}^T \mathbf{s} & &= 0 \\ \mathbf{s}, \psi, \boldsymbol{\omega} & &\geq 0 \end{aligned}$$

Algorithm 2 LMB(G, β)

- 1: $t = \lceil 2/\beta \rceil$, $\Delta_\beta \leftarrow \lfloor \beta n(n-1) + n \rfloor$
 - 2: Compute $\lambda_1, \dots, \lambda_t$
 - 3: **for** $k = 1$ **to** n
 - 4: **if** $k > t$ **then**
 - 5: $t = \min\{2t, n\}$
 - 6: Compute $\lambda_1, \dots, \lambda_t$
 - 7: Compute ψ as in Eq. 19.
 - 8: Compute $\mathcal{D}_\beta^{(k)}$, and $\mathcal{C}_\beta^{(k)}$ as in Eqs. 20, and 21
 - 9: **if** ($\psi \geq 0$ and $\mathcal{C}_\beta^{(k)} \geq 0$) or ($k = n$) **then**
 - 10: return $\lceil \mathcal{D}_\beta^{(k)} \rceil$
 - 11: **end for**
-

Let $k = \max\{i \mid s_i > 0\}$. By Lemma 3 and the complementary slackness $\boldsymbol{\omega}^T \mathbf{s} = 0$, we have $s_i > 0$ for $i \leq k$, thus, $s_i = 0 \forall i > k$ and $\omega_j = 0 \forall j \leq k$.

Denote $\mathbf{s}^{(k)} = \{s_1, s_2, \dots, s_k\}$ and $\boldsymbol{\lambda}^{(k)} = \{\lambda_1, \lambda_2, \dots, \lambda_k\}$, the KKT condition can be simplified to

$$\nabla_{\mathbf{s}^{(k)}} \mathcal{L} = \frac{1}{2} \boldsymbol{\lambda}^{(k)} + \chi + 2\psi \mathbf{s}^{(k)} = 0, \quad i \leq k \quad (12)$$

$$\nabla_{s_i} \mathcal{L} = \frac{1}{2} \lambda_i + \chi - \omega_i = 0, \quad i > k \quad (13)$$

$$\nabla_{\chi} \mathcal{L} = \|\mathbf{s}^{(k)}\|_1 - n = 0, \quad (14)$$

$$\nabla_{\psi} \mathcal{L} = \|\mathbf{s}^{(k)}\|_2^2 - \Delta_\beta = 0, \quad (15)$$

$$\mathbf{s}^{(k)} > 0, \psi > 0, \boldsymbol{\omega}^{(k)} = 0 \quad (16)$$

For each value of k , we can solve for values of s_i and check if all $s_i \geq 0$. The other unknowns can be found as follows. First, substitute the constraint (14) into the sum of the constraints (12) to obtain χ in terms of ψ .

$$\chi = -2\frac{n}{k}\psi - \frac{\|\boldsymbol{\lambda}^{(k)}\|_1}{2k} \quad (17)$$

Therefore, we can derive $\mathbf{s}^{(k)}$ from (12) as

$$\mathbf{s}^{(k)} = \frac{n}{k} + \left(\frac{\|\boldsymbol{\lambda}^{(k)}\|_1}{4k} - \frac{\boldsymbol{\lambda}^{(k)}}{4} \right) \frac{1}{\psi} \quad (18)$$

Substituting the above equation into the condition (15) and solving for ψ , we have

$$\begin{aligned} \|\mathbf{s}^{(k)}\|_2^2 - \Delta_\beta &= 0 \\ \Leftrightarrow \left(\frac{\|\boldsymbol{\lambda}^{(k)}\|_2^2}{16} - \frac{\|\boldsymbol{\lambda}^{(k)}\|_1^2}{16k} \right) \frac{1}{\psi^2} &= \Delta_\beta - \frac{n^2}{k} \\ \Leftrightarrow \psi &= \frac{1}{4} \left(\frac{\|\boldsymbol{\lambda}^{(k)}\|_2^2 - \|\boldsymbol{\lambda}^{(k)}\|_1^2/k}{\Delta_\beta - \frac{n^2}{k}} \right)^{1/2} \end{aligned} \quad (19)$$

The objective is then

$$\begin{aligned} \mathcal{D}_\beta^{(k)} &= \frac{1}{2} \mathbf{s}^{(k)T} \boldsymbol{\lambda}^{(k)} = n \frac{\|\boldsymbol{\lambda}^{(k)}\|_1}{2k} + \left(\frac{\|\boldsymbol{\lambda}^{(k)}\|_1^2}{4k} - \frac{\|\boldsymbol{\lambda}^{(k)}\|_2^2}{4} \right) \frac{1}{2\psi} \\ &= n \frac{\|\boldsymbol{\lambda}^{(k)}\|_1}{2k} - \frac{1}{2} \left(\|\boldsymbol{\lambda}^{(k)}\|_2^2 - \frac{\|\boldsymbol{\lambda}^{(k)}\|_1^2}{k} \right)^{1/2} \left(\Delta_\beta - \frac{n^2}{k} \right)^{1/2} \end{aligned} \quad (20)$$

Since $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$, Eq. 18 implies that $s_1^{(k)} \geq s_2^{(k)} \geq \dots \geq s_k^{(k)}$. Hence, in order to satisfy $\mathbf{s}^{(k)} > 0$, it is sufficient that

$$\mathcal{C}_\beta^{(k)} = s_k^{(k)} = \frac{n}{k} + \left(\frac{\|\boldsymbol{\lambda}^{(k)}\|_1}{4k} - \frac{\boldsymbol{\lambda}_k}{4} \right) \frac{1}{\psi} \geq 0. \quad (21)$$

Theorem 3: The size of a β -edge disruptor is lower-bounded by

$$\mathcal{D}_\beta = \min_{n \geq k \geq n^2/\Delta_\beta} \left\{ \mathcal{D}_\beta^{(k)} \mid \mathcal{C}_\beta^{(k)} > 0 \right\},$$

where $\mathcal{D}_\beta^{(k)}$ and $\mathcal{C}_\beta^{(k)}$ are given by Eqs. 20 and 21.

The steps to solve the relaxation of the QP is summarized in the Algorithm 2 (LMB Algorithm).

Time complexity. The LMB algorithm spends its major time on computing the eigenvalues. This can be done with Implicitly Restarted Lanczos Method which has worst-case

time complexity $O(mKh + nK^2h + K^3h)$ where K is the number of eigenvalues to be computed, and h is the number of iterations for the eigenvalue algorithm to converge [14]. Given the eigenvalues, the rest of LMB takes only $O(n)$ time in the worst-case.

The number of required eigenvalues K is small in our algorithm. At beginning, the algorithm computes $t = \lceil 2/\beta \rceil$ smallest eigenvalues and the number of computed eigenvalues is double each time if necessary. In our experiments, the number of needed eigenvalues is $2/\beta$ in most cases. For example, to bound the number of necessary links whose removal disrupts 90% pairwise connectivity we only need to compute about 20 smallest eigenvalues of the Laplacian matrix. We found the LMB algorithm to be scalable, taking *linear time* with respect to the number of nodes and edges.

C. Time and quality trade-off

On one hand, the ILB algorithm (Algorithm 1) provides a better bound than that of the LMB algorithm. The reason is that ILB solves for exact solutions of the QP while LMB only targets a relaxation of the QP. However, the difference between the output of two algorithms is negligible small and either zero or one ² in our experiments.

On the other hand, the LMB has much more practical time complexity. The ILB has high time complexity $O(n^4)$ and can only applied for network up to few thousand nodes. In contrast, LMB takes only linear time to compute its competitive bound. Overall for small and medium networks, one can apply ILB algorithm (or other mathematical approaches [5]) to compute the lower-bound, however, for large networks LMB remains the only choice.

IV. EXPERIMENTAL RESULTS

We compute our spectral lower-bound for both synthetic and real-world networks and compare the results with the optimal results whenever possible.

A. Synthetic Networks

We generate the synthetic networks following well-known complex network models. All networks have 100 nodes and around 300 edges. The details of those networks are as follows.

- **Erdos-Reyni**: A random graph of 100 vertices and 300 edges following the Erdos-Reyni model [15].
- **Barabasi-Albert**: A power-law model using preferential attachment mechanism [10].
- **Small world**: A random graph following Watts and Strogatz model [16]. The dimension of the lattice is set to be 3 and the rewiring probability is 0.3.

The optimal solutions are found with the integer programming using the sparse metric technique in [5]. The technique in [5] is also applied to compute the lower-bound given by solving the linear programming. The results produced by ILB and LMB algorithms are identical (after rounded up) and plotted under the same name “spectral bound”. All algorithms were run on a

PC with Intel Xeon 2.93 Ghz processor and 12 GB memory. The integer programming (IP) and the linear programming (LP) are solved with the mathematical optimization package GUROBI 4.5.

The minimum number of links whose removal causes certain level of disruption, are shown in Fig. 1. For all three different networks, solving LP gives good lower-bound on the minimum number of links to remove. The spectral bounds are much worse than the LP bounds in the random and small-world networks; however, the spectral bound closely approaches the LP bounds and the optimal solution when the network has the power-law topology of the Barabasi model.

As shown in Fig. 2, there is a big gap between the running time of the spectral bound and those of LP and IP. Note that all the spectral bound are computed at once, i.e., the provided running time is the total running time over all different values of β . Even though the running time of the spectral bound is still thousand of times faster than LP and IP.

Overall, while IP is best used for small networks, and LP can be used for medium networks of few thousand nodes, the only feasible method to compute the lower-bound in large networks is the spectral bound. One of the attractive aspect of the LMB spectral bound, described in the Alg. 2, is that the algorithm can be easily implemented in a distributed manner. The most time-consuming part of the algorithm is to compute the few smallest eigenvalues. This can be done distributedly with the existing mathematical software [17].

TABLE I: Sizes of the investigated networks and the corresponding running time to compute the lower-bound

	CAIDA AS	Oregon AS	P2P Gnutella
Vertices	8,020	11,174	22,663
Edges	36,406	23,410	109, 386
Time (s)	1530.1	321.0	207.9

B. Real-world Datasets

We compute the spectral lower-bounds for real networks are shown in Fig. 3. Neither LP nor IP can run on these networks due to both time and memory limits. The studied networks are

- **Gnutella P2P**: Gnutella peer-to-peer network from from Aug. 25, 2002 [18]. Nodes represents hosts in the network and edges are the connections between the Gnutella hosts.
- **Oregon AS**: AS peering information inferred from Oregon route-views between Mar. 31 and May 26, 2001 [18].
- **CAIDA AS**: The CAIDA AS Relationships Datasets, from September 17, 2007 [18].

The lower-bounds in Fig. 3 indicates that it is difficult to destroy major connectivity in communication networks. For examples, even after removing 369 links at least 50% node pairs in the CAIDA AS network stay connected; and to bring down the connectivity level in the Gnutella P2P network to 15% one has to destroy at least 960 links. Due to low edge density, the Oregon AS network tends to be more vulnerable than the other two networks. Nevertheless, utterly disrupting the connectivity in the network to 5% level would require removing more than 763 links.

²Both algorithms round up their results to the nearest integers.

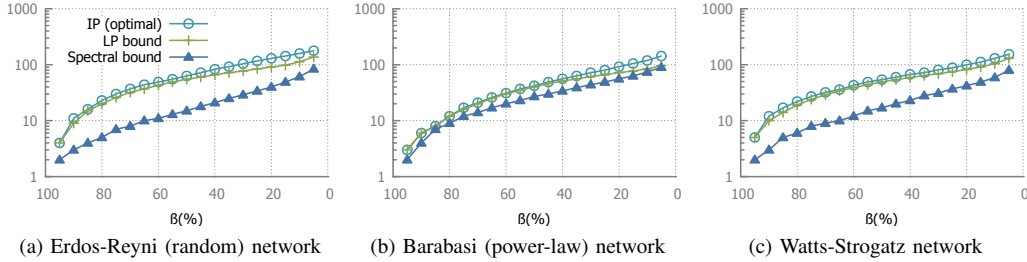


Fig. 1: Minimum cost and lower-bounds for β -disruptor on the synthesis networks

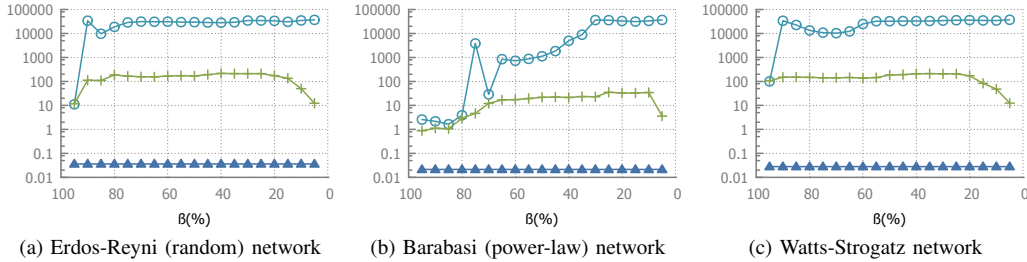


Fig. 2: Running time on the synthesis networks

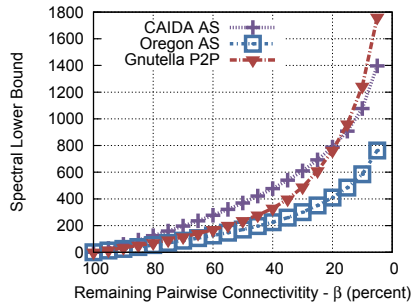


Fig. 3: Lower bounds on the number of link-attack for real networks found with the LMB algorithm.

V. CONCLUDING REMARKS

Assessing network for topological vulnerabilities is an important and challenging problem. We present in this paper a spectral lower-bound method for the link vulnerability assessment problem, β -edge disruptor. The new lower-bound method is useful in both comparing the vulnerability of different networks and providing guarantees for other heuristics assessment methods. In addition, the Lagrange multiplier method to compute the lower-bound requires only a portion of the eigenvalues and is applicable for large-scale networks.

VI. ACKNOWLEDGEMENT

This work is partially supported by NSF CAREER 0953284, and DTRA, Young Investigator Award, HDTRA1-09-1-0061.

REFERENCES

[1] G. D. Bissias, B. N. Levine, and A. Rosenberg, “Bounding damage from link destruction, with application to the internet,” *SIGMETRICS Perform. Eval. Rev.*, 2007.

[2] G. D. Bissias, “Bounds on service quality for networks subject to augmentation and attack,” Ph.D. dissertation, University of Massachusetts Amherst, 2010.

[3] A. Murray, T. Matisziw, and T. Grubestic, “Multimethodological approaches to network vulnerability analysis,” *Growth Change*, 2008.

[4] A. Arulselman, C. W. Commander, L. Elefteriadou, and P. M. Pardalos, “Detecting critical nodes in sparse graphs,” *Computers and Operations Research*, vol. 36, no. 7, 2009.

[5] T. N. Dinh and M. T. Thai, “Precise structural vulnerability assessment via mathematical programming,” in *Proc. of IEEE MILCOM*, 2011.

[6] T. N. Dinh, Y. Xuan, M. T. Thai, P. M. Pardalos, and T. Znati, “On new approaches of assessing network vulnerability: Hardness and approximation,” *IEEE/ACM Trans. Netw.*, vol. 20, no. 2, pp. 609–619, 2012.

[7] S. Neumayer, G. Zussman, R. Cohen, and E. Modiano, “Assessing the vulnerability of the fiber infrastructure to disasters,” *IEEE/ACM Trans. Netw.*, pp. 1610–1623, 2011.

[8] F. R. K. Chung, *Spectral Graph Theory (CBMS Regional Conference Series in Mathematics, No. 92)*. American Mathematical Society, Feb. 1997.

[9] T. C. Matisziw and A. T. Murray, “Modeling s-t path availability to support disaster vulnerability assessment of network infrastructure,” *Comput. Oper. Res.*, vol. 36, pp. 16–26, January 2009.

[10] A. Barabasi, R. Albert, and H. Jeong, “Scale-free characteristics of random networks: the topology of the world-wide web,” *Phys. A*, 2000.

[11] Mohar and Poljak, “Eigenvalue in combinatorial optimization,” *Combinatorial and Graph-Theoretical Problems in Linear Algebra*, 1992.

[12] W. E. Donath and A. J. Hoffman, “Lower bounds for the partitioning of graphs,” *IBM J. Res. Dev.*, vol. 17, 1973.

[13] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[14] S. White and P. Smyth, “A spectral clustering approach to finding communities in graph,” in *SDM*, 2005.

[15] P. Erdos and A. Renyi, “On the evolution of random graphs,” *Publ. Math. Inst. Hungar. Acad. Sci.*, vol. 5, pp. 17–61, 1960.

[16] D. J. Watts and S. H. Strogatz, “Collective dynamics of ‘small-world’ networks,” *Nature*, vol. 393, no. 6684, 1998.

[17] L. S. Blackford, J. Choi, A. Cleary, E. D’Azevedo, J. Demmel, I. Dhillon, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker, and R. C. Whaley, *ScaLAPACK user’s guide*. SIAM, 1997.

[18] J. Leskovec, J. Kleinberg, and C. Faloutsos, “Graphs over time: densification laws, shrinking diameters and possible explanations,” in *KDD*. ACM, 2005, pp. 177–187.