Asif, W., Lestas, M., Qureshi, H. K. & Rajarajan, M. (2015). Spectral Partitioning for Node Criticality. 2015 IEEE Symposium on Computers and Communication (ISCC), pp. 877-882. doi: 10.1109/ISCC.2015.7405624

CITY UNIVERSITY LONDON
EST 1894

City Research Online

Original citation: Asif, W., Lestas, M., Qureshi, H. K. & Rajarajan, M. (2015). Spectral Partitioning for Node Criticality. 2015 IEEE Symposium on Computers and Communication (ISCC), pp. 877-882. doi: 10.1109/ISCC.2015.7405624

Permanent City Research Online URL: http://openaccess.city.ac.uk/15499/

Spectral Partitioning for Node Criticality

Waqar Asif*‡, Marios Lestas §, Hassaan Khaliq Qureshi* and Muttukrishnan Rajarajan‡
‡School of Engineering and Mathematical Sciences, City University, London, UK.
Email: {waqar.asif.1, r.muttukrishnan}@city.ac.uk
* National University of Sciences & Technology (NUST), Islamabad, Pakistan.
Email: {09mscsewasif, hassaan.khaliq}@seecs.edu.pk
§Department of Electrical Engineering, Frederick University, Nicosia, Cyprus.
Email: eng.lm@frederick.ac.cy

Abstract—**Finding critical nodes in a network is a significant task, highly relevant to network vulnerability and security. We consider the node criticality problem as an algebraic connectivity minimization problem where the objective is to choose nodes which minimize the algebraic connectivity of the resulting network. Previous suboptimal solutions of the problem suffer from the computational complexity associated with the implementation of a maximization consensus algorithm. In this work, we use spectral partitioning concepts introduced by Fiedler, to propose a new suboptimal solution which significantly reduces the implementation complexity. Our approach, combined with recently proposed distributed Fiedler vector calculation algorithms enable each node to decide by itself whether it is a critical node. If a single node is required then the maximization algorithm is applied on a restricted set of nodes within the network. We derive a lower bound for the achievable algebraic connectivity when nodes are removed from the network and we show through simulations that our approach leads to algebraic connectivity values close to this lower bound. Similar behaviour is exhibited by other approaches at the expense, however, of a higher implementation complexity.**

I. INTRODUCTION

The occurrence of an unexpected event, such as a natural disaster or an attack by a malicious node on a few links or edges can have a great impact on the performance of the network. This effect can vary from having a small decrease in the QoS of a portion of the network upto the complete breakdown of the network [1]. In March 2011, a Georgian woman who was scavenging for copper to sell as scrap, cut off the web access to almost the whole of Armenia [2]. She damaged the main fiber optic that was connecting 90% of Armenia thus, depriving 3.2 million people from the access to the internet for five hours. In terms of graph theory, the woman cut off an edge between two critical nodes in the network thus disconnecting a major portion of the network.

Among various nodes that exist in the network, the nodes that have the highest influence on the performance of the network upon their removal are referred to as the Critical Nodes (CNs) of a network [2]. The identification of these CNs beforehand leads to an appreciation of the vulnerability of a network and in some cases aid in formulating a suitable solution which can help avoid, the degradation in performance or the network partitioning that will result from node failure. Numerous approaches have been proposed in literature to determine the critical nodes in a network. A few of the most commonly used metrics are: degree centrality [3] which accounts for the number of neighbours of a node to determine node centrality, betweenness centrality [4] and ego centrality [5], which determine the criticality of a node based on its contribution in forming the shortest path routes in a network with the former being a global metric whereas, the later using information gathered from the two hop neighbours of a node, the closeness centrality [3], which uses the distance of every node to every other node in the network to determine the CN set in the network and eigenvector centrality [6] which uses the largest eigenvector of the adjacency matrix of a network to determine node criticality. Another approach, which has been adopted in a number of recent studies, considers the algebraic connectivity as a metric for evaluating the node criticality.

The algebraic connectivity was introduced by Fiedler [7] and is defined as the second smallest eigenvalue of the Laplacian matrix of a network. It has been shown to be a good measure of the connectivity robustness of a network in the sense that the smallest its value is, the closer the network is in becoming disconnected. So, in many studies a node is considered as being critical if its removal leads to a high reduction in the algebraic connectivity of the network. Finding such critical nodes can be done in number of ways. A basic but tedious approach is to use an exhaustive search over n possible sub-graphs each of which is the resultant of the removal of one of the total n nodes in the network. This approach requires the information of the complete network and thus can be computationally expensive when dealing with large network structures. In addition, when multiple critical nodes need to be found the approach becomes computationally expensive with the number of subgraphs that need to considered increasing exponentially with the network size.

To avoid the use of this tedious and computationally expensive approach, researchers have proposed various sub-optimal solutions for determining the CNs of a network. These sub-optimal solutions were derived on the basis that, the algebraic connectivity of a network is the sum of the difference in the Fiedler vector values that correspond to the neighbouring nodes of a network. The term Fiedler vector refers to the eigenvector which is associated with the second smallest eigenvalue of the Laplacian of a network. Each entry of the eigenvector corresponds to a single node in the network. According to a popular suboptimal solution, a node is deemed as critical if it maximizes the summation of the differences with the

eigenvectors of its neighbouring nodes in a network as shown below [8][9]:

$$CN = \arg \max_{i \in V} \sum_{j \in N_i} (v_i - v_j)^2 \qquad (1)$$

Here, N_i is the set of neighbours of node i where $i, j \in V$ and v_i is the Fiedler vector value corresponding to node i. A slight variant of the suboptimal solution of Eq (1) has been proposed in literature, which uses a normalized version of the difference in the Fiedler vector values [10][11]:

$$CN = \arg \max_{i \in V} \frac{\sum_{j \in N_i} v_j(v_i - v_j)}{1 - v_i^2} \qquad (2)$$

Both these suboptimal solutions have been shown to perform well in specific scenarios but they suffer from the fact that a global maximisation consensus algorithm must be employed which can be slow and significantly increases the convergence time of the proposed algorithm. This problem is particularly vivid in distributed implementations of the proposed algorithm.

In this work we use spectral partitioning concepts that were introduced by Fiedler [7], to propose a new suboptimal solution which is shown to work effectively and is less computational expensive. A node compares its Fiedler value with the Fiedler values of its neighbours and classifies itself as being critical if it identifies a sign change with at least one of its neighbours. This approach is combined with a recently proposed distributed mechanism to calculate Fiedler vector values in a network [12], so that each node can evaluate its criticality role in a network. When, multiple critical nodes are sought then, no maximization consensus algorithm is required, whereas, when a single critical node must be detected in the entire network then, a maximization consensus algorithm must be employed only over a restricted node set which consists of all the nodes which report a sign change in Fiedler value with at least one of its neighbours. Despite the significant reduction in implementation complexity the approach is shown through simulations to perform at least equally well with other solutions which have been proposed in literature. In addition we derive, using mathematical analysis, a lower bound on the algebraic connectivity of the network when nodes are removed and we show that the proposed approach is able to choose critical nodes whose removal leads to algebraic connectivity values which are close to this bound. This demonstrates that the proposed distributive approach can correctly identify critical nodes in a network with reduced computational complexity.

The paper is organised as follows. In Section II, we introduce the relevant mathematical notation and review previous work in the area, in Section III, we present the proposed algorithm, in Section IV, we derive the lower bound on the algebraic connectivity upon node removal, in Section V, we evaluate the performance of the proposed approach using simulations and finally in Section VI we conclude our work.

II. PROBLEM FORMULATION

In this section, we introduce the relevant mathematical framework, we formulate the considered problem and present existing approaches found in literature.

We consider a Graph $G = (V, E)$ where $|V| = n$ and $|E| = m$ are the number of nodes and edges respectively. The incidence matrix A is the $n \times m$ matrix where the existence of an edge $l \in E$ between node i and j defines the lth column of the matrix such that $a_{l_i} = 1$ and $a_{l_j} = -1$. For such a graph the Laplacian matrix is defined as:

$$L = AA^T = \sum_{l=1}^{m} a_l a_l^T \qquad (3)$$

The diagonal entries of this Laplacian matrix $L_{i,i}$ denote the degree of node i and the non diagonal entries denote the existence of a link between two nodes. L is positive semi-definite and $L\mathbf{1} = 0$ where $\mathbf{1}$ is the vector of all ones.

The eigenvalues of the Laplacian matrix are arranged in ascending order such that $0 = \lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$. The multiplicity of zero in the eigenvalues represents the number of disconnected components of a network. The second smallest eigenvalue $\mu = \lambda_2$ represents the algebraic connectivity of the network with the corresponding normalized eigenvector being referred to as the Fiedler vector of the network [7]. The algebraic connectivity of a graph is well known to constitute a measure of the graph connectivity in the sense that the lower the value is, the closer the network is in becoming disconnected. Considering as critical nodes the ones which contribute the most to the network connectivity, one may define critical nodes as the ones which when removed minimize the algebraic connectivity of the network. This is referred to as optimization problem P which is shown formally below:

$$P: \quad CN = \arg \min_{\alpha \in V} \mu(G(V - \alpha)) \qquad (4)$$

One way of solving P when a single node is removed is through exhaustive search. However this approach is computationally expensive. In addition, when multiple nodes are removed, the complexity of the exhaustive solution increases combinatorially with increasing network size. So, people have sought suboptimal solutions which are simple to implement in a distributed manner. The most popular solutions are inspired from the following characterization of the algebraic connectivity [13]. The algebraic connectivity is the solution of the following optimization problem:

$$\mu(L(x)) = min\{\frac{y^T L(x) y}{y^T y} | y \neq 0, \mathbf{1}^T y = 0\} \qquad (5)$$

Where y is a non zero vector that is orthogonal to the all one vector $\mathbf{1}$. We can reduce Eq (5) into:

$$\mu(L(x)) = min\{\frac{y^T L(x) y}{||y||^2} | y \neq 0, \mathbf{1}^T y = 0\} \qquad (6)$$

If we substitute y with the normalized vector $v = y/||y||$ in Eq (6) then, it can be written as:

$$\mu(L(x)) = min\{v^T L(x) v | \; ||v|| = 1, \mathbf{1}^T v = 0\} \qquad (7)$$

which can also be expressed in the form:

$$\mu(L(x)) = \min\{\sum_{i=1}^{n} \sum_{j \in N_i} (v_i - v_j)^2 | \ ||v|| = 1, \mathbf{1^T} v = 0\} \quad (8)$$

The value of v_i, $i \in V$ are the entries of the Fiedler vector $\vec{v}$ of the Laplacian L. To each node corresponds a Fiedler vector entry which refers to as its Fiedler value of the node. Now the node which contributes most to the algebraic connectivity according to Eq (8) is the one which maximizes the sum of squared differences of the Fiedler values of the nodes compared with its neighbouring nodes i.e $\sum_{j \in N_i} (v_i - v_j)^2$. This observation has led to the suboptimal solution of Eq (1) and its variant of Eq (2). These solutions can be implemented in a distributed manner based on a recently proposed distributed algorithm that calculates the Fiedler vector values of each node using local information only. Their main drawback as indicated by the authors in [12] is that, a maximization consensus algorithm must be employed to find the maximum of the criticality metric among all nodes in the network. This significantly reduces the speed of convergence of the algorithm.

III. PROPOSED ALGORITHM

In this section, we describe the rationale with which we propose a new suboptimal solutions of problem P which is computationally less expensive to implement in a distributed manner. Our approach is based on spectral partitioning concepts.

Spectral partitioning is a method of partitioning a graph into two subgraphs in such a way that, the subgraphs have nearly equal number of vertices while also minimizing the number of edges in-between these two subgraphs [14]. In a network, if $\vec{v} = (v_1, ...v_n)$ is the Fiedler vector of the Laplacian of the graph G, then spectral partitioning finds the suitable value s such that the graph is divided in the form that $v_i > s$ and $v_j \leq s$ with $i + j = n$ and $i \neq j$ [7]. Such a partition is known as the Fiedler cut. A Fiedler cut can be of various forms, a bisection in which s is the median of $\vec{v}$, a ratio cut in which s gives the best ratio cut, a sign cut in which s is equal to zero and a gap cut in which s is the value of the largest gap in the sorted list of Fiedler vector components. In this work, we use the sign cut approach for determining CNs in a network.

The Fiedler vector has both positive and negative entries due to the condition $\mathbf{1^T} v = 0$ of Eq (8). It has been shown that the set of nodes with positive Fiedler values form a subgraph which is well connected and the set of nodes with negative Fiedler value form another subgraph which is also well connected. The two subgraphs between them are poorly connected. So the Fiedler vector values can be used to partition the network into well connected clusters. The sign of each Fiedler vector entry determines to which subgraph the corresponding node belongs, while its magnitude characterizes its position in the associated cluster. A higher magnitude is the evidence of a nodes placement close to the center of a cluster whereas, a lower magnitude is the evidence that the node is placed close to the edge of a cluster [12].
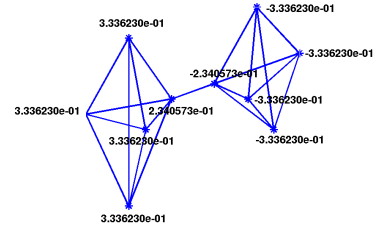

Figure 1. Sample Network.

We demonstrate these concepts through the sample network of Fig 1. The network consists of two well connected clusters. These clusters are poorly connected between them by means of a single link. The Fiedler vector values are calculated and indicated in the diagram. At each node we observe that all the Fiedler values corresponding to the nodes in the left-hand cluster have positive values, whereas, the elements corresponding to nodes in the right-hand cluster have negative values. The most critical nodes are evidently the ones which lie on the boundary of the clusters sharing the link which connects the two clusters. Removal of one of these two nodes will render the network disconnected. We observe that Fiedler vector values for these two nodes have different signs, as they belong to different subgraphs. It can also be observed from the figure that the nodes that lie in the center of these clusters have a higher Fiedler vector magnitude when compared to the nodes that are at the boundary of a cluster.

This demonstrates that nodes which lie on the boundary of the well-connected subgraph and share common links have Fiedler vector values with different signs. We generalize this concept and consider as critical, the nodes which report a Fiedler value which is different from at least one of its neighbours. So in a general graph $G = (V, E)$, a node i is considered as critical it if satisfies the following criterion for some $j \in N_i$:

$$sign(v_i) - sign(v_j) > 0 \quad (9)$$

This approach however can lead to multiple nodes in a network classified as critical. If only one node is to be classified as critical then we choose the one among all nodes which satisfy criterion Eq (9), which maximizes a particular criticality criteria. Different criticality criteria may be selected but along the lines of the previous work conducted, we choose the sum of the squared differences in Fiedler vector values between neighbouring nodes $\sum_{j \in N_i} (v_i - v_j)^2$. Let V' denote the set of nodes which satisfy criterion Eq (9), then:

$$CN = \arg \max_{l \in V'} \sum_{k \in N_l} (v_l - v_k)^2 \quad (10)$$

The suboptimal solution of Eq (10) is less computationally expensive than previous approaches as the maximization algorithm is applied over the restricted set V' instead of the whole node set V. We conjecture that the set V' is connected, forming a bipartite graph. This connectivity enables a distributed maximization consensus algorithm to be employed locally.

The proposed algorithm is amenable for implementation in a distributed fashion. This is achieved by employing a recently proposed distributed algorithm which calculates the Fiedler vector values at each node. This is particularly attractive in our approach as it allows each node to decide locally whether it is a critical node or not. By merely exchanging information with its neighbours each node calculates its Fiedler value and decides whether it is critical by checking if one of its neighbours has Fiedler value with a different sign.

The distributed computation of the Fiedler vector is based on the observation that the Laplacian matrix of a network is implicitly coded inside the network itself and thus for the matrix product $Lx = y$, the kth entry of x that is denoted by x_k is stored inside the node itself and thus by using the $x'_q s$ of its neighbours where $q \in N_k$, x_k^{i+1} can be computed as [12]:

$$x_k^{i+1} = \begin{cases} \frac{1}{\alpha|1-p_k|}(|N_k|x_k^i - \sum_{q \in N_k} x_q^i), & \text{if (i mod N)} = 0 \\ \frac{1}{p_k}((1 - \frac{|N_k|}{\alpha})x_k^i + \frac{1}{\alpha}\sum_{q \in N_k} x_q^i), & \text{otherwise} \end{cases} \tag{11}$$

$$\psi_k^{i+1} = \begin{cases} r_k^i, & \text{if (i mod N)} = 0 \\ \beta_{[1-\frac{\theta}{\alpha},1]}(r_k^i + \frac{x_k^i(p_k x_k^{i+1} - r_k^i x_k^i)}{\max(\delta,(x_k^i)^2)}), & \text{otherwise} \end{cases} \tag{12}$$

Where $\alpha = n$ is a strictly positive constant, i and p are iteration indexes but p is incremented after every few iterations, $\theta = \frac{K}{K-1}\min_{k \in K}|N_k|$ and δ is a small positive value used to avoid division by zero. Here:

$$r_k^{i+1} = \frac{1}{1 + |N_k|}(\psi_k^{i+1} + \sum_{n \in N_k} \psi_n^{i+1}) \tag{13}$$

IV. ANALYSIS

In this section, we derive analytically a lower bound on the algebraic connectivity when nodes are removed from the network under consideration. This bound allows one to evaluate how conservative our suboptimal solution is when multiple nodes are removed from the network.

Theorem 1. *If $G = (V, E)$ is a graph of n nodes with eigenvalues $0 \leq \wedge_2 \leq \wedge_3 \leq ... \leq \wedge_n$, then, upon removal of w of the most critical nodes from the graph, the algebraic connectivity of the resultant graph is lower bounded by:*

$$\lambda \geq \wedge_2 - \frac{u_2^2}{1 + (b_n - u_2^2)/(\wedge_2 - \wedge_n)} \tag{14}$$

where

$$u_2 = \sum_{w \in n} \sum_{j \in N_i, i \in w} (v_i - v_j), \tag{15}$$

$$b_n = n(tr(A) - u_2) + \sqrt{n(1 - n)f(A)} \tag{16}$$

and

$$f(A) = tr\left(A - \frac{tr(A)}{2}I\right)^2 - \left(2\left(u_2 - \frac{tr(A)}{2}\right)^2\right) \tag{17}$$

with

$$tr\left(A - \frac{tr(A)}{2}I\right)^2 = tr(A^2) - \frac{(tr(A))^2}{2} \tag{18}$$

Here, A is the Laplacian matrix defined by the set of nodes w that are being removed from the graph.

Proof: We use the eigenvalue decomposition of $L = QDQ^T$ where $D = Diag(0, \wedge_2, ..., \wedge_n)$ is the diagonal matrix of ascending eigenvalues and Q is an orthogonal matrix with corresponding eigenvectors of L in its columns. The eigenvalues of a Laplacian matrix L can be found using $Lv = \lambda v$, therefore, in this expression we substitute L to get [15]:

$$(QDQ^T)v_j = \wedge_j v_j \tag{19}$$

Where v_j is the linear combination of the eigenvectors corresponding to the jth eigenvalue $\wedge_j$ of L. The removal of w nodes from the network reduces D by a factor uu^T where $u = Q^T H$ and H is the incidence matrix defined by the set of nodes being removed [16]. Thus we have:

$$Q(D - uu^T)Q^T v_j = \wedge_j v_j \tag{20}$$

We know from [17] that, the eigenvalues of Eq (20) can be obtained by solving $D - uu^T - \lambda I$ for the determinant of the matrix, where I is the identity matrix [17]:

$$det(D - uu^T - \lambda I) = 0 \tag{21}$$

$$det(D - \lambda I)det(I - (D - \lambda I)^{-1}uu^T) = 0 \tag{22}$$

Eq (22) can be reduced to [17]:

$$\prod_{i=1}^n (\wedge_i - \lambda)\left(1 - \sum_{i=1}^n \frac{u_i^2}{(\wedge_i - \lambda)}\right) = 0 \tag{23}$$

This shows that, the eigenvalue of Eq (20) can be computed by finding the roots of the secular equation:

$$1 = \sum_{i=1}^n \frac{u_i^2}{\wedge_i - \lambda} \tag{24}$$

We solve Eq (24) for the the eigenvalue λ of the network that results after the removal of w node from the network. Here, we know that $u_1 = 0$ and $u_2 = \sum_{w \in n} \sum_{j \in N_i, i \in w}(v_i - v_j)$. Therefore we have:

$$\frac{u_2^2}{\wedge_2 - \lambda} = 1 - \sum_{i=3}^n \frac{u_i^2}{\wedge_i - \lambda} \tag{25}$$

This can be re-arranged into:

$$\lambda = \wedge_2 - \frac{u_2^2}{1 + \sum_{i=3}^n u_i^2/\lambda - \wedge_i} \tag{26}$$

According to the eigenvalue interlacing theorem, the algebraic connectivity of network that results from the removal of a node is bounded by $0 \leq \lambda_2 \leq \wedge_2$ [18].

Theorem 2. *Let X be a graph with n vertices and let Y be obtained by removing a vertex from X then*

$$\lambda_{i-1}(L(X)) \leq \lambda_i(L(Y)) \leq \lambda_i(L(X))$$

We use Theorem 2 along with the observation in Eq (26), that the LHS is a decreasing function whereas the RHS is an increasing function of λ, therefore we obtain the lower bound of λ by using the appropriate substitution of $\lambda = \wedge_2 > \lambda_2$. This gives us:

$$\lambda \geq \wedge_2 - \frac{u_2^2}{1 + \sum_{i=3}^{n} u_i^2 / (\wedge_2 - \wedge_n)} \tag{27}$$

From [19] we know that $\sum_{i=1}^{n} u_i^2 \leq b_n$, thus we approximate $\sum_{i=3}^{n} u_i^2$ with the difference $b_n - u_2^2$ to obtain the final expression of Eq (14), where:

$$b_n = n(tr(A) - u_2) + \sqrt{n(1-n)f(A)} \tag{28}$$

and $f(A)$ is:

$$f(A) = tr\left(A - \frac{tr(A)}{2}I\right)^2 - \left(2\left(u_2 - \frac{tr(A)}{2}\right)^2\right) \tag{29}$$

In Eq (29) the square of the matrix can be avoided by using Eq (30) [19].

$$tr\left(A - \frac{tr(A)}{2}I\right)^2 = tr(A^2) - \frac{(tr(A))^2}{2} \tag{30}$$

Here $tr(A^2) = ||A||_f^2$ and $||A||_f$ is the Frobenius matrix norm of A. ∎

V. SIMULATION AND RESULTS

In this section, we present the simulation analysis conducted to evaluate the performance of the proposed critical node selection method. A comparative study was conducted against the exhaustive search approach and the suboptimal solutions presented in Eq (1) and Eq (2). In the first set of experiments, we consider a network comprising of two connected clusters which are between them connected via a few bridge links. The emphasis in this first set of experiments is on the particular critical nodes selected by each method. In the next set of experiments, we use a network with randomly distributed nodes to evaluate the performance of the proposed approach in terms of the algebraic connectivity against existing approaches and the lower bound of Eq (14).

In the first set of simulations we use an area of $100 \times 100 m^2$ where 16 nodes are deployed in a way such that two clusters are formed each containing 8 nodes. These two clusters are connected through 6 bridge edges. Nodes 16 and 15 of one cluster are connected to nodes 2, 6, 7 and 8 of the other cluster as shown in Fig 2a. As shown in Table I all methods identify

Selection Mechanism	Nodes identified as critical	
	Network size=16	Network size=40
Exhaustive Search	16	11
Eq (1)	16	11
Eq (2)	16	11
Proposed approach	16	11

Table I.   NODE SELECTION BASED ON DIFFERENT APPROACHES FOR NETWORK SIZE OF 16 AND 40 NODES.

node 16 as the most critical node in the network. However, our approach is able to achieve the correct node selection in the least computationally expensive way.
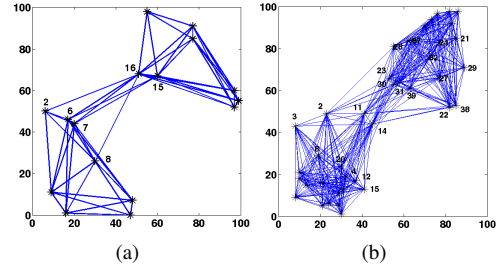

Figure 2.   Network model for a)16 nodes with 6 bridge links, b)40 nodes with 40 bridge links.

We further investigate the performance of the proposed approach by increasing the number of nodes to 40 in the same area of $100 \times 100 m^2$ and among these nodes we form two clusters, each of 20 nodes, but this time the number of links among the two clusters is increased to 40. Despite the change in the network size and the number of bridge links, all methods identify node 11 as the most critical node in the network as reported in Table I.

In the next set of experiments we consider an area of $200 \times 200 m^2$ to deploy 100, 150 and 200 nodes according to uniform random distribution. We consider transmission radius values for the network nodes in the range $50m$ to $150m$ and evaluate the effect of network density on the algebraic connectivity of the network. We then test the algebraic connectivity of the resulting network after a single CN is removed from the network and after five of the most CNs are removed from the network. To avoid random fluctuations due to single simulation run, simulations were conducted for 50 different network topologies and the results were then averaged.

In Fig 3a, we show the algebraic connectivity of the aforementioned network as a function of the transmission radius when the most critical node is removed according to the criticality metric under consideration i.e. the proposed approach, the exhaustive search approach and the suboptimal solutions of Eq (1) and Eq (2). This is done for a network of 100, 150 and 200 nodes. The first thing to note is that, as expected, the algebraic connectivity increases monotonically as the transmission radius increases. The other thing to note is that all suboptimal solutions including the proposed approach, are very close to the algebraic connectivity values reported by the exhaustive search solution, which is the optimal. This demonstrates the ability of the suboptimal solution to approximate the optimal one to a surprisingly good extent. It also demonstrates
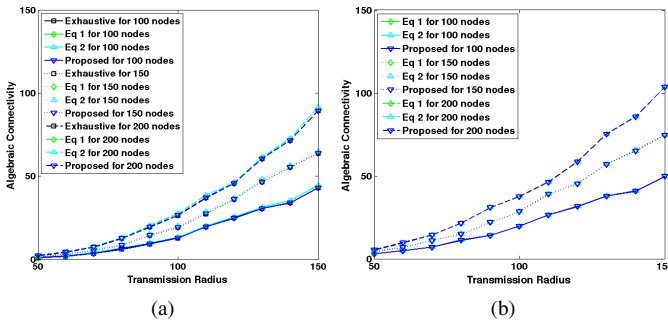
Figure 3.   Algebraic Connectivity versus the trasnsmission radius when: a) a single node is removed from the network, b) five nodes are removed from the network.
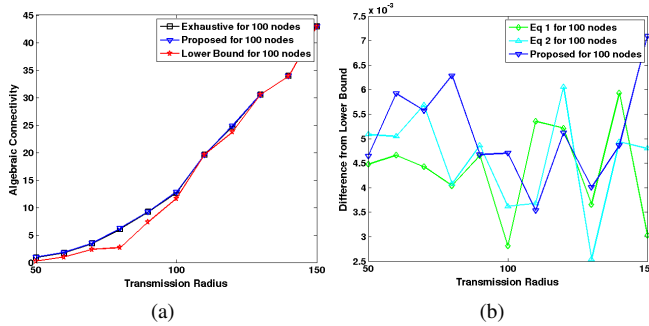

Figure 4.   a) Algebraic connectivity versus transmission radius when a single node is removed from the network, b) Difference between the lower bound and the algebraic connectivity of the network against the transmission radius, after five nodes are removed from the network.

the superiority of our approach which is computationally less expensive to implement without undermining the overall performance. Similar behaviour is observed when removing 5 nodes from the network as shown in Fig 3b.

Fig 4a shows the algebraic connectivity versus the transmission radius when a single node is removed according to the exhaustive search and the proposed method together with the lower bound obtained in section IV. We observe that the lower bound is close to the exhaustive search results which demonstrate that it is not a conservative bound.

Having established that the lower bound is not conservative, we use it to evaluate the performance of the considered approaches when multiple nodes are removed from the network. Note that when multiple nodes are removed, the exhaustive search approach is difficult to implement as its complexity increases combinatorially with network size. So the only reference for comparison is the lower bound obtained. In Fig 4b, we show the difference of the algebraic connectivity values obtained when removing five nodes from the network using the considered criticality metric relative to the lower bound. We observe that the difference is extremely small (of the scale 10^{-3}) demonstrating the ability of the considered solutions to detect the most critical nodes. This again demonstrates the advantage of our approach which can achieve the latter at a lower implementation cost.

VI.   CONCLUSION

In this work, we proposed a new method which uses spectral partitioning concepts for identifying critical nodes in a network. The proposed approach is computationally less expensive to implement than previous proposals and is shown to perform at least equally well using analysis. We also show that the proposed approach is not conservative relative to the best that can be achieved. In the future, we aim at further evaluating the proposed approach using extensive simulations and analysis. We also aim at pursuing connection based enhancements using the same optimization based approach.

REFERENCES

[1] T. H. Grubesic, T. C. Matisziw, A. T. Murray, and D. Snediker, "Comparative approaches for assessing network vulnerability," *International Regional Science Review*, vol. 31, no. 1, pp. 88–112, 2008.

[2] G. Ausiello, D. Firmani, *et al.*, "Real-time monitoring of undirected networks: Articulation points, bridges, and connected and biconnected components," *Networks*, vol. 59, no. 3, pp. 275–288, 2012.

[3] L. C. Freeman, "Centrality in social networks conceptual clarification," *Social networks*, vol. 1, no. 3, pp. 215–239, 1979.

[4] L. C. Freeman, "A set of measures of centrality based on betweenness," *Sociometry*, pp. 35–41, 1977.

[5] M. Everett and S. P. Borgatti, "Ego network betweenness," *Social networks*, vol. 27, no. 1, pp. 31–38, 2005.

[6] P. Yu and H. Van de Sompel, "Networks of scientific papers," *Science*, vol. 169, pp. 510–515, 1965.

[7] M. Fiedler, "Algebraic connectivity of graphs," *Czechoslovak Mathematical Journal*, vol. 23, no. 2, pp. 298–305, 1973.

[8] P. Wei and D. Sun, "Weighted algebraic connectivity: An application to airport transportation network," in *Proceedings of the 18th IFAC World Congress, Milan, Italy*, 2011.

[9] P.-Y. Chen and A. O. Hero, "Local fiedler vector centrality for detection of deep and overlapping communities in networks," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1120–1124, IEEE, 2014.

[10] H. Liu, X. Cao, J. He, P. Cheng, J. Chen, and Y. Sun, "Distributed identification of the most critical node for average consensus," in *World Congress*, vol. 19, pp. 1843–1848, 2014.

[11] T. Watanabe and N. Masuda, "Enhancing the spectral gap of networks by node removal," *Physical Review E*, vol. 82, no. 4, p. 046102, 2010.

[12] A. Bertrand and M. Moonen, "Distributed computation of the fiedler vector with application to topology inference in ad hoc networks," *Signal Processing*, vol. 93, no. 5, pp. 1106–1117, 2013.

[13] B. Mohar and Y. Alavi, "The laplacian spectrum of graphs," *Graph theory, combinatorics, and applications*, vol. 2, pp. 871–898, 1991.

[14] R. Merris, "Laplacian matrices of graphs: a survey," *Linear algebra and its applications*, vol. 197, pp. 143–176, 1994.

[15] D. A. Spielman and N. Srivastava, "Graph sparsification by effective resistances," *SIAM Journal on Computing*, vol. 40, no. 6, pp. 1913–1926, 2011.

[16] A. Ghosh and S. Boyd, "Growing well-connected graphs," in *45th IEEE Conference on Decision and Control*, pp. 6605–6611, IEEE, 2006.

[17] G. H. Golub, "Some modified matrix eigenvalue problems," *Siam Review*, vol. 15, no. 2, pp. 318–334, 1973.

[18] W. H. Haemers, "Interlacing eigenvalues and graphs," *Linear Algebra and its applications*, vol. 226, pp. 593–616, 1995.

[19] O. Rojo, R. Soto, and H. Rojo, "Bounds for sums of eigenvalues and applications," *Computers & Mathematics with Applications*, vol. 39, no. 7, pp. 1–15, 2000.