Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2015, Article ID 873941, 11 pages
http://dx.doi.org/10.1155/2015/873941

Review Article

Avoiding the Opportunist: The Role of Simmelian Ties in Fostering the Trust in Sensor-Cloud Networks

Ming Xiang,[1] William Liu,[1] Quan Bai,[1] and Adnan Al-Anbuky[2]

[1]*School of Computer and Mathematical Sciences, Auckland University of Technology, Auckland 1010, New Zealand*
[2]*Sensor Network and Smart Environment (SeNSe) Research Laboratory, School of Engineering, Auckland University of Technology, Auckland 1010, New Zealand*

Correspondence should be addressed to Ming Xiang; sxiang@aut.ac.nz

Received 17 April 2015; Accepted 2 August 2015

Academic Editor: Ana Alejos

The wireless sensor-cloud networks (WSCNs) are becoming popular nowadays. The new concept of trust has emerged in recent studies as an alternative mechanism to address the security concern in WSCN. Most of the studies on trust are focusing on how to model and evaluate trust so as to effectively detect any malicious activity in the network and then isolate and avoid them. In addition, WSCNs are very dynamic and flexible, thus being hard to keep a static network topology and connectivity which bring more challenges to be secured. In this paper, we have introduced the new angle of adaptive network approach to discover the interplay between network node's trust evaluation and its underlying topology change. It has been found that the network connectivity change will also have strong impact on the trust behavior running over it. Moreover, inspired from the trust studies in sociology, we propose that the Simmelian tie structured networks enable more positive impact on fostering trustworthiness among wireless sensor nodes, but the structural hole characterized networks provide more opportunity for misbehaviors and have negative impact on securing the sensor-cloud networks. The extensive simulation studies have confirmed our new concepts and validated our hypothesis

1. Introduction

The rapid growth in Information and Communication Technologies (ICT) makes many network applications become feasible. One of the popular applications today is Internet of Things (IoT) and it is becoming the emerging paradigm of the future Internet [1]. Wireless Sensor Network (WSN) is one of pillar stones to enable IoT [2]. WSN is composed of sensor nodes and they are connected wirelessly without a static network topology, and this makes the WSN become flexible and easy to be deployed in various environments according to the different applications. WSN is normally used as sensor monitor network such as security, military sensing, and intelligent environment monitoring [3].

Cloud computing is a popular concept nowadays that it is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources [4]. So as to be able to fully exploit its power, a seamlessly integration with our physical life is necessary [5]. As mentioned before, WSNs can seamlessly integrate the digital world with physical world that the integration of WSN and cloud computing which is also named Wireless Sensor-Cloud Network (WSCN) is becoming popular nowadays [5–7]. The sensors in the networks are collecting data and transmitting them to the backend cloud servers, and the cloud is enabled with graphic user interface (GUI) and visualization tools to easily allow users to query, store, and analyse the data without requiring any knowledge of the backend [7].

On the other hand, each sensor in WSCN is self-organised and connected with control center by using wireless connection, sensor exposition in an open environment, and so forth. All these features of WSCN make the network entities and their information flows easily accessed by the malicious parties. The security issues are crucial in WSCN environment. Compared to the traditional cable network, the computing resources such as capacity, processing, and battery are limited in sensor nodes in WSCN; the traditional security mechanisms such as heavy computing oriented encryption mechanism need to be tailored to apply on this

situation. Moreover, as WSCN is self-organized distributed network the traditional trust infrastructure such as Public Key Infrastructure (PKI) is no longer suitable. The distributed trust management is a popular solution for this nowadays. Each node in the WSCN is an independent entity to evaluate its neighbors on trust by their previous interactive experience and indirect reputation from other nodes. It normally selects the neighbor node with highest trustworthiness to forward data to the destination [8–13].

In this paper, we are interested in discovering how the underlying network topology can affect the overlay trust behavior in a WSCN. The rest of the paper is organized as follows. Section 2 reviews the definition of trust in different disciplines, the state of art of distributed trust and reputation management research, adaptive networks, and the pillar stone structural concepts of Simmelian tie and structural hole in sociology. The metrics such as clustering coefficient and effective size are introduced to characterize and measure the Simmelian tie and structural hole in Section 3. A trust and reputation evaluation algorithm for end-to-end routing in WSCN, as a model of overlay trust behavior, has been proposed in Section 4. Section 5 has conducted two simulation case studies to discover how the Simmelian tie and structural hole affect trust evaluation behavior regarding malicious attacks. Finally in Section 6 we conclude the significant findings and also lay out the future works.

2. Related Works

The traditional security mechanisms such as cryptography are mainly focused on providing data integrity, confidentiality, access control, node authentication, and so forth. All these can make sure that the nodes in the network are well protected from malicious parties from outside, but it is not very effective to protect against the malicious activities from inside. For example, legitimate nodes in the network behave selfishly that they refuse to forward packets (black-hole attack) or some of the packets (grey-hole attack). In such situation, the distributed trust management system is a more effective way to protect the network availability.

2.1. Trust Definition. Trust is a very important factor in our everyday life as all relationships rely on the trust in the human society, and each interaction with other people involves trust as well. The study in [14] has described the formation of human friendship as ego observes his own behavior and that of alters during the interaction and evaluates this in terms of his own values, norms, interests, and so forth. The ego pays attention not only to interaction in which he plays part by himself, but also to interaction among alters. The more information ego has collected about all alters, the more reliable not only ego's estimation of the suitability of alters in terms of continuation of the relationship, but also his estimation of the willingness of alters to reciprocate his personal interest. If alters have positive match of ego's expectation, ego will invest more time and effort to interact with these alters so as to enable these friendly relationships becoming more reliable friendship. The study [15] suggested if the ego has more trust alters in the past, then he will more likely trust more in the present, and vice versa. Also, if the ego hears more trust from 3rd party about alters, it has more trust in alters as well. But when the ego hears distrust from 3rd party, it will have more impact than the impact of good reputation from 3rd party. These two studies have implied that human relationship is building on trust which is based on the previous direct interaction and also indirect reputation from 3rd parties.

Trust is important not only in the human society, but also in the computer security area. Taking the public key cryptography as an example, it requires the key that can only be accessed by the authorized person; otherwise this security mechanism becomes compromised which involves the trust in both sides of the communications.

2.2. Modelling Trust in Computer Network. Trust in computer world can be interpreted as the expectation of other machines' performance and being without malicious attacks, and also this expectation will become the experience shared through networks of machines [16, 17]. We can simply translate this definition into two metrics which are direct trust and indirect trust (i.e., reputation or recommendation). Direct trust is the direct experience with target and reputation is the comments from neighbors about the target. Many studies on trust in computing are including these two metrics to measure the trust. Moreover, the paper [18] suggested that there are 3 dimensions in the trust which are originator (trustor), purpose (e.g., trust in math, but not in writing), and target (trustee). For the purpose that they think different cases will have different purposes, trust should be evaluated based on these different purposes. For example, the paper [8] is using direct trust, recommendation, incentive function, and active degree on the nodes to measure the trust; and the paper [9] uses communication trust, data trust, and energy trust that in communication trust has direct trust and indirect trust; and the papers [10–13] all have different purposes and metrics, but they both use the direct trust and indirect trust metrics. There are two popular ways to combine the metrics in current trust algorithms which are using weight factors in [13] and fuzzy logic in [19]. The studies [11, 20] have tried something different and they select some of the powerful nodes as trust nodes to perform trust evaluation on the rest. The rest of the nodes are only collecting trust evidences rather than evaluating each other. All these studies are focusing on how to effective measure trust so as to avoid any malicious activity in the network, but none of them is discussing how the overlay trust evaluation and underlying network topology can affect and interplay with each other. The paper [21] proposed a trust-based routing protocol but more focused on the network topologies control. The nodes in the network are self-organized to form many 3-node triangle trust groups, so the network becomes of 2-level hierarchy in order to reduce traffic transmission overhead and delay. It is only focused on how the trust can affect the evolution of network topology, but it has not considered, on the other side, how the network topology change can affect the performance of the trust algorithm. As the density of nodes in the network increases, the triangle trust groups can become ineffective. In a dense network, a node with more than 3 neighbors that

the triangle structure can make the packet travel hops' count distance increase.

2.3. Trust in Cloud Computing. In [22], the trust has been defined as "a positive expectation or assumption on future outcomes that results from proven contextualized personal interaction-histories corresponding to conventional relationship types. It can be leveraged by formal and informal rules and conventions within a Social Cloud to facilitate as well as influence the scope of collaborative exchange." The study in [23] suggests that as users cannot see what is behind the cloud service providers, so building up a trust requires the 3rd party organization to audit and rate the cloud service providers. The research in [7] proposed a trust-based algorithm under a cloud-integrated Wireless Sensor Network which has three subsystems which include sensors, network, and cloud-based data servers. It suggested that, in the trust algorithm, the trust should increase slowly with good behaviors but decrease fast with dissatisfied behaviors so as to ensure detecting the malicious activities as fast as possible. They also suggested that the users have more trust in the cloud service provider who can provide more security mechanisms to their servers, access control, and transparent data process activities. These indicate that the trust is building on the knowledge of target and the approaches they have used to protect users' security.

Overall, the current research studies on distributed trust and reputation management are mainly focusing on how to effectively model and measure trust so as to best detect any malicious activity in the network, but there is no work discussing how the underlying topology change can affect the overlay trust behaviors. This steers our motivation to pave a new direction in the field of trust management in computer networks by exploring the novel concept of adaptive networks.

2.4. Adaptive Network. Adaptive network is a novel combination of two concepts which are dynamics *on* networks and dynamics *of* networks [24–26]. The dynamics *on* networks is defined as the status change on the network service running over the network such as network performance metrics of Quality of Service (QoS), but the dynamics *of* networks is defined as the change of the underlying network topological connectivity. There is actually an interplay between them which is termed the coevolution of networks. The study in [24] has explored how the Internet web state or behavior change can affect the underlying topologies, as well as how the underlying network connectivity change can impact the overlay Internet service state change. For example, if the users change their behaviors to conduct more online shopping this could cause more online shops (i.e., web servers) being established to be linked to the Internet and more network connectivity should be built. On the other hand, the website position in search results could affect user's preference to access content; this is how topologies affect state.

The studies in [27, 28] have discovered the rewiring issues to study how topology change can affect the state of nodes in the network. The research questions are normally as "where to add a link or node can best increase network performance?" as well as "where to remove a link in the network can result maximum reduction of network performance?" It has been found in [28] that adding a link between two weak connected nodes will result in most increase in network robustness and that removing a link between two strong connected vertices can have least reduction in network robustness. The studies in [29–31] have identified some effective metrics to measure network robustness. Here the network robustness is defined as the survivability of the network when it is under attack or major failure occurred. The effective metrics to measure network robustness are algebraic connectivity, betweenness, clustering coefficient, and effective resistance. Betweenness is to find out the centrality of the node in the network, and the clustering coefficient is to find out the percentage of clusters out of the maximum number of possible clusters that can be formed in the network. When calculating the clustering coefficient, it is to use a close triad network structure as a cluster which is called as Simmelian triad in the sociology. In sociology, the study in [32] has suggested that the Simmelian tie has positive effect on increasing the effectiveness of interdependent tasks, but the structural hole has negative effect on the effectiveness of interdependent tasks. The trust based routing in WSCN, in other words, a sensor node, is sending the collected information to the cloud center to process by routing through others nodes. This can be considered as an interdependent task (i.e., to form an end-to-end path) in WSCN. Therefore we can have the following hypotheses to be validated:

(1) Simmelian tie characterized network structure has positive effect on fostering trustworthiness among wireless sensor nodes and can ensure a more trustable network structure for end-to-end communications service.

(2) On the other hand, the structural hole characterized network structure has negative effect on fostering trustworthiness and it provides high opportunity for misbehavior by node itself or easily becomes the target to be attacked by malicious nodes and is in a risk as being less trustable network structure.

In the following section we will introduce the concepts on Simmelian tie and also structural hole and how they are related to the formulations of trustworthiness as well as their related measurement.

3. Simmelian Tie versus Structural Hole and Their Measurement

3.1. Simmelian Ties versus Structural Hole. Simmelian tie [33] is defined as triad tie or closed structure tie that ties embedded in the cliques as shown in Figure 1. In sociology, it is believed that the Simmelian tie is stronger than other regular strong ties between two actors as it discourages misbehavior by introducing a third party to become "shadow of the others" and "shadow of the future." This strongly fosters a normative environment against opportunist and engenders mutual trust, reciprocity norms, and shared identity. It facilitates the collaborative efforts by making the actors more willing to exchange information among themselves.

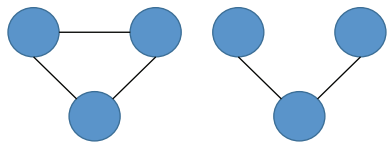

Figure 1: Simmelian ties versus structural hole.

On the other hand, the structural hole [34] is an actor connecting between two or more actors or parties who are not related or connected. It is opposite to the close structure of Simmelian tie, and it is an open structure tie as shown in Figure 1. In such case, this actor normally act as broker or gatekeeper which has positioning advantage to control the information flows among the networks. It plays a very critical role that once it is broken then the whole network is to be disconnected.

In terms of the trustworthiness, the actor is located in structure hole and is being positioned uniquely with an advantage to control the information flow among the network, which can provide the actor a fearless opportunity to act unethically toward all other parties without fear of the other person learning of his act. Being positioned in structure hole, the actor becomes a gatekeeper of information that might otherwise be transmitted between contacts. In addition to the opportunity of withholding critical information, the actor spans structural hole and may also have great opportunity to distort or terminate information flow that passes through each party. Moreover, from the network attack point of view, this actor is usually more attractive to be attacked because its bottleneck position. Taking the network routing as example, it means that there is no alternative route to be selected and cannot avoid the malicious attack if the actor is comprised or even itself has misbehavior.

The study in [35] suggested that the Simmelian tie can be good in some case like the interdependent task which requires teamwork and cooperation. It can also be negative as it enforces the group behavior that limits the innovation for the individual. On the other hand, the structural hole is the same. It is good in individual task as actor can more easily access the exclusive information from different parties to better deliver the task while it also presents opportunities for misconduct because when an individual spanning a gap between otherwise unrelated contacts, this individual is positioned to act unethically toward another individual or group without fear of the other person learning of the act.

In the following, we are going to identify some metrics for measuring the Simmelian tie and structural hole in the network.

For the measurement of Simmelian ties in a network, we first consider the undirected and unweight network as a graph $G(N, L)$, where N is the nodes set and L is the links set between nodes in G. As mentioned in Section 2, the clustering coefficient is reported as the most suitable metric for its measurement [29–31, 35]. In addition, the effective size is reported as a suitable metric to detect structural hole according to study in [35] as this metric can be used to detect nonredundancy connection in the network.

3.2. Clustering Coefficient. The clustering coefficient has been calculated by using the following equation in most studies [27–30]:

$$c_i = \frac{2}{n_i \times (n_i - 1)} \times s_i. \tag{1}$$

In (1), c_i is the clustering coefficient for node i in network G, n_i is the number of neighbors of node i, and s_i is the number of closed triples having been formed with node i. $n_i \times (n_i - 1)$ is number of possible triples that can be formed with node i. Then s_i can be defined as follows:

$$s_i = \frac{1}{2} \times \sum_{j=1}^{n_i} \sum_{k=1}^{n_i} l_{ij} l_{ik} l_{jk}, \quad j \neq k, \tag{2}$$

where j and k are the two different neighbors of node i. l_{ij}, l_{ik}, and l_{jk} are the connection links between nodes i and j, i and k, and j and k. If the connection between the nodes exists, then $l = 1$; otherwise $l = 0$. While node i has less than 2 neighbors, the clustering coefficient is 0. In such case, we can have the equation below:

$$c_i = \begin{cases} \dfrac{1}{n_i \times (n_i - 1)} \displaystyle\sum_{j=1}^{n_i} \sum_{k=1}^{n_i} l_{ij} l_{ik} l_{jk}, & n_i > 1, \\ 0, & n_i \leq 1. \end{cases} \tag{3}$$

For the whole network G, the average clustering coefficient can be used to measure the density of Simmelian ties. The equation is as below:

$$C_G = \frac{\sum_{i=1}^{n} c_i}{n}, \tag{4}$$

where C_G is the average clustering coefficient for network G and n is the number of nodes in the network G.

3.3. Effective Size. The clustering coefficient measures the redundant path in the network where the effective size is in the opposite way that it measures the nonredundancy connection in the network. In the study [35], the effective size is defined as

$$E_i = n_i - \frac{1}{n_i} \sum_{j=1}^{n_i} \sum_{k=1}^{n_i} l_{ij} l_{ik} l_{jk}, \tag{5}$$

where E_i is the effective size for node i. Equation (3) can be substituted into (5) and become

$$E_i = n_i - (n_i - 1) \times c_i. \tag{6}$$

As shown in Figure 2, we show how to calculate the clustering coefficient and effective size for these three different topologies. These network topologies have exactly the same number of nodes and links. Taking node 3, for example, in topology A, node 3 has 4 neighboring nodes, so $n_3 = 4$. In addition, there are 3 closed triples associated with node 3, so $s_3 = 3$, and $c_3 = (2/(4 * (4 - 1))) * 3 = 1/2$. Then we have $c_3 = (2/(3 * (3 - 1))) * 2 = 2/3$ for topology B and $c_3 = (2/(4 * (4 - 1))) * 3 = 1/2$ for in topology C, respectively. For the average clustering coefficient, $C_A = (1 + 2/3 + 1/2 +$

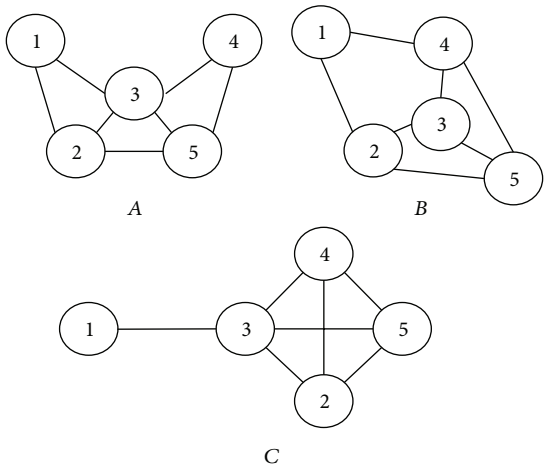

Figure 2: Three network topologies with the same node number and connection number.

$1 + 2/3)/5 = 3/4$, $C_B = (0 + 1/3 + 2/3 + 2/3 + 1/3)/5 = 2/5$, and $C_C = (0 + 1 + 1/2 + 1 + 1) = 2/3$. Regarding the values of average clustering coefficient, topology A has the highest value and it has better network topology structure too, as every node has Simmelian ties which can enforce the trustworthiness and also quickly detect the misbehaved node and find an alternative route.

In terms of effective sizes for node 3, they are $E_3 = 4 - (4-1) * 1/2 = 5/2$ in network A, $E_3 = 3 - (3-1) * 2/3 = 5/3$ in network B, and $E_3 = 4 - (4-1) * 1/2 = 5/2$ in network C. It can be seen that the effective size for node 3 in networks A and C is the same, but node 3 in network C is located as a structural hole position. Obviously the effective size does not effectively identify the structure hole where normally a node with higher effective size value is more likely at a structural hole position.

4. Trust-Based Geography Routing Algorithm

We have developed a trust-based geographical routing model associated with trust threshold mechanism to validate our hypothesis mentioned in the end of Section 2. This algorithm can find the destination by geographical information efficiently in large WSCN. It can filter out all the neighboring nodes with trust value below the trust threshold and select the best neighboring nodes as next hop from the remaining qualified neighbors. Figure 3 shows the decision flow chart for this algorithm.

It calculates trust value through direct trust and indirect trust (i.e., reputation). The direct trust is the trust evaluation performed by the trustor directly, while indirect trust is the direct trust value from other neighbors regarding the targeted node. These indirect values can be combined to reach a final trust value between 0 and 1 by using confidence factors. The direct trust metric equation is as shown below:

$$t_{\text{direct}} = \frac{s}{s+f}, \tag{7}$$

where s is the total number of good behaviors of target node and f is the total number of bad behaviors. In such case, whenever the neighbors have malicious behaviors, this routing algorithm can record them so as to decrease the trust value. The indirect trust is the trust value obtained from other neighbors who know the target node. The final trust value can be calculated using the following equation:

$$t_{\text{final}} = c_{\text{direct}} \times t_{\text{direct}} + \left(1 - c_{\text{direct}}\right) \times t_{\text{indirect}}. \tag{8}$$

There are also confident factors for direct trust and indirect trust which is c_{direct} in (8). The algorithm first sets up an initial trust threshold value and also the maximum trust threshold value $h_{\max}$; the equation is shown as below:

$$h_{\max} = \left(\frac{\sum_i^n t_i}{n}\right) - 0.1. \tag{9}$$

In (9), n is the number of selected nodes with good behavior, and t_i is the direct trust value. In such case, this threshold can make sure that all the nodes with good behavior are in the safe forwarding list. There is a chance that some of the good nodes have bad performance by accident, and the safe forwarding list size could be decreasing over time. Moreover, sometimes the node can be surrounded by malicious neighbors which can end up with an empty safe forwarding list. In such case, the algorithm will make sure that there are sufficient choices in the list and also give second chance to the nodes having poor performance previously to ensure the basic communications service. When the threshold is equal to $h_{\max}$ and safe forwarding list size is less than 30% of number of neighbors, threshold value will drop by 0.1 so as to give second chance to those nodes having bad performance before. If those nodes' performance becomes good again, their trust values should be increased back to the standard level. If after the first decrease in threshold value the safe forwarding list is still less than 30%, then nothing is to happen until the list is empty. When the list is empty, the algorithm will drop the threshold by 0.1 again until the list is not empty any more.

After the safe forwarding list is generated, the distance metric selects the neighbor with the shortest distance to the destination as next hop from the list. The distance equation is shown as below:

$$d = \sqrt{\left|x_n - x_d\right|^2 + \left|y_n - y_d\right|^2}, \tag{10}$$

where (x_n, y_n) and (x_d, y_d) are the longitude and latitude of neighbor and destination. The algorithm selects the neighbor from safe forwarding list with lowest distance d as next hop to forward the packets. But some nodes do not always have neighbors on the direction to the sink. In such case, the study in [36] has provided a solution called perimeter forwarding which selects the first neighbor on the counter clockwise the direction to sink, as shown in Figure 4.

In such case, the nodes switch between distance and perimeter forwarding mode to ensure that the next hop is properly determined. Moreover, no-cross heuristic was suggested in [35] as well to support perimeter forwarding so

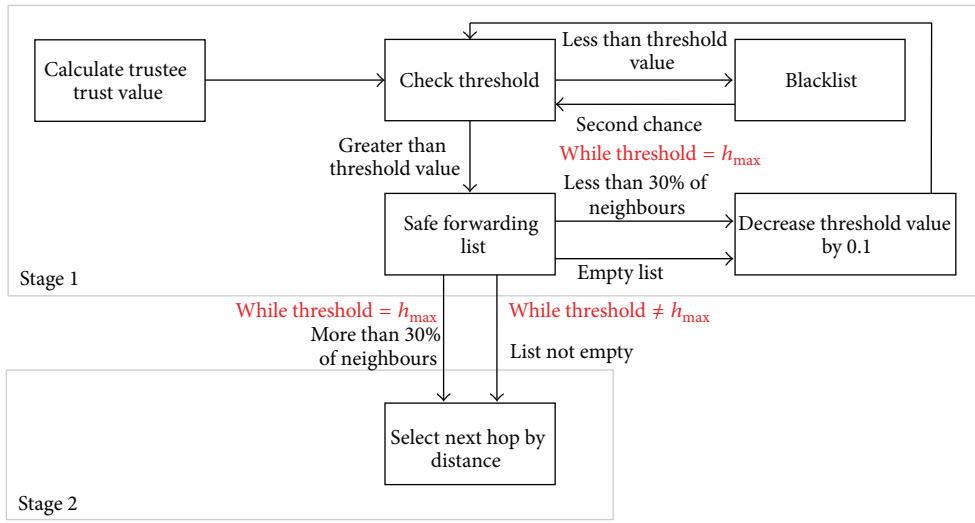

Figure 3: The decision flow of the proposed trust-based routing model.

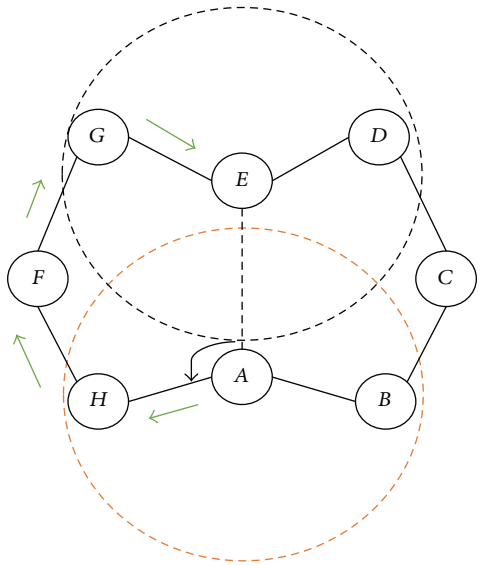

Figure 4: An example of perimeter forwarding.

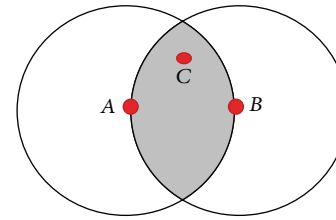

Figure 5: Relative Neighborhood Graph (RNG).

as to make sure that the packet is not travelling in loop by using the Relative Neighborhood Graph (RNG) approach as in Figure 5.

In RNG, if there is an edge existing between node A and node B, it should not have the third node C in the grey area. The circle indicates the wireless radio range for node A and B. This mechanism is to ensure that the algorithm can find a route to the destination if there is one.

5. Simulation Studies

The simulation studies have been conducted to validate our hypothesis on the roles of Simmelian ties and structural hole for fostering trustworthiness in sensor-cloud networks by using J-Sim tool [37]. We are using the clustering coefficient to differentiate the Simmelian ties amount in the network and effective size to detect the structural hole. The higher the clustering coefficient, the more Simmelian ties in the network. In the same way, the higher the effective size is, the more likely the node is structural hole.

5.1. First Case Study. First of all, we set up four network topologies which all have 16 nodes and 26 links as shown in Figures 6–9.

As shown in Figures 6–9, network 1 has nine Simmelian triangles and network 2 has seven Simmelian triangles. Then there are three Simmelian triangles in network 3, and network 4 has ten Simmelian triangles associated with two structural holes (in nodes 10 and 6). The average clustering coefficients for networks 1–4 are listed in Table 1. The value of clustering coefficient is reflecting the number of Simmelian triangles in network where network 4 has the highest and network 3 has the smallest coefficient.

We are using the trust-based routing algorithm introduced in Section 4 to validate our hypothesis. For each network, we basically set up three traffic demands from node 1 to node 16, node 2 to node 14, and node 3 to node 13. The packets are being sent one by one at a time. For example, node 1 sends a packet to node 16, then node 2 sends the second packet to node 14, and node 3 sends the third packet to node 15 until all packets are being sent. There are 300 packets to send for each traffic demand and thus 900 packets in total.

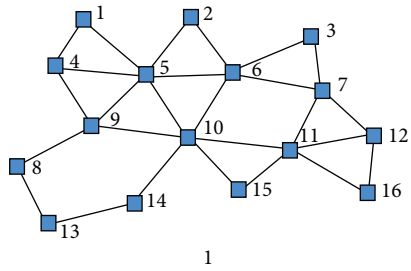

Figure 6: Network 1 with 9 Simmelian triangles.

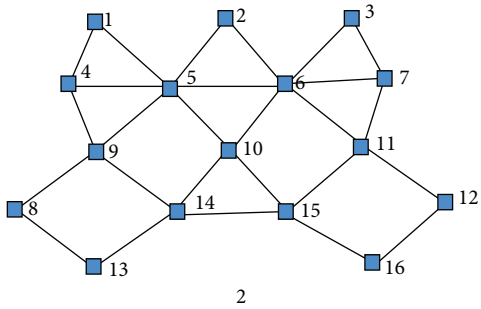

Figure 7: Network 2 with 7 Simmelian triangles.

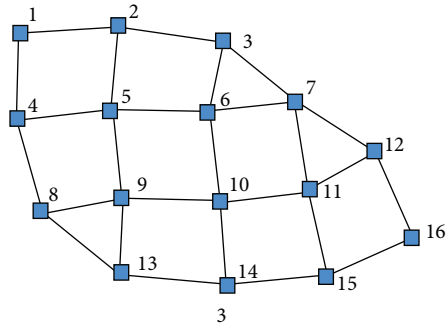

Figure 8: Network 3 with 3 Simmelian triangles.

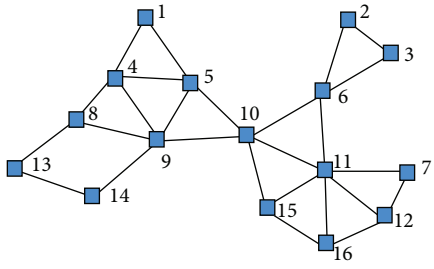

Figure 9: Network 4 with 10 Simmelian triangles and 2 structural holes.

The packet interval is set as 3 seconds with 32 bytes data and 128 ms time to live (TTL). h_{max} is set as 0.7.

There are six nodes that have been set up as source nodes and destination nodes, so there are ten nodes left to deploy the malicious attack in the case study. Here we assume that these nodes are nodes 4 to 12 and node 15.

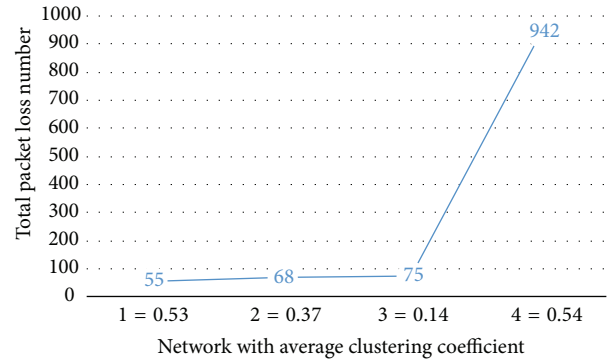

Figure 10: Packet loss versus average clustering coefficient.

Table 1: Average clustering coefficient.

Network	1	2	3	4
Average clustering coefficient	0.53	0.37	0.14	0.54

Table 2: Packet loss and latency.

Network	1	2	3	4
Total packet loss	55	68	75	942
Average packet latency (ms)	6.23	6.42	7.25	8.84

We have deployed grey-hole attack with 50% packet drop rate in a sensor-cloud environment. For each network we deploy the attack on those leftover ten nodes one at a time. This means that there are ten attack scenarios for each network.

As we can see in Figure 10 and Table 2, the same trust-based routing algorithm achieved the lowest total number of packets lost regarding the all ten attack scenarios. According to the average clustering coefficient number for each network in Figure 10 and Table 1, network 1 has the second highest coefficient out of four networks, followed by network 2, and network 3 is smallest. These are supporting our first assumption that the Simmelian tie has positive impact on fostering trustworthiness by achieving the lower packet loss under malicious attacks. Moreover, as the number of Simmelian ties increases, the average packet latency is decreasing which means a shorter distance to the destination node. As more Simmelian ties also means more alternative routes and the routing algorithm has more chance to find a short route to destination. Moreover, the reason for less packet loss in more Simmelian triangles network is that the Simmelian ties effectively enforce the reputation (i.e., indirect trust) share among the nodes within triangle which can identify the malicious behaviors earlier and quicker to avoid, so they can be avoided earlier to have less packet loss. For example, we deploy the grey-hole attack on node 10 in network 3; it takes 9 packets' loss to identify that node 10 is acting maliciously and start avoiding it, but in network 1 it only takes 5 packets' loss time to detect that node 10 is malicious. This is because the neighbor sooner has negative opinion on node 10 (low indirect trust value) and informs other nodes.

However in network 4 with the highest average clustering coefficient a huge amount of packet loss occurs in total among ten attack scenarios. Though it has the largest number of Simmelian ties, more significantly, it also has two structural holes in the network which is the main reason causing the huge packet loss. When we deploy the attacks on node 6 or node 10 which are the structural hole positions in the network, although the trust routing algorithm has detected the malicious behavior on these two nodes, an alternative route did not exist to avoid the attacks, so still a huge amount of packet loss occurs. This validates our second hypothesis that the structural hole characterized network has heavy negative impact on trustworthiness and degrades the effectiveness of Simmelian ties. Here it is worth highlighting that the node located in the structure hole usually has more attractiveness to be attacked from outsiders because of its significant impact on the network performance like in the above case. In addition to this situation, the structure hole can also provide the node itself with a fearless opportunity to act unethically toward all other nodes without fear of the other nodes' learning about its misbehavior as explained above about the high packet loss and being helpless to find an alternative path to avoid this malicious node. Being positioned in structure hole, the node becomes a gatekeeper of information that might otherwise be transmitted between contacts. The node spanning structural hole may have great opportunity to hold or distort the critical information, that is, grey-hole attack or even terminating all the information flows, that is, black-hole attack, that pass through each party.

Secondly, taking network 4 as an example, we have calculated the effective size for all nodes, and their values are shown as Table 3.

It can be seen that the two structural hole nodes are nodes 6 and 10, but the node with the highest value of effective size is node 11 rather than nodes 6 or 10. Node 10 is the second large one. It is interesting to confirm again that the metric of effective size is not very effective, as reported in other literatures such as [34], to identify all the nodes located in the structural hole. This raises a new direction to discover more finer and effective metric for future work.

In addition, we conclude that the structural hole characterized network also has significant negative impact on Simmelian tie characterized network. Although network 4 has the largest number of Simmelian ties, the existence of two structural hole has significantly weakened or even totally disabled the positive effect of Simmelian ties on forming the overall trustworthiness through the network, and we need to consider both characteristics of Simmelian ties and structure holes to evaluate the overall network trustworthiness.

5.2. Second Case Study. In the second case study, we set up five network scenarios with the same number of nodes (i.e., 16 nodes) and their locations in the network. Rather than having the same number of connection links like the previous case, we gradually increase the connectivity, by adding 4 links each time, to the first benchmark network *A* with 18 links in Figure 11. All the other derivational network topologies are shown in Figures 12–15. The rest of the simulation parameters set up is as same as the first case study. There are a total of 900 packets to send from node 1 to node 16, node 2 to node 14, and node 3 to node 13, of which 300 packets are for each traffic flow. The grey-hole attack is deployed on nodes 4 to 12 and node 15 in each network case with 50% packet drop rate.

Table 3: Effective size of nodes in network 4.

Nodes	4	5	6	7	8	9	10	11	12	15
Effective size	2.5	2.5	3	1	2.3	3.8	3.8	4.7	1.7	1.7

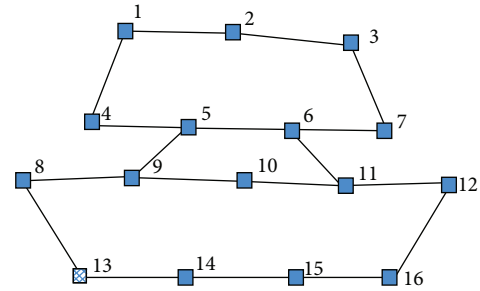

Figure 11: Network *A* with 18 links and no Simmelian triangle.

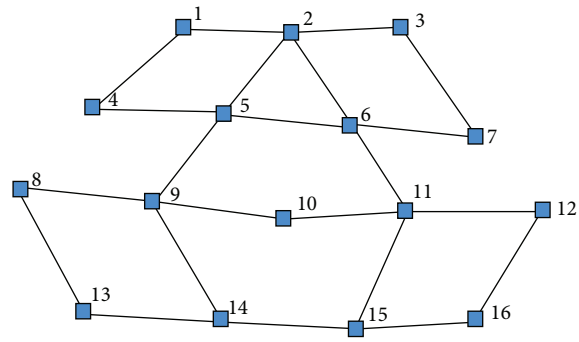

Figure 12: Network *B* with 22 links and 1 Simmelian triangle.

It can be seen that, from Figures 11–15 with the same number of nodes and their locations, the links are gradually being added and also the number of Simmelian increases from 0 to 19 where every node is embedded in at least two Simmelian triangles. Then the average clustering coefficient and network performance are summarized in Figure 16 and Table 4.

We can see that network *C* has the highest average clustering coefficient among the five networks, but the best network structure, in terms of overall trustworthiness and network performance, is network *E* with the similar lowest packet loss to network *D* and lowest average packet latency. Network *E* has the highest number of Simmelian triangles which are distributed evenly to cover the whole network to make it best structured. Although network *C* has much less links to connect all the nodes it still enables nodes 4, 7, 8, and 12 forming the most essential Simmelian triangles within these limited links. From this fluctuation of the clustering coefficient values, we can see that the average clustering coefficient is more accurate to measure the density difference of Simmelian triangles, for example, in first case study, regarding the various networks with similar numbers

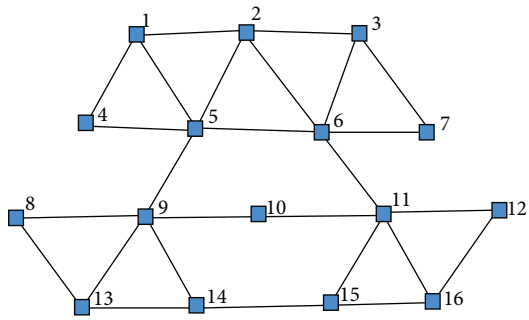

Figure 13: Network C with 26 links and 9 Simmelian triangles.

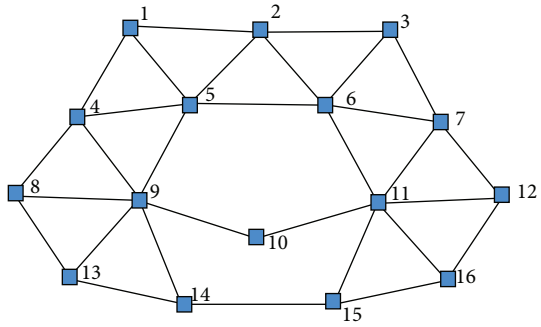

Figure 14: Network D with 30 links and 13 Simmelian triangles.

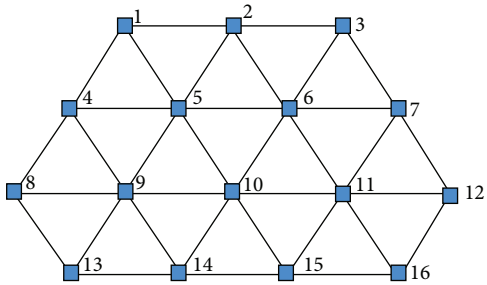

Figure 15: Network E with 34 links and 19 Simmelian triangles.

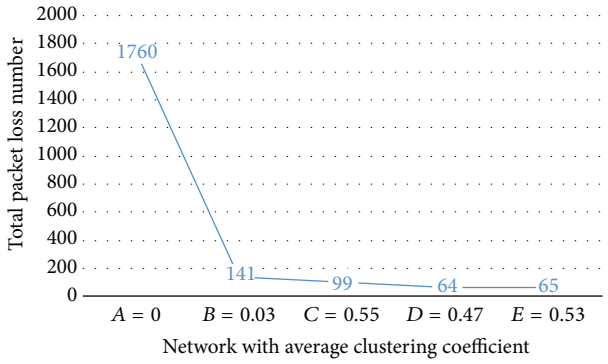

Figure 16: Packet loss versus average clustering coefficient.

Table 4: Packet loss and latency.

Network	A	B	C	D	E
Average clustering coefficient	0	0.03	0.55	0.47	0.53
Total packet loss	1760	141	99	64	65
Average packet latency (ms)	36.08	8.28	6.5	6.46	5.86

of nodes and links, but it is not that effective to measure the various networks with different number of links such as in this case.

Moreover, once again it can be seen that, as more and more Simmelian triangles increase in the network, the same trust-based routing algorithm can perform better and better to detect then avoid any malicious attack in terms of packet loss and packet detour latency. More Simmelian triangles embedded means more backup routes to be selected and more possible expressing warning messages, that is, indirect trust and reputation value collected from other neighbors so as to detect the malicious nodes. Network A with no Simmelian triangle significantly constrains the trust-based routing algorithm to find the trustable end-to-end path which causes a huge amount of packet loss and latency under malicious attack. Network A only has 18 links for 16 nodes that every node only has 2 connections on average. Once malicious attack occurs, most likely there would be only one route choice left to detour all the traffic flows. Another critical reason for the huge packet loss and latency is that all the detoured traffic flows will overload and congest the only leftover node or link to be routed which contributes more packet loss and latency worsening the case caused by the primary malicious attacks. In addition, the traffic flow needs to be detoured by using longer path to reach the destination and this contributes extra hop count to the high average packet latency. As the links increase, the number of alternative routes is increasing as well. The trust-based routing algorithm has more options to be selected, so it can work more effectively in terms of finding better trustable end-to-end routes. This is well confirmed by the results listed in Table 4; the packet loss decreases as the connection links increase, and the packet latency is decreasing as the shorter route is available to be used as well.

6. Conclusion

In this paper, we have studied how the underlying topology can impact the overlay trust and reputation evaluation behavior in sensor-cloud networks, specially how the Simmelian tie and structural hole play roles in establishing the trustworthiness among the nodes. We have piloted the average clustering coefficient metric to calculate the percentage of Simmelian triangle and then study the trust evaluation behavior by using a trust-based routing algorithm to implement end-to-end packet forwarding application in the sensor-cloud networks. The extensive studies have confirmed that the Simmelian tie characterized network structure has positive impact on frosting the trustworthiness, but the structural hole characterized network structure has high risk opportunities for misbehavior and malicious attacks. Moreover, Simmelian tie

and structural hole have an opposite structural feature in the network, and the structural hole can have negative impact on the advantage created by Simmelian tie characterized network as well.

In addition, more Simmelian triangles in the network can enable the same trust-based routing algorithm to perform more effectively. Furthermore, the average clustering coefficient can describe the finer structure difference regarding the networks with the same number of links and nodes.

The effective size metric cannot identify the structural hole in the network effectively in our simulation; we are discovering other better metrics or propose new metric so as to identify the structural hole in the future work. We have started looking at the betweenness metric in our second case study, and we found that the node in the network with higher betweenness value is more likely in a structure hole like position and can create more damage when it has misbehavior or is being attacked. We believe that the current work is thought-provoking, and there are more studies underway.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] L. Tan and N. Wang, "Future internet: the internet of things," in *Proceedings of the 3rd International Conference on Advanced Computer Theory and Engineering (ICACTE '10)*, pp. V5376–V5380, IEEE, Chengdu, China, August 2010.

[2] A. Iera, C. Floerkemeier, J. Mitsugi, and G. Morabito, "The Internet of things," *IEEE Wireless Communications*, vol. 17, no. 6, pp. 8–9, 2010.

[3] M. Momani and S. Challa, "Survey of trust models in different network domains," *International Journal of Ad Hoc, Sensor & Ubiquitous Computing*, vol. 1, no. 3, p. 1, 2010.

[4] P. Mell and T. Grance, "The NIST definition of cloud computing," July 2015, http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf.

[5] R. Kumar and S. Rajalakshmi, "Mobile sensor cloud computing: controlling and securing data processing over smart environment through mobile sensor cloud computing (MSCC)," in *Proceedings of the International Conference on Computer Sciences and Applications (CSA '13)*, pp. 687–694, December 2013.

[6] S. Saha, "Secure sensor data management model in a sensor—cloud integration environment," in *Proceedings of the Applications and Innovations in Mobile Computing (AIMoC '15)*, pp. 158–163, IEEE, Kolkata, India, Feburary 2015.

[7] O. Savas, G. Jin, and J. Deng, "Trust management in cloud-integrated Wireless Sensor Networks," in *Proceedings of the International Conference on Collaboration Technologies and Systems (CTS '13)*, pp. 334–341, May 2013.

[8] H. Xia, Z. Jia, L. Ju, X. Li, and Y. Zhu, "A subjective trust management model with multiple decision factors for MANET based on AHP and fuzzy logic rules," in *Proceedings of the IEEE/ACM International Conference on Green Computing and Communications (GreenCom '11)*, pp. 124–130, August 2011.

[9] H.-H. Dong, Y.-J. Guo, Z.-Q. Yu, and C. Hao, "A wireless sensor networks based on multi-angle trust of node," in *Proceedings of the International Forum on Information Technology and Applications (IFITA '09)*, vol. 1, pp. 28–31, May 2009.

[10] C. Tchepnda and M. Riguidel, "Distributed trust infrastructure and trust-security articulation: application to heterogeneous networks," in *Proceedings of the 20th International Conference on Advanced Information Networking and Applications (AINA '06)*, vol. 2, pp. 33–38, April 2006.

[11] F. Bao, I.-R. Chen, M. Chang, and J.-H. Cho, "Hierarchical trust management for wireless sensor networks and its applications to trust-based routing and intrusion detection," *IEEE Transactions on Network and Service Management*, vol. 9, no. 2, pp. 169–183, 2012.

[12] T. Liqin, L. Chuang, and J. Tieguo, "Quantitative analysis of trust evidence in Internet," in *Proceedings of the International Conference on Communication Technology (ICCT '06)*, pp. 1–5, IEEE, Guilin, China, November 2006.

[13] S. Peng, J. He, and Y. Meng, "Reputation-based trust update in network environment," in *Proceedings of the International Symposium on Electronic Commerce and Security*, pp. 118–123, August 2008.

[14] G. G. V. D. Bunt, M. A. J. V. Duijn, and T. A. B. Snijders, "Friendship networks through time: an actor-oriented dynamic statistical network model," *Computational & Mathematical Organization Theory*, vol. 5, no. 2, pp. 167–192, 1999.

[15] D. Barrera and G. G. van de Bunt, "Learning to trust: networks effects through time," *European Sociological Review*, vol. 25, no. 6, pp. 709–721, 2009.

[16] A. Jøsang, "The right type of trust for distributed systems," in *Proceedings of the Workshop on New Security Paradigms (NSPW '96)*, pp. 119–131, ACM, Lake Arrowhead, Calif, USA, September 1996.

[17] D. E. Denning, "A new paradigm for trusted systems," in *Proceedings of the Workshop on New Security Paradigms (NSPW '93)*, pp. 36–41, ACM, Little Compton, RI, USA, September 1993.

[18] B. Mu and S. Yuan, "A method for evaluating initial trust value of direct trust and recommender trust," in *Proceedings of the International Conference on Computer Design and Applications (ICCDA '10)*, pp. V2185–V2190, IEEE, Qinhuangdao, China, June 2010.

[19] P. N. Mahalle, P. A. Thakre, N. R. Prasad, and R. Prasad, "A fuzzy approach to trust based access control in internet of things," in *Proceedings of the 3rd International Conference on Wireless Communications, Vehicular Technology, Information Theory and Aerospace and Electronic Systems (VITAE '13)*, pp. 1–5, June 2013.

[20] Y. Zhang, L. Wang, and W. Sun, "Trust system design optimization in smart grid network infrastructure," *IEEE Transactions on Smart Grid*, vol. 4, no. 1, pp. 184–195, 2013.

[21] Z. Liu, L. Yu, W. Cheng, and K. Wang, "An independent trust routing framework based on trust topology control," in *Proceedings of the 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM '11)*, pp. 1–4, September 2011.

[22] S. Caton, C. Dukat, T. Grenz, C. Haas, M. Pfadenhauer, and C. Weinhardt, "Foundations of trust: contextualising trust in social clouds," in *Proceedings of the 2nd International Conference on Cloud and Green Computing (CGC '12)*, pp. 424–429, IEEE, Xiangtan, China, November 2012.

[23] D. W. Chadwick, S. F. Lievens, J. I. Den Hartog, A. Pashalidis, and J. Alhadeff, "My private cloud overview: a trust, privacy and

security infrastructure for the cloud," in *Proceedings of the IEEE 4th International Conference on Cloud Computing (CLOUD '11)*, pp. 752–753, July 2011.

[24] C. McCabe, R. A. Watson, J. Prichard, and W. Hall, "The web as an adaptive network: coevolution of web behavior and web structure," in *Proceedings of the 3rd International Web Science Conference*, pp. 22:1–22:7, New York, NY, USA, June 2011.

[25] T. Gross and H. Sayama, "Adaptive networks," in *Adaptive Networks*, pp. 1–8, Springer, Berlin, Germany, 2009.

[26] H. Sayama, I. Pestov, J. Schmidt et al., "Modeling complex systems with adaptive networks," *Computers & Mathematics with Applications*, vol. 65, no. 10, pp. 1645–1664, 2013.

[27] H. Wang and P. Van Mieghem, "Algebraic connectivity optimization via link addition," in *Proceedings of the 3rd International ICST Conference on Bio-Inspired Models of Network, Information and Computing Systems (BIONETICS '08)*, p. 22, November 2008.

[28] A. Sydney, C. Scoglio, and D. Gruenbacher, "Optimizing algebraic connectivity by edge rewiring," *Applied Mathematics and Computation*, vol. 219, no. 10, pp. 5465–5479, 2013.

[29] W. Ellens and R. E. Kooij, "Graph measuresand network robustness," http://arxiv.org/pdf/1311.5064v1.pdf.

[30] J. Martin-Hernandez and P. Van Mieghem, *TU Delft: Electrical Engineering, Mathematics and Computer Science: Intelligent Systems (INSY), and TU Delft*, Measuring Robustness of Complex Networks, Delft University of Technology, Delft, The Netherlands, 2013.

[31] H. Wang and P. Van Mieghem, "Robustness of networks," TU Delft: Electrical Engineering, Mathematics and Computer Science: Telecommunications, and TU Delft, Delft University of Technology, Delft, The Netherlands, 2009.

[32] S. L. Engle, *Structural holes and Simmelian ties: exploring social capital, task interdependence, and individual effectiveness [Ph.D. thesis]*, University of North Texas, Denton, Tex, USA, 1999.

[33] D. Krackhardt, "The ties that torture: simmelian tie analysis in organizations," *Research in the Sociology of Organizations*, vol. 16, no. 1, pp. 183–210, 1999.

[34] R. S. Burt, *Structural Holes: The Social Structure of Competition*, Harvard University Press, Cambridge, Mass, USA, 2009.

[35] V. Latora, V. Nicosia, and P. Panzarasa, "Social cohesion, structural holes, and a tale of two measures," *Journal of Statistical Physics*, vol. 151, no. 3-4, pp. 745–764, 2013.

[36] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the 6th Annual International Conference on Mobile Computing and Networking (MOBICOM '00)*, pp. 243–254, August 2000.

[37] J-Sim, https://sites.google.com/site/jsimofficial/.