Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2012, Article ID 145702, 9 pages
doi:10.1155/2012/145702

Research Article

MDS-Based Wormhole Detection Using Local Topology in Wireless Sensor Networks

Xiaopei Lu, Dezun Dong, and Xiangke Liao

College of Computer Science, National University of Defense Technology, Hunan 410073, China

Correspondence should be addressed to Xiaopei Lu, luxp02@gmail.com

Received 28 September 2012; Accepted 27 November 2012

Academic Editor: Shuai Li

Wormhole attack is a severe threat to wireless sensor networks (WSNs), which has received considerable attentions in the literature. However, most of the previous approaches either require special hardware devices or depend on rigorous assumptions on the network settings, which greatly limit their applicability. In this work, we attempt to relax the limitations in prior work, and propose a novel approach to detect wormhole attacks by only local topology information in WSNs. The basic idea is as follows. Each node locally collects its neighborhood information and reconstructs the neighborhood subgraph by multidimensional scaling (MDS). Potential wormhole nodes are detected by validating the legality of the reconstruction. Then, a refinement process is introduced to filter the suspect nodes and to remove false positives. Our approach solely relies on the local connectivity information and is extremely simple and lightweight, which makes it applicable in practical systems. Extensive simulations are conducted, and the results demonstrate the effectiveness and superior performance of our approach.

1. Introduction

Wormhole attack is a severe threat to wireless networks, which has attracted considerable attentions since it was introduced in previous works [1]. Recently, wormhole attack has become a more critical problem, especially in large-scale WSNs [2]. In a wormhole attack, the adversary places two radio transceivers, which are connected through high-speed channel. Each transceiver, captures signals in the network and delivers them to the other end. These signals are replayed, respectively, at the two ends. Then, two distant sensor nodes that are, respectively, around these two transceivers will consider each other as a close neighbor. By building these tunnels, wormhole attacker can fundamentally change the network connectivity, create a set of shortcut paths, attract a large amount of network traffic, and launch many kinds of attacks, such as selectively dropping or modifying packets and breaking the order of packets. Moreover, by attracting network traffic and collecting and analyzing network data, the attacker can perform many other more aggressive and severe attacks, such as denial of service attacks, network hijacking, and man-in-the-middle attacks. Since wormhole attacks are independent of the MAC layer protocol and immune to the cryptographic techniques, most of traditional security mechanisms are vulnerable to them.

To address wormhole attack in WSNs, a number of countermeasures have been proposed in the literature. Those solutions are respectively based on catching different symptoms of wormhole attack. However, most of them have various limitations, for example, requiring additional hardware devices, depending on special assumptions on the network settings. For instance, a number of methods are based on additional hardware devices, such as GPS [3], special radio frequency (RF) hardware [4], and directional antennas [5], which all significantly increase the hardware cost of the systems. Another kind of solutions depends on special assumptions on the network, such as global tight clock synchronization [6], special guarding nodes [7, 8], and attack-free initial networks [9, 10], which all greatly limit their applicability. In order to relax these limitations, a number of topology-based solutions are proposed [11–16]. These methods can detect wormholes by capturing various

symptoms on the network topology, by only exploring the network topology information. However, most of them still have various limitations, for example, centralized algorithms, requiring unit disk graph (UDG) model or relatively high node density, high false positive rate, and so forth. To sum up, wormhole attack has not been well addressed presently, especially in large-scale practical systems.

In this work, we propose a purely new topology-based wormhole detection approach in WSNs. We basically focus on exploring the abnormal structures introduced by wormhole attacks to the network topology. Each node v locally collects its k-hop neighborhood information and obtains the neighborhood subgraph. Then, we construct an estimation distance matrix that consists of the shortest distances (i.e., hop counts) of all node pairs in this subgraph. Next, the estimation distance matrix is used to reconstruct the subgraph and embed it on a plane by *multidimensional scaling (MDS)*, during which each node will be assigned a virtual position (i.e., node coordinates). The basic idea of our wormhole detection approach is based on an important observation as follows. If node v is a normal node, the layout of the MDS would well accord with the estimation distances, which means the distortion factor of the reconstruction would be relatively small. Otherwise, if node v is a wormhole node, its neighborhood subgraph cannot be smoothly embedded on a plane or at least would produce a great distortion factor. Based on this observation, we can detect potential wormhole nodes by validating whether the distortion factor of each node exceeds a threshold. Finally, we propose a simple but novel necessary condition for wormhole links and utilize it to filter the suspect nodes in a *refinement process*. Then, all wormhole nodes and wormhole links can be explicitly identified, with almost no false positives. Figure 1 briefly illustrates the detection results by our approach and the state-of-the-art methods. Black points in the gray regions denote real wormhole nodes, and circles denote detected wormhole nodes by wormhole detection algorithms. The given network graphs in Figures 1(a)–1(d), respectively, present the detection results by MDS-VOW method [12], LCT method [16], and our approach. We can see that MDS-VOW method can hardly work on this kind of wormhole attack, LCT method can detect all wormhole nodes, but with many false positives, and our approach can effectively detect all wormhole nodes with no false positives.

The main contributions of this work are as follows. Our approach does not require any additional hardware devices, but only needs each node to locally collect its k-hop neighborhood information. The algorithm is very simple and the overhead is extremely low, which makes it very applicable in practical WSNs. Moreover, not only can our approach identify all wormhole nodes and wormhole links, but also it produces very few false positives (almost no false positives according to extensive simulations).

The rest of this paper is organized as follows. We discuss related works in Section 2 and introduce the problem formulation in Section 3. Section 4 presents our detection approach in details. We evaluate this design through extensive simulations in Section 5 and conclude this work in Section 6.

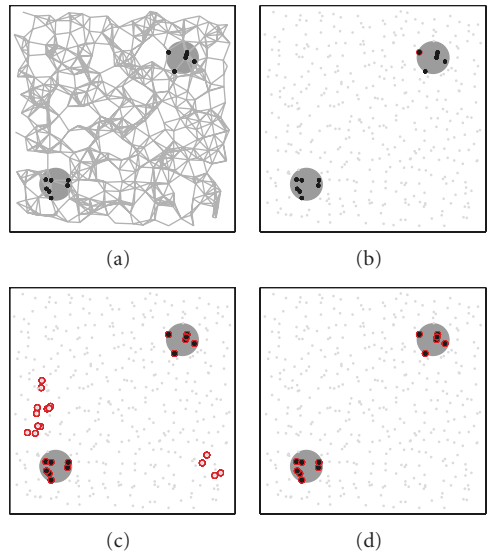


Figure 1: An illustration of wormhole detection results by different approaches. Gray areas denote the impact range of wormhole antennas. Black points in the gray areas denote real wormhole nodes that are directly affected by wormhole antennas. Circles denote detected wormhole nodes by respective detection approaches. (a) The original sensor network. 400 nodes are deployed over a square region. The average node degree is 7.5. Edges connecting wormhole nodes at different ends are omitted. (b) Detection results of MDS-VOW. Most of wormhole nodes are not detected. (c) Detection results of LCT. A number of false positives are produced. (d) Detection result of our approach.

2. Related Work

2.1. Wormhole Detection. A number of countermeasures have been proposed in the literatures. Existing methods are all based on capturing various symptoms induced by wormholes. In this section, we review and analyze the prior work.

The first line of existing solutions is based on the distance or timing analysis of data transmissions. Some methods attempt to detect wormhole attacks by validating the legality of packet traverse distance or time. By appending the location or time information of the sending nodes in each packet, they verify whether the hop-by-hop transmission is physically possible and accordingly detect the wormholes. However, such methods require the preknowledge of node locations by special hardware devices such as GPS [3, 6] or depend on the assumption of accurate globally synchronized clocks to capture the packet propagation time [4, 6]. These methods significantly increase the hardware cost of sensor nodes, and it is unclear whether these techniques would be effective in resource constraint WSNs.

Another line of existing solutions uses special communication devices. Some methods provide physical layer mechanism by using special radio frequency hardware to perform authentications in packet modulation and demodulation [6]. Hu and Evans [5] propose to adopt directional antennas to find and prevent infeasible communication links. The requirement of special hardware devices limits the applicability of these methods.

The third line of existing solutions is based on the discovery and maintenance of node neighborhood. For instance, LiteWorp [7] assumes that the network is attack-free before a time point, and each node collects its 2-hop neighbors. Then, LiteWorp selects a number of guard nodes to detect wormhole channels by overhearing the infeasible transmissions among those nonneighboring nodes. MobiWorp [8] is further proposed to complement LiteWorp by introducing some location-aware mobile nodes. Obviously, the assumption of attack-free environment significantly limits the applicability of these methods.

The forth line of existing solutions detects wormhole attacks by observing the symptom of traffic flow mismatch based on statistical analysis on the network traffic. For instance, Buttyan et al. [9] propose to detect wormhole attacks by capturing the abnormal increase of neighbor number and the decrease of the shortest path lengths that are induced by wormhole channels. This method is centralized because the base station needs to detect wormhole attacks by hypothesis testing based on the prestatistics of normal networks. Another statistical approach [10] is based on the observation that the wormhole links are selected for routing with abnormally high frequency. They identify wormhole links by comparing them with normal network statistics.However, these methods all require the prestatistics of normal network (i.e., attack-free environments).

The last line of existing solutions, which our approach would belong to, is based on the network topology. Wormhole attacks drastically change the network connectivity by introducing fake links among nodes near wormhole antennas, which will result in various abnormal symptoms to the network topology. Lazos et al. [11] present a graph-based framework to tackle wormhole attacks. They assume that a number of guard nodes that have extraordinary communication range exist in the network. The direct communication links between guard nodes and regular nodes would form special geometric structures, and the presence of wormholes would break these structures. Wang and Bhargava propose MDS-VOW [12] to reconstruct the whole network using MDS technique and detect wormhole links by capturing the abnormal features of the "network layout" introduced by wormholes. However, this method is centralized, and it can only work for special cases with only one infected node at both ends of the wormhole attack. In [13], the authors propose a wormhole detection approach with only local connectivity information. In networks with UDG model, their approach can accurately detect wormholes by looking for "forbidden substructures" that should not be present in a normal connectivity graph. However, it is inaccurate under non-UDG graph. Dong et al. [14] propose a distributed connectivity-based wormhole detection method. Each node collects its k-hop neighborhood and checks whether the boundary of its k-hop neighborhood subgraph has one or two circles. Its basic idea is based on the observation that the neighborhood that encloses a wormhole link will have two cycles and single cycle otherwise. However, Wormcircle requires relatively high node density to ensure that boundary detection algorithm works well. In another work [15], they propose to leverage global topological properties to detect wormhole attacks. They consider a legitimate multihop wireless network deployed on the surface of a geometric terrain as a 2-manifold surface of genus 0. Wormholes would introduce singularities or higher genus into the network topology. Ban et al. [16] propose local connectivity test (LCT) to identify wormhole attacks. Their basic idea is that removing the wormhole would disconnect its neighborhood from two components. Their algorithm works well in relatively dense and regular networks but results in many false positives in sparse or random networks.

To sum up, the wormhole attack problem has not been perfectly addressed presently. Existing solutions have various limitations, which make them lack applicability in practical WSNs. In this work, we attempt to propose a new wormhole detection approach to relax the limitations in prior work.

2.2. MDS and Its Applications. Multidimensional scaling was originally a method for visualizing dissimilarity data, which was developed from the behavioral and social sciences for studying the structure of objects. MDS takes a dissimilarity matrix among objects as input and produces a layout of the objects in a low-dimensional space as output. Its basic goal is to create a configuration of objects in a low-dimensional space (e.g., one, two, or three dimensions), and the distances between object pairs are close to the original dissimilarities.

Recently, MDS was applied in WSNs for solving the localization problem. As a fundamental problem in wireless networks, localization problem has been widely studied [17–22]. Shang et al. [17] propose a MDS-based localization algorithm that only relies on mere connectivity information and well tolerates measurement error. Ji and Zha [18] propose a distributed MDS-based sensor localization mechanism that presents a multivariate optimization-based iterative algorithm to calculate the positions of the sensors. In this work, we apply MDS to reconstruct the neighborhood subgraph of each node in WSNs. The input is the distance matrix of all node pairs, and the output is a set of virtual positions of all nodes. The virtual positions are used to calculate a virtual distance matrix of all node pairs. Then, the dissimilarity of these two distance matrices is utilized to evaluate the legality of the reconstruction.

3. Problem Formulation

3.1. Network Model. In our model, a WSN consists of a set of sensor nodes deployed over a plane region. Each node has a unique identity (ID). Nodes are only capable of communicating with other nodes in their proximity. We use G to denote the communication graph, where vertices and edges depict the nodes and communication links, respectively. We do not require the sensor nodes to be equipped with any special hardware, or achieve accurate globally synchronized clocks. Moreover, we do not place any restrictions on the network settings or topology, for example, static or dynamic nodes, node density, communication model, the uniformity of deployment, attack-free initial environment, and so forth. We set an assumption to the network as follows.

Each vertex v in the network G is capable of collecting its k-hop neighbor information. We use $N_G^k(v)$ to denote the neighbors of vertex v that are away from v within k hops in G. Let X be a vertex set in G, and let $G(X)$ be the vertex-induced subgraph by X, which consists of vertexes in X and edges among them. The k-hop neighborhood subgraph of vertex v is denoted by $\Gamma_G^k(v) = G(N_G^k(v) \cup v)$. This assumption is common in the literatures and is realistic in practical WSNs. It is worth noting that k would be a relatively small value, for example, $k = 2$ is sufficient for our algorithm, which makes our approach extremely lightweight.

3.2. Threat Model. In this work, wormhole attacks are defined based on the minimum capabilities required by the attacker to perform these attacks. In particular, the attacker does not need to compromise any node or have any knowledge of the network protocol used. Wormhole endpoints deployed by the adversary do not have valid network identities and do not become part of the network. We assume that in the network exist mechanisms that authenticate legitimate nodes and establish secure links between authenticated nodes. Although wormhole attacks impact neighboring discovery mechanisms in the physical or link layer greatly, transmitted data over encrypted network protocols remain transparent and unobservable to the wormhole attacker. These assumptions are common in prior work [3–6, 12, 13].

Then, we set an assumption on the threat model as follows. Each wormhole link e in network G is long enough to well separate nodes at the two ends of it. We denote nodes at the two ends of e by $V_1(e)$ and $V_2(e)$ and denote the shortest distance between $V_1(e)$ and $V_2(e)$ by $d_G(V_1(e), V_2(e))$. Then, we assume that $d_G(V_1(e), V_2(e)) > 2k$, where $2k$ presents the length of the wormhole attack, that is, the shortest distance between nodes at the two ends of the wormhole without wormhole links. The length of the wormhole determines the threat level of the wormhole attack. Longer wormholes are more dangerous because they have larger impact range and longer impact distance. For a short wormhole attack, its impact on the network connectivity would be negligible since only a small fraction of nodes are affected.

4. Local MDS-Based Wormhole Detection

In this section, we present the analysis and design details of our MDS-based wormhole detection approach.

4.1. Overview of Our Approach. Wormhole attacks introduce essential changes to the network topology. In order to detect wormhole attacks by only topology information, we have to capture the typical topological characteristics of wormhole links. The main idea of our detection approach is based on an observation as follows.

Each node v in the network G collects its k-hop neighborhood information, in particular, $k = 2$. The shortest distances (i.e., hop count) between all node pairs in the neighborhood subgraph $\Gamma_G^k(v)$ are used to construct an estimation distance matrix. Then, the distance matrix is used to reconstruct the subgraph by applying MDS on the subgraph and embedding it on a plane. There would be two conditions. First, if v is a normal node, the reconstructed subgraph would be relatively approximating to the original network. Thus, the embedded distance between each node pair would be relatively close to their estimation distance. Otherwise, if v is a wormhole node, its 2-hop neighborhood subgraph would contain all the wormhole nodes. Topologically, each wormhole node would connect with all nodes at the other end. Therefore, if we still constrainedly embed the subgraph on a plane, the distance constraints cannot be well maintained during the reconstruction. Based on this observation, we let all nodes in the network perform local MDS-based reconstruction and detect potential wormhole nodes according to the legality of their reconstructions. Additionally, we introduce a simple and effective necessary condition of wormholes to filter the suspect nodes detected by the previous process. Through this refinement process, we can remove most of false positives and identify all wormhole links.

As discussed previously, our detection approach mainly includes two components: (1) performing local MDS-based reconstruction and (2) performing refinement process. The first component obtains a number of suspect wormhole nodes. The second component filters the suspect nodes and presents the final detection results. We, respectively, describe these two components in detail as follows.

4.2. Local MDS-Based Reconstruction. For ease of representation, we divide this component into three subprocesses, as described hereinafter.

4.2.1. Distance Estimation. For an arbitrary node v in network G, it first collects its k-hop neighborhood information and obtains its k-hop neighborhood subgraph $\Gamma_G^k(v)$. Next, a classical shortest-path algorithm, for example, Dijkstra's shortest path algorithm, is applied to calculate the shortest distances between all node pairs in $\Gamma_G^k(v)$. Then, the shortest distance matrix $M[\Gamma_G^k(v)]$ is constructed, which is an $n \times n$ matrix (n denotes the number of nodes). Each element in $M[\Gamma_G^k(v)]$ is utilized as the estimation distance between each node pair.

4.2.2. Network Reconstruction. Using the shortest distance matrix $M[\Gamma_G^k(v)]$ as input parameter, we apply MDS to reconstruct the k-hop neighborhood subgraph of v. We denote the reconstructed network by $\overline{\Gamma}_G^k(v)$, in which each node would be assigned a virtual position (i.e., node coordinations). Then, the Euclidian distance between each node pair is calculated in $\overline{\Gamma}_G^k(v)$, and a virtual distance matrix $M[\overline{\Gamma}_G^k(v)]$ is produced.

4.2.3. Wormhole Judgement. Then, we describe how to decide whether a node is a wormhole node candidate by its reconstructed neighborhood subgraph. First, the distortion factor of the MDS reconstruction is calculated for each node v. The distortion factor is defined as follows.

Definition 1 (distortion factor). The distortion factor $\lambda(v)$ is defined as the root mean square error (RMSE) between the shortest distance matrix $M[\Gamma_G^k(v)]$ and the reconstructed virtual distance matrix $M[\overline{\Gamma}_G^k(v)]$, that is, $\lambda(v) = \sqrt{(1/(n \times n)) \sum_{i=1,j=1}^{n} (M[\overline{\Gamma}_G^k(v)](i,j) - M[\Gamma_G^k(v)](i,j))^2}$.

As discussed previously, each node produces large distortion factor if it is a wormhole node and little distortion factor otherwise. Based on this observation, we set a predefined threshold and label nodes that produce distortion factors above this threshold as suspect wormhole nodes. In our experiment, we set the threshold to be the median value of the distortion factors of all nodes in G, that is, $\lambda_{\text{threshold}} = (\lambda_{\max} + \lambda_{\min})/2$ and $\lambda_{\max} = \max\{\lambda(v) : v \in V(G)\}$, $\lambda_{\min} = \min\{\lambda(v) : v \in V(G)\}$, respectively.

Then, we present an efficient way to generate the threshold and distribute it to all nodes. Each node floods a message that contains its distortion factor and records the maximum and minimum values of all distortion factors in all flooding messages it receives. Each node only relays messages that contain a new maximum or minimum value. Thus, only two messages that, respectively, contain the globally maximum and minimum values would be flooded to the whole network. After the flooding is finished, each node calculates the threshold from the maximum and minimum values it records and compares it with its own distortion factor. If its distortion factor exceeds the threshold, it is labeled as a suspect wormhole node and normal node otherwise.

After the implement of this component, a number of suspect wormhole nodes are produced.

4.3. Performing Refinement Process. There is still an issue to be addressed. Some normal nodes may be wrongly labeled as suspect wormhole nodes, and false positives will be introduced. Too many false positives would result in normal links being removed and consequentially degrade the network capacity. In order to address this issue, we introduce this refinement process to filter the suspect nodes and remove false positives. By fully investigating the topology changes introduced by wormholes, we are able to capture some typical topological characteristics of wormhole links. Let X and Y denote two sets that, respectively, contain wormhole nodes at the two ends of a wormhole in network G; let $X \times Y$ denote the edge set between an arbitrary node pair $x \in X$ and $y \in Y$. Then, we present Theorem 2.

Theorem 2. *Given a network graph G and two wormhole node sets X and Y, the following two conditions hold.*

(1) *The subgraph G' that contains node set $X \cup Y$ and edge set $X \times Y$ is a complete bipartite subgraph of G.*

(2) *In the subgraph G'', which is constructed by removing all edges in $X \times Y$ from G, the k-hop neighbor sets of an arbitrary vertex pair $x \in X$ and $y \in Y$ have no common elements, that is, $N_{G''}^k(x) \cap N_{G''}^k(y) = \emptyset$.*

Proof. We first prove condition 1. Because X and Y, respectively, contain and only contain nodes at the two ends of a wormhole, each node v at one end is given the illusion that all nodes at the other end are its direct neighbors. Thus, there will be an edge between v and each node at the other end. According to the construction of G', it will obviously be a complete bipartite subgraph of G.

We then prove condition 2. If there are two nodes $x \in X$ and $y \in Y$ and $N_{G''}^k(x) \cap N_{G''}^k(y) \neq \emptyset$, the shortest distance between x and y must be less than $2k$, that is, $d(x,y) < 2k$. Consequentially, the shortest distance between node sets X and Y would be less than $2k$, that is, $d_{G''}(X,Y) < 2k$, which will contradict with our assumption in the threat model. □

Theorem 2 is a necessary condition of wormholes and is utilized to filter suspect wormhole nodes. First, all connected components are found in these suspect nodes. We denote the set of such connected components by $\mathcal{C}$. Isolated nodes can be certainly excluded. Next, all maximal complete bipartite subgraphs (MCBSs) are found in these connected components. In order to improve the detection rate, we expand each connected component by adding all 1-hop neighbors of the nodes in the component into this component. By doing this, all wormhole nodes can be included in the component if at least one wormhole node at both ends of the wormhole is suspect node. The algorithm in [23] that finds the maximal complete bipartite subgraphs in any graph is applied on each $C \in \mathcal{C}$. Let $\mathcal{B}$ be the set of maximal complete bipartite subgraphs generated by this algorithm, and let $B = (X, Y)$ be an element in $\mathcal{B}$, where X and Y are the two partitions of the bipartite graph. Then, condition 2 in Theorem 2 is applied on each $B \in \mathcal{B}$. If $N_{G''}^k(X) \cap N_{G''}^k(Y) = \emptyset$, all nodes in B will be labeled as final wormhole nodes. Otherwise, they are excluded. Till now, the final detection results are produced.

Moreover, our ultimate goal of detecting wormhole attacks is to neutralize them without breaking regular network functions. In particular, we want to eliminate the high volume of traffic passing through the wormhole links that create the wormhole effect with keeping the sensing and computational capabilities of the nodes. After detecting all wormhole nodes, this can be easily done by removing edges $X \times Y$ in each bipartite subgraph $B \in \mathcal{B}$.

4.4. Algorithm and Discussion. We present Algorithm 1 that describes our wormhole detection approach. Then, several parameters that may influence the performance of our algorithm are discussed as follows.

First, we discuss the influence of k. In our simulations, k is set to be small constant $k = 2$. The reasons are twofold. First, small k introduces low communication overhead of each node for collecting its k-hop neighborhood information. Second, if v is a wormhole node, its 2-hop neighbors would cover all wormhole nodes. Therefore, setting $k = 2$ is sufficient for capturing the abnormal embedding chrematistics induced by this wormhole. Actually, setting k to be a larger value is even adverse to the detection, because larger k induces larger subgraph, which will reduce the proportion of wormhole nodes in the subgraph and accordingly degrade the distinguishability of wormhole nodes.

Then, we discuss the influence of $\lambda_{\text{threshold}}$. The selection of the threshold dramatically impacts the detection accuracy

Input:
A network graph $G(V, E)$.
Output:
A set of complete bipartite graphs $\mathcal{B}$.
(1) **for** each $v \in V$ **do**
(2) Collect k-hop neighborhood subgraph $\Gamma_G^k(v)$.
(3) Calculate the shortest distance matrix $M[\Gamma_G^k(v)]$.
(4) Reconstruct the subgraph by MDS.
(5) Calculate the virtual distance matrix $M[\overline{\Gamma}_G^k(v)]$.
(6) Calculate the distortion factor $\lambda(v)$.
(7) Flood $\lambda(v)$ to the network.
(8) Calculate the threshold $\lambda_{\text{threshold}}$.
(9) **if** $\lambda_v > \lambda_{\text{threshold}}$ **then**
(10) Add v to the suspect node set S.
(11) **end if**
(12) **end for**
(13) Find all connected components $\mathcal{C}$ from S.
(14) **for** each $C \in \mathcal{C}$ **do**
(15) Find each MCBS B from C.
(16) Add B to the MCBS set $\mathcal{B}$.
(17) **end for**
(18) **for** each $B = \{X, Y\}$ in $\mathcal{B}$ **do**
(19) **if** $N_{G''}^k(X) \cap N_{G''}^k(Y) = \varnothing$ **then**
(20) Remove edges $X \times Y$.
(21) **else**
(22) Remove B from $\mathcal{B}$.
(23) **end if**
(24) **end for**

Algorithm 1: Our wormhole detection algorithm.

of our approach. In particular, lower threshold guarantees to catch all wormhole nodes, but causes more false positives, which will increase the workload of refinement process. Otherwise, a higher threshold induces fewer false positives but may produce false negatives. Comparatively, we are more concerned with detecting all wormhole nodes. Therefore, our approach will be on the aggressive side and select a relatively lower threshold. In our simulations, the threshold is set to be the median value of all distortion factors. Moreover, it is also a concerning issue, which makes the generation and distribution of the threshold easier.

5. Evaluation

In this section, we conduct extensive simulations to evaluate the effectiveness and performance of our design and compare it with the state-of-the-art methods.

5.1. Simulation Setup

5.1.1. Node Deployment. Two node deployment models are used: perturbed grid and random deployment. Perturbed grid model is adopted [24] to approximate manual deployments of nodes, in which all nodes are placed on an $m \times n$ grid and perturbed around their initial positions with a perturbed ratio p. Let each cell in the grid be a square with edge length d. Then, the node with coordinate (x, y) will be randomly placed in the region $[x - pd, x + pd] \times [y - pd, y + pd]$. In random deployment model, each node is assigned a coordinate randomly drawn from the network field.

5.1.2. Communication Model. Although our approach does not require specific communication models, both UDG and quasi-UDG models are adopted to build the networks. In the UDG model, there is a link between nodes u and v if and only if their distance is no larger than R, where R is the communication radius. In quasi-UDG model, nodes u and v have a link if their distance is no larger than ρR and have a link with probability q if their distance is within $[\rho R, R]$, where $0 < \rho < 1$.

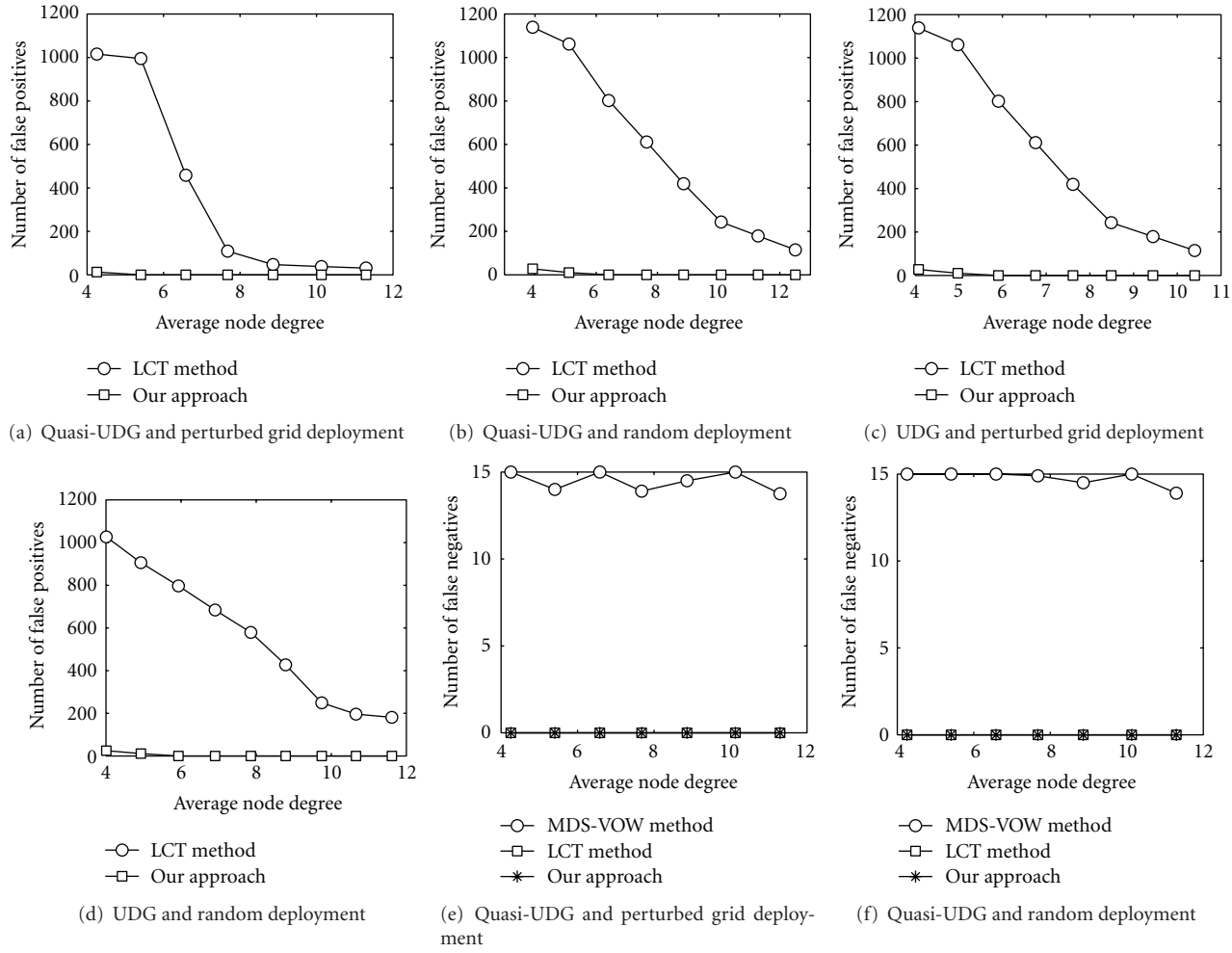
5.1.3. Wormhole Position. The wormhole position is a crucial factor for wormhole detection, because it could impact the significance of wormhole symptoms. Especially when multiple wormholes exist in the network, their relative position will dramatically influence the wormhole detection. In the simulations, our approach is evaluated for detecting wormholes placed at different positions of the network. Moreover, multiple wormholes with different relative positions are also evaluated.

5.2. Simulation Results. In this subsection, we present the results of the simulations under various network settings and compare them with the state-of-the-art MDS-VOW [12] and LCT [16] methods.

The basic network contains 1600 nodes deployed over a square region. In all simulations, $p = 2$ for perturbed grid model, and $\rho = 0.75$ for quasi-UDG model. The average node degree varies from 4 to 13. A set of wormhole nodes are placed at the diagonal of the network. The average number of wormhole nodes is 15. We require all algorithms to detect wormholes that are not shorter than 8 hops, that is, the shortest distance between nodes at the two ends of the wormhole is not less than 8. All simulations take 100 runs with random network generation and present the average results.

First, four sets of simulations are conducted to evaluate the number of false positives of our approach. Each set of simulations adopts different node deployments and communication models. The results are, respectively, presented in Figures 2(a)–2(d). From the results, we can obtain several observations as follows.

5.2.1. Influence of Node Density. The results in Figure 2 indicate that the number of false positives decreases for all approaches as the node degree increases. And our approach always greatly outperforms LCT method. However, when the degree is very low, there are still some false positives. The reason is analyzed as follows. In extremely sparse networks, there would be some special cases called bridge links, as shown in Figure 3. Although it is a normal link in the network, it topologically accords with the property of wormhole links. Some of these links may be wrongly labeled as wormhole candidates in MDS-based reconstruction and cannot be filtered by the refinement process.



(a) Quasi-UDG and perturbed grid deployment
(b) Quasi-UDG and random deployment
(c) UDG and perturbed grid deployment
(d) UDG and random deployment
(e) Quasi-UDG and perturbed grid deployment
(f) Quasi-UDG and random deployment

Figure 2: Simulation results. 1600 nodes are deployed over a square region. The average node degree varies from 4 to 13. In all simulations, $p = 2$ for perturbed grid model, and $\rho = 0.75$ for quasi-UDG model. A wormhole is launched at the diagonal of the network. The average number of wormhole nodes is 15. (a)–(d) evaluate the number of false positives under various network settings. (e)-(f) evaluate number of false negatives.

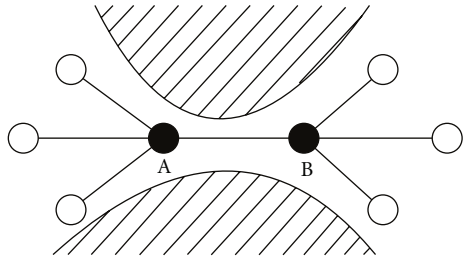


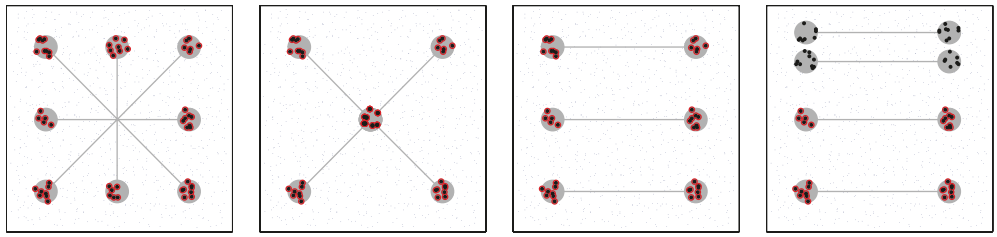
Figure 3: An example of bridge link. Link between nodes A and B may be aggressively labeled as wormhole link. The hatched areas denote holes of network deployment.

5.2.2. Influence of Node Deployment. It is shown in Figure 2 that our approach always produces few false positives for both perturbed grid distribution and random distribution. LCT produces fewer false positives for perturbed grid model than random deployment model. The reason is that perturbed grid model produces more regular networks.

5.2.3. Influence of Communication Model. Figure 2 demonstrates that our approach is not clearly influenced by the communication model. And it also demonstrates that our approach always induces much fewer false positives under both UDG and quasi-UDG models.

Then, we evaluate the number of false negatives of our approach, as shown in Figures 2(e) and 2(f). The results show that our approach can always detect all wormhole attacks. More results are constant under UDG model and are omitted here. The MDS-VOW method cannot even detect any wormholes because it does not work for the general wormhole model.

More simulations are conducted by placing wormholes at different positions in the network. The results are constant and are omitted due to the space limit. To sum up, our approach still works well in sparse and irregular networks



(a) (b) (c) (d)

Figure 4: Detection results for multiple wormholes. 900 nodes are deployed over a square region. Perturbed grid deployment with $p = 1.5$ and quasi-UDG with $\rho = 0.75$ are adopted to generate the networks. The average node degree is 7.5. Multiple wormholes are placed at different positions in the network.

and is not clearly influenced by the communication model. Moreover, our approach produces few false positives. It is worth noting that LCT method can obtain better results by increasing the shortest length of wormholes required to be detected. However, that will greatly restrict its applicability and increase the communication and computation cost.

5.3. Multiple Wormholes. In this subsection, our approach is evaluated for detecting multiple wormholes.

When the distance between two different wormholes is long enough, they will not affect each other. Thus, our approach can well detect all wormhole nodes, as shown in Figures 4(a)–4(c). Otherwise, if multiple wormholes are close, they may interfere with each other, which makes the detection more difficult. Particularly, if the distances of both ends of the wormholes are relatively short, as shown in Figure 4(d), our approach fails to detect the wormholes. The reason is as follows. When both ends of two wormholes are very close to each other, wormhole nodes at different ends are connected by short paths through wormhole links in the adjacent wormhole. Therefore, these nodes would be filtered during the refinement process. Actually, to the best of our knowledge, this situation cannot be solved by any purely topology-based detection methods.

6. Conclusions

As a severe threat to WSNs, wormhole attack has received considerable attentions during the past decade. However, most of existing countermeasures lack applicability for requiring special hardware devices or depending on rigorous assumptions on the network. In this work, we fundamentally analyze the essential wormhole symptoms by topological methodology and propose a local MDS-based wormhole detection approach. Our approach does not depend on any hardware requirements and is extremely simple and lightweight, which make it quite feasible in practical WSNs. Extensive simulations are conducted, and the results show that our approach can effectively identify all wormhole nodes for a large class of network instances.

Acknowledgments

The first author is supported by the National Natural Science Foundation of China (NSFC) under Grants no. 60903224 and no. 61202484. D. Dong is supported by NSFC under Grants no. 61272482 and no. 61170261.

References

[1] K. Sanzgiri, B. Dahill, B. Levine, and F. Belding-Royer, "A secure routing protocol for Ad Hoc networks," in *Proceedings of the IEEE International Conference on Network Protocols (IEEE ICNP '02)*, 2002.

[2] X. Mao, X. Miao, Y. He, X.-Y. Li, and Y. Liu, "CitySee: urban CO_2 monitoring with sensors," in *Proceedings of the 32nd IEEE International Conference on Computer Communications (IEEE INFOCOM '12)*, 2012.

[3] W. Wang, B. Bhargava, Y. Lu, and X. Wu, "Defending against wormhole attacks in mobile ad hoc networks," *Wireless Communications and Mobile Computing*, vol. 6, no. 4, pp. 483–503, 2006.

[4] S. Capkun, L. Buttyan, and J. P. Hubaux, "Sector: secure tracking of node encounters in multi-hop wireless networks," in *Proceedings of the ACM Workshop on Security in Ad Hoc and Sensor Networks (ACM SASN '03)*, 2003.

[5] L. Hu and D. Evans, "Using directional antennas to prevent wormhole attacks," in *Proceedings of the Network and Distributed System Security Symposium Conference (NDSS '04)*, 2004.

[6] Y. C. Hu, A. Perrig, and D. B. Johnson, "Packet leashes: a defense against wormhole attacks in wireless networks," in *Proceedings of the 22nd Annual Joint Conference on the IEEE Computer and Communications Societies (IEEE INFOCOM '03)*, pp. 1976–1986, April 2003.

[7] I. Khalil, S. Bagchi, and N. B. Shroff, "LITE WORP: a lightweight countermeasure for the wormhole attack in multihop wireless networks (DSN '05)," in *Proceedings of the International Conference on Dependable Systems and Networks*, pp. 612–621, July 2005.

[8] I. Khalil, S. Bagchi, and N. B. Shroff, "MOBIWORP: mitigation of the wormhole attack in mobile multihop wireless networks," in *Proceedings of the Securecomm and Workshops (SECURECOMM '06)*, September 2006.

[9] L. Buttyan, L. Dora, and I. Vajda, "Statistical wormhole detection in sensor networks," in *Proceedings of the Security and Privacy in Ad-hoc and Sensor Networks (IEEE ESAS '05)*, vol. 3813, pp. 128–141, 2005.

[10] N. Song, L. Qian, and X. Li, "Wormhole attacks detection in wireless ad hoc networks: a statistical analysis approach," in *Proceedings of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS '05)*, April 2005.

[11] L. Lazos, R. Poovendran, C. Meadows, P. Syverson, and L. W. Chang, "Preventing wormhole attacks on wireless ad hoc networks: a graph theoretic approach," in *Proceedings of the IEEE Wireless Communications and Networking Conference, Broadband Wirelss for the Masses—Ready for Take-off (WCNC '05)*, pp. 1193–1199, March 2005.

[12] W. Wang and B. Bhargava, "Visualization of wormholes in sensor networks," in *Proceedings of the ACM Workshop on Wireless Security (WiSe '04)*, pp. 51–60, October 2004.

[13] R. Maheshwari, J. Gao, and S. R. Das, "Detecting wormhole attacks in wireless networks using connectivity information," in *Proceedings of the 26th IEEE International Conference on Computer Communications (IEEE INFOCOM '07)*, pp. 107–115, May 2007.

[14] D. Dong, M. Li, Y. Liu, and X. Liao, "WormCircle: connectivity-based wormhole detection in wireless ad hoc and sensor networks," in *Proceedings of the 15th International Conference on Parallel and Distributed Systems (ICPADS '09)*, pp. 72–79, December 2009.

[15] D. Dong, M. Li, Y. Liu, X. Y. Li, and X. Liao, "Topological detection on wormholes in wireless ad hoc and sensor networks," in *Proceedings of the 17th IEEE International Conference on Network Protocols (ICNP '09)*, pp. 314–323, October 2009.

[16] X. Ban, R. Sarkar, and J. Gao, "Local connectivity tests to identify wormholes in wireless networks," Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing (ACM MobiHoc '11), 2011.

[17] Y. Shang, W. Ruml, Y. Zhang, and M. P. J. Fromherz, "Localization from mere connectivity," in *Proceedings of the PROCEEDINGS OF The Fourth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '03)*, pp. 201–212, June 2003.

[18] X. Ji and H. Zha, "Sensor positioning in wireless ad-hoc sensor networks using multidimensional scaling," in *Proceedings of the IEEE Computer and Communications Societies (IEEE INFOCOM '04)*, pp. 2652–2661, March 2004.

[19] S. Li and F. Qin, "A dynamic neuralnetwork approach for solving nonlinear inequalities defined on a graph and Its application to distributed, routing-free, range-free localization of WSNs," *Neurocomputing*. In press.

[20] S. Li, Y. Lou, and B. Liu, "Bluetooth aided mobile phone localization: a nonlinear neural circuit approach," *Transactions on Embedded Computing Systems*. In press.

[21] S. Li, B. Liu, B. Chen, and Y. Luo, "Neural network based mobile phone localization using bluetooth connectivity," *Neural Computing and Applications*. In press.

[22] S. Li, Z. Wang, and Y. Li, "Using laplacian eigenmap as heuristic information to solve nonlinear constraints defined on a graph and its application in distributed range-free localization of wireless sensor networks," *Neural Processing Letters*. In press.

[23] D. Eppstein, "Arboricity and bipartite subgraph listing algorithms," *Information Processing Letters*, vol. 51, no. 4, pp. 207–211, 1994.

[24] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Location-based compromise-tolerant security mechanisms for wireless sensor networks," *IEEE Journal on Selected Areas in Communications*, vol. 24, no. 2, pp. 247–260, 2006.