WSNs Under Attack! How Bad Is It? Evaluating Connectivity Impact Using Centrality Measures

Rodrigo Vieira Steiner, Martín Barrère, Emil Lupu

Department of Computing, Imperial College London, UK
{r.v.steiner, m.barrere, e.c.lupu}@imperial.ac.uk

Keywords: wireless sensor networks, network health, centrality measures, connectivity impact, network security

Abstract

We propose a model to represent the health of WSNs that allows us to evaluate a network's ability to execute its functions. Central to this model is how we quantify the importance of each network node. As we focus on the availability of the network data, we investigate how well different centrality measures identify the significance of each node for the network connectivity. In this process, we propose a new metric named *current-flow sink betweenness*. Through a number of experiments, we demonstrate that while no metric is invariably better in identifying sensors' connectivity relevance, the proposed current-flow sink betweenness outperforms existing metrics in the vast majority of cases.

1 Introduction

Wireless Sensor Networks (WSNs) will play a major role in the Internet of Things (IoT) as sensor nodes collect environmental data enabling system automation and supporting decision-making processes. However, it is highly likely that such network nodes will be eventually attacked. This issue raises an important and challenging question: how bad is it when a particular node or set of nodes gets compromised? The health of a network is not a binary property indicating whether it has been compromised or not, but rather an indicator of how well it can operate in its current state and fulfil its function. This question might be tackled from different perspectives, for example, addressing confidentiality issues, integrity, or availability of the data handled by the network. In this paper, we focus on the latter.

We view the availability of the network data as a connectivity problem. If an adversary compromises a node and forces it to stop reporting its data or relaying data from other nodes, the result would be pretty much the same as disconnecting the compromised node from the network. Moreover, distinct nodes may play different roles in connectivity terms according to their topological position in the network. For example, while some nodes might only be responsible for capturing and sending data at the edge of a WSN, other nodes may also act as relay hubs forwarding a considerable amount of network traffic. Therefore, the compromise of distinct nodes may impact differently on the network health.

In network theory, centrality measures can be used to identify the most important nodes in a network, e.g., degree centrality, closeness centrality, betweenness centrality, among others [1]. Nevertheless, no ultimate centrality measure suits all applications. For instance, betweenness centrality only considers communication along shortest paths. Conversely, current flow betweenness centrality assumes information flows like an electrical current across the entire network. However, WSN traffic usually originates or terminates at the sink node (base station) and does not necessarily go through the shortest paths or spread across all network nodes. In this paper, we evaluate which metric is more suitable to recognize the impact each node has on the connectivity of WSNs. As of today, no previous work has explored this issue in depth. The main contributions of this paper are: (I) a model to represent the health of WSNs; (II) an extensive analysis of centrality measures and their adaptability and suitability in the context of WSNs; and (III) a new centrality measure which we name current-flow sink betweenness.

2 Related Work

Centrality measures have been previously applied in WSNs to fulfil several tasks, such as routing [2, 3, 4], topology control [5, 6, 7, 8], access control [9], connectivity restoration [10], clustering [11], and target tracking [12] to cite a few. While some of these works even propose new centrality measures, none of them perform a comprehensive analysis comparing metrics as shown in this paper. Moreover, none of these works are concerned with measuring the network health and the damage caused by a compromised node.

Labatut and Ozgovde [13] illustrate the applicability of different topological measures for the analysis of WSNs through simulated experiments. However, they only cover one centrality measure, *betweenness*, and an adapted version of it, *sink-betweenness*. Jain and Reddy [14], on the other hand, give a general overview of how some centrality measures could be applied to WSNs. Nonetheless, their work has no analytical evaluation or actual experiments to further support and evidence their claims.

The work of Cartledge et al. [15] is closer to our notion of connectivity damage and network health. Nevertheless, the authors assume that nodes can still meet their responsibilities even if they are part of different disconnected components. This is not the case in WSNs since nodes that are disconnected from the sink will never be able to report their data or receive new commands. Similarly, the well-known work from Albert, Jeong, and Barabási [16], which quantifies damage by measuring the size of the largest connected component, the average size of the remaining components and the mean vertex-vertex distance, is not directly applicable to WSNs as these metrics do not necessarily consider the connection between the sink and sensor nodes.

3 Network Health

Our concept of network health refers to the ability of a network to properly perform its functions. A network operates at maximum health when all of its nodes are working correctly. As nodes fail or are even attacked, the health of the network degrades until it becomes fully dysfunctional. Moreover, distinct nodes might have a different impact on the network operation. We here propose a model that captures this notion and is capable of expressing the health of a WSN into a single value.

3.1 Preliminary Concepts and Notation

Let $W = (V, E)$ be a WSN modelled as an undirected graph where V represents the set of sensors and E corresponds to the set of connectivity edges among nodes. An edge $(v_i, v_j) \in E$ expresses the fact that node v_i is within communication range of node v_j. This relationship is symmetric, meaning edges (v_i, v_j) and (v_j, v_i) are identical. The *order* of the graph is $|V|$, the number of nodes in the network. Whereas the *size* of the graph is $|E|$, the number of edges.

Within a WSN, a path $p = (V', E') \subseteq W$ is a non-empty subgraph of W where $V' = \{v_0, v_1, \ldots, v_k\}$ is a sequence and for each vertex pair $v_i, v_{i+1} \in V'$, $i \in [0, k-1]$, there exists an edge $(v_i, v_{i+1}) \in E' \subseteq E$ that links them.

Finally, we assume a sink-to-sensors/sensors-to-sink communication model. We denote the sink node, also known as base station, by $\mathcal{S}$ and the remaining nodes in the network as V^*, such that $V^* = V - \{\mathcal{S}\}$.

3.2 Model

We specify the *health* of a WSN as a combination of two main aspects: (I) the importance of each node to the network operation, and (II) whether or not the node in question is contributing to the network operation. In particular, a node contributes to the network operation if and only if it is functional and there is a safe path connecting it to the sink. However, a node that is either not functional or that is disconnected from the sink causes damage to the network proportionally to how important such node is.

In this context, we formally define the health of a network $W = (V, E)$ as:

$$\mathcal{H}(W) = 1 - \frac{\sum\limits_{v_i \in V^*} I(v_i) \cdot \Big(1 - \big(F(v_i) \cdot S(v_i)\big)\Big)}{\sum\limits_{v_i \in V^*} I(v_i)}$$

where $I(v_i)$ quantifies the importance of node v_i, $F(v_i)$ is a boolean function indicating if node v_i is functional, and $S(v_i)$ is a boolean function indicating if there is a *safe path* connecting v_i to the sink.

Every node v_i is considered to be functional ($F(v_i)$ = 1) until either it fails (e.g., runs out of energy) or gets compromised. In such cases, $F(v_i)$ will return 0.

We define a *safe path* as a path where all nodes involved are functional. If there is such a path connecting node v_i to the sink then $S(v_i)$ outputs 1, otherwise 0.

Although it is typically assumed that the sink node has an unlimited amount of energy and cannot be compromised, for completeness, it is possible to extend our health definition to consider such scenario:

$$\mathcal{H}'(W) = F(\mathcal{S}) \cdot \mathcal{H}(W)$$

Note that our health definition always returns a value in the interval [0, 1], where 0 denotes a completely dysfunctional network and 1 denotes the network is fully functional. This means that the fraction being subtracted in the equation is effectively the damage sustained by the network.

We can define the importance of a node based on the tasks it performs as well as the role it plays on the network connectivity. Different nodes might execute different operations, such as monitoring dissimilar physical phenomena, working as an actuator, among others. The value of each task is intrinsically dependent on the network application. For simplicity, let us assume that all nodes in the network perform the same tasks or tasks of equivalent values. In this case, a node's importance is defined solely by how it may affect the network at a connectivity level given its topological position. Intuitively, a node that forwards the traffic of a few nodes in the network should be less important than a node that forwards traffic from one dense network segment to another. To this end, we can use centrality measures to identify which nodes are more important. However, as it can be observed in Fig. 1, distinct metrics may give different values for the same node. For the remainder of this paper we investigate which metric is more suitable to our requirements.

4 Centrality Measures

Several centrality measures have been proposed in the literature, and while they all try to identify the most important nodes in a network, each metric has a different concept of what

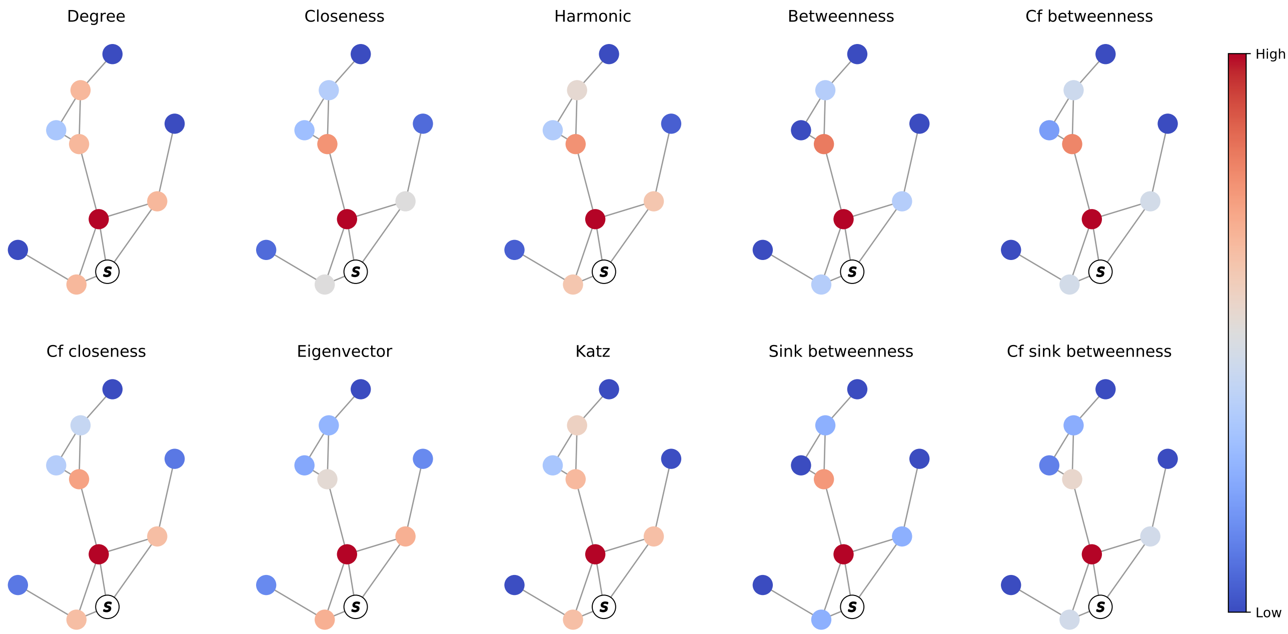

Fig. 1: Node score value coloured according to different centrality measures.

makes a node important. In this section, we give an overview of the most traditional metrics as well as some of the more recently proposed ones, and also introduce a new measure called *current-flow sink betweenness*.

4.1 Degree Centrality

Degree centrality measures the number of direct connections a node has with other nodes in the network [17]. Formally:

$$C_D(v_i) = \sum_{v_i \neq v_j} e(v_i, v_j)$$

where $e(v_i, v_j)$ is 1 if there is an edge connecting v_i to v_j and 0 otherwise.

4.2 Closeness Centrality

Closeness centrality quantifies how close, on average, a node is to all other nodes in the network [17]. It is defined as:

$$C_C(v_i) = \frac{1}{\sum_{v_i \neq v_j} d(v_i, v_j)}$$

where $d(v_i, v_j)$ is the distance — number of edges in a shortest path — between v_i and v_j. Note that if there is no path between two nodes, the distance is infinite by convention. Thus, this metric is not suitable in such scenarios.

4.3 Harmonic Centrality

With the same goal of closeness centrality, but still applicable to disconnected graphs, harmonic centrality is defined as the sum of the inverted distances rather than the inverted sum of distances [18]. Formally:

$$C_H(v_i) = \sum_{v_i \neq v_j} \frac{1}{d(v_i, v_j)}$$

where $1/\infty$ (when there is no path from v_i to v_j) is 0.

4.4 Betweenness Centrality

Betweenness centrality measures the number of times a node appears on the shortest paths between all pairs of nodes in the network [17]. It is formally defined as:

$$C_B(v_i) = \sum_{v_i \neq v_j \neq v_k} \frac{\sigma(v_j, v_k | v_i)}{\sigma(v_j, v_k)}$$

where $\sigma(v_j, v_k)$ is the total number of shortest paths from v_j to v_k and $\sigma(v_j, v_k | v_i)$ is the number of those paths that pass through v_i.

4.5 Current-flow Betweenness Centrality

Whereas betweenness centrality assumes information flows over shortest paths, current-flow betweenness centrality adopts a model where information spreads as an electrical current [19]. Formally:

$$C_{CB}(v_i) = \sum_{v_i \neq v_j \neq v_k} \tau(v_j, v_k | v_i)$$

where $\tau(v_j, v_k|v_i)$ is the amount of current that flows through v_i when a unit of current is injected at vertex v_j and extracted at vertex v_k. This metric is also known as random-walk betweenness centrality.

4.6 Current-flow Closeness Centrality

While closeness centrality measures distance, current-flow closeness centrality quantifies the absolute potential differences among the nodes in the network [20]. It is defined as:

$$C_{CC}(v_i) = \frac{1}{\sum_{v_i \neq v_j} \hat{p}(v_i, v_j | v_i, v_j)}$$

where $\hat{p}(v_i, v_j|v_i, v_j)$ quantifies the effective resistance between v_i and v_j when a unit of current is injected at vertex v_i and extracted at vertex v_j. This metric is equivalent to information centrality [21].

4.7 Eigenvector Centrality

Whilst degree centrality simply counts the number of connections of a node, eigenvector centrality assigns proportional scores to a node based on the scores of its neighbours [22]. Formally:

$$C_E(v_i) = \frac{1}{\lambda} \sum_{v_i \neq v_j} e(v_i, v_j) C_E(v_j)$$

where λ is a constant. Given the adjacency matrix A of the network, such that $a_{i,j} = e(v_i, v_j)$, and defining a vector $x = (C_E(v_1), C_E(v_2), ...)$ one can rewrite the previous equation as:

$$Ax = \lambda x$$

which allows us to observe that x is the eigenvector of A with eigenvalue λ.

4.8 Katz Centrality

Katz centrality also defines the score of a node based on its neighbours and can be viewed as a generalization of the eigenvector centrality [23]. It is defined as:

$$C_K(v_i) = \alpha \sum_{v_i \neq v_j} e(v_i, v_j) C_E(v_j) + \beta$$

where α is as an attenuation factor penalizing the contribution of distant nodes while β can give extra weight to immediate neighbors.

4.9 Sink Betweenness Centrality

Sink betweenness is an adaptation of betweenness centrality explicitly designed for WSNs [2]. Since the communication in such networks is usually between sensor nodes and the sink, instead of considering all pair of nodes, this metric only considers the shortest paths between each node and the sink. It is formally defined as:

$$C_{SB}(v_i) = \sum_{v_i \neq v_j \neq \mathcal{S}} \frac{\sigma(v_j, \mathcal{S}|v_i)}{\sigma(v_j, \mathcal{S})}$$

where $\mathcal{S}$ is the sink node.

4.10 Current-flow Sink Betweenness Centrality

In the same way sink betweenness adapts betweenness centrality, we here propose to adapt the current-flow betweenness centrality to only take into account the paths terminating at the sink. Formally:

$$C_{CSB}(v_i) = \sum_{v_i \neq v_j \neq \mathcal{S}} \tau(v_j, \mathcal{S}|v_i)$$

4.11 Discussion

One limitation of centrality measures is that the characteristics used to identify the most important nodes might not generalize to the remaining nodes in the network. This means that rankings might be meaningless for less important nodes [24]. Another limitation of centrality measures is that while they rank nodes by importance, they do not specify the difference in importance over distinct rank levels. However, for all centrality measures, we can compute the Freeman centralization measure [17], which gives a notion of node importance based on the differences of their scores.

5 Experimental Evaluation

Since distinct centrality measures have different interpretations of what makes a node important, it is usually the case that the best metric for a given application is not as good for other operations. Our objective is to analyse the performance of such metrics in ranking the impact that each node has on the network connectivity. However, there is no standard procedure to carry out an analysis of this kind. Therefore, we performed three different experiments, as described in this section. In all experiments, we used the NetworkX [25] python package to generate the network graphs and compute the centrality measures scores.

5.1 Network Model

While centrality measures are usually evaluated over the Erdős-Rényi or Barabási-Albert models, in this paper, we use

Table 1: Mean correlation coefficients between metrics and ranking by counting paths.

$\|V\|$	Centrality	Kendall τ	Spearman ρ
15	Degree	0.192659	0.209119
	Closeness	**0.313026**	**0.365837**
	Harmonic	0.259808	0.302857
	Betweenness	0.293757	0.360487
	Cf betweenness	0.286593	0.360374
	Cf closeness	0.276209	0.312523
	Eigenvector	0.190848	0.180138
	Katz	0.185320	0.194818
	Sink betweenness	**0.333496**	**0.390212**
	Cf sink betweenness	**0.481101**	**0.577486**

Table 2: Mean correlation coefficients between metrics and ranking by counting node independent paths.

$\|V\|$	Centrality	Kendall τ	Spearman ρ
50	Degree	0.117484	0.147835
	Closeness	0.254139	0.327075
	Harmonic	0.220759	0.287228
	Betweenness	**0.408120**	**0.511300**
	Cf betweenness	**0.408739**	**0.516205**
	Cf closeness	0.227013	0.292216
	Eigenvector	0.122902	0.163820
	Katz	0.095954	0.124506
	Sink betweenness	**0.408180**	**0.496516**
	Cf sink betweenness	**0.516054**	**0.632177**

Random Geometric Graphs (RGG) as they are a more realistic representation of WSNs [26, 27, 28]. Nodes are placed uniformly at random in the unit cube, and any two nodes are connected by an edge if the distance between them is within a transmission radius r. Other than requiring the network to be connected, we do not impose any other restrictions on the network structure, such as a minimum k-connectivity, presence or absence of cycles, or centralizing the sink node.

5.2 Counting paths

When we examine a WSN, we are not particularly interested in the connection between any two specific nodes, but rather we are concerned about nodes being connected to the sink. Therefore, one way of measuring the network connectivity is by counting the total number of paths connecting the sink to each node in the network, let us name this value T_{P_S}. We can then temporarily disconnect each node, one at a time, to recompute the number of paths between the sink and the other nodes, which we denote by R_{P_S}. Finally, we can rank the nodes such that node i outranks node j if $T_{P_S} - R_{P_{S_i}} > T_{P_S} - R_{P_{S_j}}$. Consequently, a node whose disconnection removes more paths is considered more important. In practice, we do not need to compute T_{P_S} and can merely give higher ranks to nodes with smaller R_{P_S} values.

For this experiment, we assume the ranking obtained by counting paths as the ground truth and analyse the correlation with each ranking given by the centrality measures described in Section 4. We do this computation over 100 random geometric graphs of order 15. In particular, we compute the Kendall's τ and the Spearman ρ correlation coefficients. Both values range in the interval $[-1, 1]$ indicating strong disagreement around -1, no correlation around 0, and strong agreement around 1. Thus, the higher the correlation value, the better is the metric.

The mean correlation coefficients over the 100 graphs can be seen in Table 1. We can observe that the current-flow sink betweenness measure presents the highest correlation, followed by sink betweenness and closeness centrality, while Katz, eigenvector, and degree centrality present the lowest values. Also, except for the eigenvector centrality, the Spearman ρ is higher than Kendall's τ for all metrics. Typically, Spearman correlation coefficients tend to be higher than Kendall's. However, Spearman is more sensitive to large discrepancies even if they occur in a small amount. Which means that the eigenvector centrality is probably giving low ranks to some nodes that receive high ranks when counting paths or vice-versa.

There are two issues when counting all paths connecting the sink to network nodes. The first issue is that counting all these paths can be really expensive — which is the reason why this experiment is only performed over graphs with 15 nodes. While a single path can be found in $O(V + E)$ time, the number of paths in a network can be enormous, for instance, $O(n!)$ for a complete graph of order n. The second issue is that not all paths contribute to the sink connectivity. Let us consider the paths $\{\mathcal{S}, v_1, v_3\}$ and $\{\mathcal{S}, v_1, v_2, v_3\}$. Node v_2 is not giving the sink an alternative independent path to v_3 since it still requires node v_1 which is already considered in the first path.

5.3 Counting node independent paths

Any two paths in the network that connect the same two non-adjacent nodes are said to be *node-independent* if they do not have any internal nodes in common. According to Menger's theorem, the number of node-independent paths between two nodes is always equal to the minimum number of nodes that must be removed to disconnect them.

This experiment is the same as the previous one, except this time we rank a node n by counting the remaining number of independent paths between the sink and each other network node when n is temporarily disconnected. Table 2 shows the mean correlation coefficients over 100 graphs of order 50. It is possible to notice that, in comparison to the previous experiment, the correlation of some metrics increased while others decreased. Current-flow sink betweenness still has the highest correlation, but this time is followed by current-flow betweenness, betweenness, and sink betweenness, these three presenting very similar results. Katz, eigenvector, and degree centrality measures still exhibit the lowest correlations. This time the Spearman ρ is higher than Kendall's τ for all metrics.

Although counting independent paths is less expensive than counting all paths, this method still does not scale to large networks. We can improve its performance by using an approximation algorithm that gives a lower bound on the number of node-independent paths between two nodes [29]. Table 3 shows the results when using this algorithm over 100 graphs

Table 3: Mean correlation coefficients between metrics and ranking by counting an approximation of node independent paths.

$\|V\|$	Centrality	Kendall τ	Spearman ρ
50	Degree	0.130927	0.165050
	Closeness	0.249960	0.321991
	Harmonic	0.226517	0.295999
	Betweenness	**0.390498**	**0.491875**
	Cf betweenness	**0.400498**	**0.506592**
	Cf closeness	0.221726	0.286106
	Eigenvector	0.138828	0.184423
	Katz	0.115615	0.151474
	Sink betweenness	**0.391404**	**0.477312**
	Cf sink betweenness	**0.511963**	**0.626929**
100	Degree	0.065708	0.089039
	Closeness	0.217286	0.290192
	Harmonic	0.187782	0.254746
	Betweenness	**0.340509**	**0.438525**
	Cf betweenness	**0.343290**	**0.441455**
	Cf closeness	0.198794	0.267816
	Eigenvector	0.089037	0.122594
	Katz	0.031533	0.049557
	Sink betweenness	**0.330092**	**0.416705**
	Cf sink betweenness	**0.457480**	**0.575776**

of order 50 and 100 graphs of order 100. The results on graphs of order 50 are very similar to the ones achieved without the approximation. Whereas the correlation of every metric decrease on graphs of order 100, the analysis remains the same, thus indicating a similar behaviour on larger graphs.

5.4 Disconnecting nodes

Another way of analysing the performance of different centrality measures is to simulate an attack that disconnects nodes sequentially (following the ranking provided by each metric) and observe the effects on the network [16, 15].

In the context of WSNs, we are interested in monitoring the number of nodes that remain connected to the sink. If the network gets fragmented into multiple components, only the nodes connected to the sink will continue to be functional since they will be the only ones capable of reporting their data and receiving new instructions. Accordingly, we assume an attack strategy that always disconnects the highest ranking node within the sink component until the sink is isolated. Fig. 2 illustrates this process for a graph of order 50. Note that the goal is not to simply isolate the sink as fast as possible. If that was the case, one could merely disconnect all of the sink's neighbours and be done with it. There is a subtle difference. For instance, a node that is only connected to the sink and nothing else has no impact on the connectivity of the other nodes in the network. We are looking for a metric that is capable of identifying the most important nodes and, in the average case, this metric will be able to isolate the sink faster (requiring fewer disconnections) than others.

There are two possible disconnection strategies. The first one computes the centrality scores only once before removing any node. The second strategy recomputes the centrality scores for the remaining nodes of the sink component after each disconnection. We evaluate both strategies over graphs of different

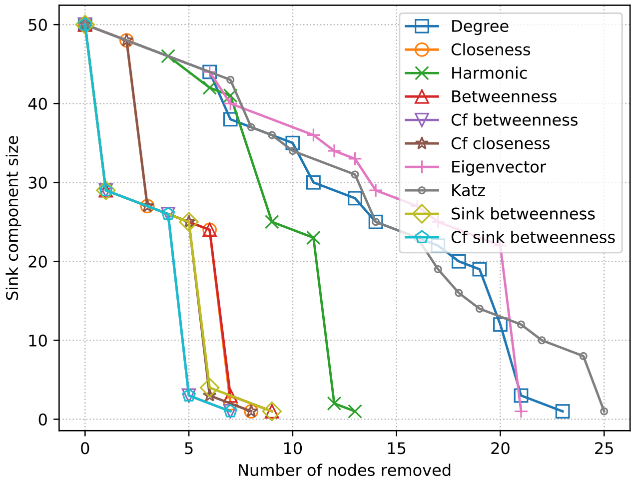

Fig. 2: Sink component size as nodes are removed by highest score according to each centrality measure.

orders and the results are shown in Table 4 and Table 5. We compute the mean (μ) and standard deviation (σ) of the number of disconnected nodes and also rank the metrics among themselves (columns 1st, 2nd, ..., 10th) according to the number of nodes they need to disconnect to isolate the sink in each graph. Note that there can be ties, for instance, in the example illustrated in Fig. 2, the rank would be: 1st current-flow sink betweenness and current-flow betweenness, 3rd current-flow closeness and closeness, 5th betweenness and sink betweenness, 7th harmonic, 8th eigenvector, 9th degree, and 10th Katz. The ranking fields of each row sum up to 100[1] while column sums might exceed 100 due to ties.

While the mean and standard deviation allow us to think about the average case, the ranking allows us to see that no metric is *always* better (fewer disconnections for sink isolation) than the others. However, we can observe that when using the first strategy, current-flow sink betweenness outperforms the other measures in most cases. In the second strategy, current-flow sink betweenness is slightly worse than sink betweenness for graphs of order 50 and 100, but is slightly better for graphs of order 500. We can also notice that when using the second strategy, fewer nodes have to be disconnected to isolate the sink. This approach is more time-consuming though, since the metric has to be recomputed several times. Therefore, choosing a particular strategy involves a trade-off between efficiency and computation power.

6 Conclusions and Future Work

WSNs will play a fundamental role in the IoT era and therefore, understanding how these networks can operate in the presence of faulty or compromised nodes is vital. In this paper,

[1] Except for the Eigenvector and Katz centrality measures which may fail to converge after a maximum number of iterations and are, in such cases, not ranked. We used the default parameters from NetworkX — 100 and 1000 maximum iterations respectively.

Table 4: Disconnecting nodes using the first strategy.

$\|V\|$	Centrality	μ	σ	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
50	Degree	17.4	5.0	3	2	8	7	10	12	8	27	17	6
	Closeness	15.5	4.9	8	4	5	5	16	21	17	14	9	1
	Harmonic	15.6	4.5	5	5	7	12	20	16	24	5	5	1
	Betweenness	14.0	4.6	2	3	25	21	7	10	14	4	7	7
	Cf betweenness	12.8	4.1	2	10	32	22	10	9	5	6	3	1
	Cf closeness	14.8	4.9	8	8	14	11	17	20	7	11	4	0
	Eigenvector	18.9	7.1	13	1	2	1	2	4	4	9	8	28
	Katz	17.7	5.6	9	4	10	3	4	5	5	18	29	12
	Sink betweenness	10.1	4.6	35	32	10	6	3	5	3	5	1	0
	Cf sink betweenness	**8.2**	**3.4**	**79**	**12**	**5**	**1**	**1**	**2**	**0**	**0**	**0**	**0**
100	Degree	33.9	8.0	0	3	4	5	7	10	6	32	25	8
	Closeness	28.2	10.8	1	7	8	6	20	17	12	10	16	3
	Harmonic	28.3	9.5	2	5	5	9	15	18	36	5	5	0
	Betweenness	23.6	8.5	0	3	16	33	10	10	10	8	10	0
	Cf betweenness	19.9	7.0	0	15	44	19	9	7	4	2	0	0
	Cf closeness	26.4	10.1	5	10	7	14	16	22	15	5	5	1
	Eigenvector	34.4	14.5	5	1	4	3	4	1	3	12	7	17
	Katz	33.7	10.6	2	4	4	5	4	8	7	19	23	6
	Sink betweenness	16.9	8.7	18	41	13	9	6	5	3	3	2	0
	Cf sink betweenness	**10.8**	**5.2**	**88**	**7**	**3**	**1**	**0**	**1**	**0**	**0**	**0**	**0**
500	Degree	204.8	54.8	0	1	2	1	22	14	16	38	6	0
	Closeness	203.2	72.5	0	3	3	3	8	11	22	43	7	0
	Harmonic	190.9	68.2	0	1	4	9	8	26	40	10	2	0
	Betweenness	134.0	47.5	0	0	7	56	15	12	7	2	1	0
	Cf betweenness	83.9	32.9	0	39	48	8	3	2	0	0	0	0
	Cf closeness	172.3	57.6	0	2	7	10	43	28	8	2	0	0
	Eigenvector	240.5	95.6	0	2	2	2	1	1	2	5	29	0
	Katz	107.0	0.0	0	0	0	1	0	0	0	0	0	0
	Sink betweenness	79.9	52.7	1	52	27	10	4	5	1	0	0	0
	Cf sink betweenness	**17.9**	**11.2**	**100**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Table 5: Disconnecting nodes using the second strategy.

$\|V\|$	Centrality	μ	σ	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th
50	Degree	17.0	4.2	0	0	2	1	1	4	10	46	36	0
	Closeness	11.4	2.9	6	4	26	7	23	24	6	2	1	1
	Harmonic	13.9	3.3	0	2	6	2	3	14	66	7	0	0
	Betweenness	10.1	2.7	3	5	58	14	14	6	0	0	0	0
	Cf betweenness	9.9	2.7	3	5	63	15	13	0	1	0	0	0
	Cf closeness	10.9	2.4	6	6	34	8	30	14	1	1	0	0
	Eigenvector	10.0	0.0	0	0	0	0	0	1	0	0	0	0
	Katz	16.8	3.6	0	0	1	1	0	1	9	51	33	0
	Sink betweenness	**7.0**	**3.1**	**89**	**7**	**1**	**1**	**1**	**1**	**0**	**0**	**0**	**0**
	Cf sink betweenness	**7.0**	**3.0**	**85**	**12**	**2**	**1**	**0**	**0**	**0**	**0**	**0**	**0**
100	Degree	33.7	7.0	0	0	0	0	0	0	4	65	31	0
	Closeness	18.1	4.6	0	2	9	8	29	47	5	0	0	0
	Harmonic	25.3	6.3	0	0	0	1	3	3	90	3	0	0
	Betweenness	14.3	4.0	1	2	50	35	11	1	0	0	0	0
	Cf betweenness	14.3	4.2	3	1	67	14	9	5	1	0	0	0
	Cf closeness	16.8	4.0	0	2	13	11	54	20	0	0	0	0
	Eigenvector	-	-	0	0	0	0	0	0	0	0	0	0
	Katz	32.9	7.4	0	0	0	0	0	0	1	38	39	0
	Sink betweenness	**8.8**	**3.8**	**87**	**13**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
	Cf sink betweenness	**9.1**	**4.0**	**73**	**25**	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
500	Degree	197.3	31.9	0	0	0	0	0	0	4	96	0	0
	Closeness	70.1	17.2	0	0	0	0	65	35	0	0	0	0
	Harmonic	139.2	30.5	0	0	0	0	0	0	96	4	0	0
	Betweenness	40.1	11.3	0	0	33	67	0	0	0	0	0	0
	Cf betweenness	36.9	10.3	0	1	72	26	1	0	0	0	0	0
	Cf closeness	77.0	22.0	0	0	1	1	33	65	0	0	0	0
	Eigenvector	-	-	0	0	0	0	0	0	0	0	0	0
	Katz	170.0	11.0	0	0	0	0	0	0	0	0	2	0
	Sink betweenness	**14.3**	**6.7**	**62**	**38**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
	Cf sink betweenness	**13.5**	**6.1**	**78**	**22**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

we have proposed a model capable of expressing the health of a WSN as a single value in the range $[0, 1]$, where 0 means the network is not operational and 1 means it is working to its maximum capacity. A fundamental aspect to this model is the significance or importance attributed to each node according to how it affects the network connectivity based on its topological position. To this end, we have performed an extensive analysis on how well centrality measures can rank the impact that each sensor node has on the connectivity of the network. We have conducted three different experiments which involved counting paths, independent paths, and disconnecting nodes from the network according to the metrics scores. Our results show that no metric is superior in all cases. However, we have proposed a novel metric called *current-flow sink betweenness* which is able to outperform existing metrics in most of the cases.

As future work, we plan to instantiate our health model over topologies from real-world WSNs. Moreover, we plan to analyse the relationship between the connectivity impact estimation and routing algorithms in terms of packet delivery ratio, end-to-end delay, and network lifetime (energy consumption). We also plan to extend our model to cover cases with multiple sinks and where nodes can have different levels of functionality rather than a boolean functional/not functional representation. Additionally, we aim at incorporating integrity and confidentiality aspects into the proposed health model as part of a broader security metric framework for WSNs. Finally, novel graph theoretical models such as core graphs and core paths [30, 31] can underpin promising improvements to estimate the connectivity impact of wireless sensor nodes.

Acknowledgements

This work was supported by the Engineering and Physical Sciences Research Council (EPSRC), UK, under grant number EP/L022729/1, and the Brazilian National Council for Scientific and Technological Development (CNPq) under Grant No. 206707/2014-2 awarded to Rodrigo Vieira Steiner.

References

[1] M van Steen. *Graph Theory and Complex Networks: An Introduction*. Maarten van Steen, April 2010.

[2] E MR Oliveira, H S Ramos, and A AF Loureiro. Centrality-based routing for wireless sensor networks. In *IFIP Wireless Days*, pages 1–5. IEEE, 2010.

[3] XH Li and ZH Guan. Energy-aware routing in wireless sensor networks using local betweenness centrality. *International Journal of Distributed Sensor Networks*, 9(5):307038, 2013.

[4] A Jain. Betweenness centrality based connectivity aware routing algorithm for prolonging network lifetime in wireless sensor networks. *Wireless Networks*, 22(5):1605–1624, 2016.

[5] A Cuzzocrea, A Papadimitriou, D Katsaros, and Y Manolopoulos. Edge betweenness centrality: A novel algorithm for qos-based topology control over wireless sensor networks. *Journal of Network and Computer Applications*, 35(4):1210–1217, 2012.

[6] L Sitanayah, K Brown, and C Sreenan. Fault-tolerant relay deployment based on length-constrained connectivity and rerouting centrality in wireless sensor networks. *Wireless Sensor Networks*, pages 115–130, 2012.

[7] H S Ramos, A Boukerche, A LC Oliveira, A C Frery, E MR Oliveira, and A AF Loureiro. On the deployment of large-scale wireless sensor networks considering the energy hole problem. *Computer Networks*, 110:154–167, 2016.

[8] X Fu, Y Yang, W Li, and G Fortino. Topology upgrading method for energy balance in scale-free wireless sensor networks. In *14th International Conference on Networking, Sensing and Control*, pages 192–197. IEEE, 2017.

[9] J Duan, D Gao, C Heng Foh, and H Zhang. Tc-bac: A trust and centrality degree based access control model in wireless sensor networks. *Ad Hoc Networks*, 11(8):2675–2692, 2013.

[10] I F Senturk and K Akkaya. Connectivity restoration in disjoint wireless sensor networks using centrality measures. In *39th Conference on Local Computer Networks Workshops*, pages 616–622. IEEE, 2014.

[11] A Jain and BV R Reddy. Eigenvector centrality based cluster size control in randomly deployed wireless sensor networks. *Expert Systems with Applications*, 42(5):2657–2669, 2015.

[12] N Meghanathan. An eigenvector centrality-based mobile target tracking algorithm for wireless sensor networks. *International Journal of Mobile Network Design and Innovation*, 6(4):202–211, 2016.

[13] V Labatut and A Ozgovde. Topological Measures for the Analysis of Wireless Sensor Networks. *Procedia Computer Science*, 10:397–404, 2012.

[14] A Jain and BVR Reddy. Node centrality in wireless sensor networks: Importance, applications and advances. In *IEEE 3rd International Advance Computing Conference (IACC)*, pages 127–131. IEEE, 2013.

[15] C L Cartledge and M L Nelson. Connectivity damage to a graph by the removal of an edge or a vertex. *arXiv preprint arXiv:1103.3075*, 2011.

[16] R Albert, H Jeong, and AL Barabási. Error and attack tolerance of complex networks. *Nature*, 406(6794):378–382, 2000.

[17] L C Freeman. Centrality in social networks conceptual clarification. *Social networks*, 1(3):215–239, 1978.

[18] Y Rochat. Closeness centrality extended to unconnected graphs: The harmonic centrality index. In *ASNA*, number EPFL-CONF-200525, 2009.

[19] M EJ Newman. A measure of betweenness centrality based on random walks. *Social networks*, 27(1):39–54, 2005.

[20] U Brandes and D Fleischer. Centrality measures based on current flow. In *STACS*, volume 3404, pages 533–544. Springer, 2005.

[21] K Stephenson and M Zelen. Rethinking centrality: Methods and examples. *Social networks*, 11(1):1–37, 1989.

[22] P Bonacich. Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2(1):113–120, 1972.

[23] L Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18(1):39–43, 1953.

[24] G Lawyer. Understanding the influence of all nodes in a network. *Scientific reports*, 5:8665, 2015.

[25] A Hagberg, P Swart, and D S Chult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory, 2008.

[26] L Lima and J Barros. Random walks on sensor networks. In *5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops*, pages 1–5. IEEE, 2007.

[27] H Kenniche and V Ravelomananana. Random geometric graphs as model of wireless sensor networks. In *Computer and Automation Engineering*, volume 4, pages 103–107. IEEE, 2010.

[28] H Zheng, F Yang, X Tian, X Gan, X Wang, and S Xiao. Data gathering with compressive sensing in wireless sensor networks: a random walk based approach. *IEEE Transactions on Parallel and Distributed Systems*, 26(1):35–44, 2015.

[29] D R White and M Newman. Fast approximation algorithms for finding node-independent paths in networks. 2001.

[30] M Barrère, R V Steiner, R Mohsen, and E C Lupu. Tracking the Bad Guys: An Efficient Forensic Methodology To Trace Multi-step Attacks Using Core Attack Graphs. In *Proceedings of the 13th IEEE International Conference on Network and Service Management*, Nov 2017.

[31] M Barrère and E C Lupu. Naggen: a Network Attack Graph GENeration Tool. In *Proceedings of the IEEE Conference on Communications and Network Security*, Oct 2017.