Computer Networks 141 (2018) 17–30

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Adaptive jammer localization in wireless networks

Tongxiang Wang[a], Xianglin Wei[b], Jianhua Fan[b,*], Tao Liang[b]

[a] Graduate School, PLA Army Engineering University, Nanjing 210007, China
[b] Nanjing Telecommunication Technology Research Institute, Nanjing 210007, China

ARTICLE INFO

Article history:
Received 23 June 2017
Revised 4 April 2018
Accepted 3 May 2018

Keywords:
Wireless network
Directional jammer
Localization
Gravitational search algorithm

ABSTRACT

The shared nature of wireless communication, the open access to wireless medium, the tightly coupled design principle of wireless network protocol and the scarcity of radio spectrum make wireless networks vulnerable to different types of jammers, which prevents the normal communication among legitimate nodes by occupying wireless channel or corrupting network protocols' working process and poses a serious threat to the stability and security of the network. To tackle this security threat, a few anti-jamming schemes have been put forward including channel hopping, spatial retreat, etc. Among these strategies, jammer localization has attracted much attention in recent years since it is very helpful for jamming-avoidance routing and even jammer elimination. However, existing algorithms mainly focus on the localization of jammers equipped with omnidirectional antennas and fail to handle directional jammers, i.e., those attackers with directional antennas. In order to bridge this gap, this paper puts forward an Adaptive Jammer Localization Algorithm (AJLA) which can locating both types of jammers. At first, an identification method is employed to estimate the jammer's antenna type. If a omnidirectional antenna is adopted, typical existing algorithm, such as Centroid Localization (CL), Virtual Force Iteration Localization (VFIL), can be utilized to locate the jammer. Otherwise, an Improved Gravitational Search Algorithm (IGSA) is developed for directional jammer localization. A series of simulation experiments have been conducted to evaluate the performance of the proposed method. Experimental results show that AJLA-IGSA could locate the directional jammer efficiently and accurately. Besides, the complexity of information collection and localization process are also analyzed by experiments.

© 2018 Elsevier B.V. All rights reserved.

1. Introduction

We are now living in a huge network interwoven by various wireless communication products, which bring many convenience to our daily life. However, the shared nature of wireless communication, the open access to wireless medium, the tightly coupled design principle of wireless network protocol and the scarcity of radio spectrum make wireless network vulnerable to different kinds of security threats, such as Denial of Service (DoS) attacks, worm attacks, etc. Jamming attack, as one type of Dos attacks, has been widely adopted by attackers for its easily deployment and severe damage to the network [1]. Jamming attack can prevent communications among legitimate nodes by occupying wireless communication channel or damaging the working process of network protocols [2,3]. For instance, a jammer can break down the CSMA/CA protocol through continuously sending fabricated Clear-to-Send (CTS) packets to the network. To restore normal network service, many anti-jamming strategies have been presented at many different protocol layers, among which jammer localization has attracted special attention since it provides us a chance to remove the jammer from the network based on its location.

Jammer localization has been widely investigated in recent years, and a number of algorithms have been proposed. Existing jammer localization algorithms can be divided into two categories, i.e. range-based ones and range-free ones. Typical methods include Centroid Localization (CL), Virtual Force Iteration Localization (VFIL) [4], Double Circle Localization (DCL) [5] etc. However, most of current algorithms only focus on the jammers equipped with omnidirectional antennas and cannot adapt to those scenario where directional antennas are adopted by the jammers. The propagation of the omnidirectional antenna is isotropy and its influenced area is approximately a circle [6]. In contrast, a jammer equipped with direction antenna could impact an irregular area. Furthermore, it is very hard, or if not impossible, for us to know the jammer's type in advance.

To tackle this problem, we need a jammer-type independent localization method which can locate both types of jammers without knowing their antenna models in advance. To fulfill this require-

* Corresponding author.
E-mail addresses: tongxiangw@yeah.net (T. Wang), wei_xianglin@163.com (X. Wei), fjh7659@163.com (J. Fan).

https://doi.org/10.1016/j.comnet.2018.05.002
1389-1286/© 2018 Elsevier B.V. All rights reserved.

ment, we face two unique challenges compared with omnidirectional jammer localization problem:

- Our method needs to firstly estimate the antenna type of the jammer since it has not pre-knowledge. In other words, it can distinguish the jamming scenarios caused by omnidirectional and directional jammers.
- We need to develop a new directional jammer localization method since no existing algorithm available to the best of our knowledge. This is more challenging compared with omnidirectional jammer localization since the influenced area of the jammer is irregular.

In this paper, an Adaptive Jammer Localization Algorithm (AJLA) is presented which can tackle the above two challenges simultaneously. AJLA firstly identifies the jamming scenarios to determine the jammer's type. Then, existing localization algorithms (e.g., CL and VFIL) are employed directly to locate the omnidirectional jammer. For the directional jammer, a novel jammer localization based on Improved Gravitational Search Algorithm (IGSA) is designed. The main contributions of this paper are three-fold:

- A jamming scenario identification method is put forward based on the jammed area's topology information. At first, an information collection protocol is designed to gather the Received Jamming Signal Strength (RJSS) values from boundary nodes. Then, with no prior knowledge of the jammer's antenna type, the boundary nodes' positions and RJSS values are utilized to reflect the influenced area's characteristics and identify the jammer's antenna type. With typical network settings, this method can accurately distinguish directional and omnidirectional antennas.
- A novel localization algorithm based on IGSA is proposed to locate the directional jammer since no available localization algorithm can accomplish this task to the best of our knowledge. Compared with current adopted Multilateral localization method, our IGSA-based localization method can decrease the mean localization error from 7.5 m to roughly 1.1 m under typical directional jamming scenarios. Furthermore, the proposed method can also be adapted to locate omnidirectional jammer, and it can achieve better accuracy than existing algorithms.
- A series of experiments have been conducted to evaluate the performance of our proposed algorithm. Simulation results show that AJLA can accurately and effectively locate both jammer types.

The organization of the rest of the paper is as follows. Existing jammer localization algorithms have been summarized in Section 2. Section 3 formulates the problem of jammer localization. AJLA is presented in Section 4 and its evaluation method and results are shown and analyzed in Section 5. The last section briefly summarizes the paper.

2. Related work

Over the past few years, jamming detection and jammer localization have been investigated a lot and several strategies have been proposed to detect or locate the jammer. Generally speaking, the process of mapping jammed area or locating the jammer is usually conducted after detecting the jamming attack.

2.1. Jamming attack and detection

The impact of jamming attack on the network was analyzed by Xu et al. firstly and four types of jammers were presented in reference [7], i.e., constant jammer, random jammer, reactive jammer and proactive jammer. As the rapid development of wireless network technologies, more intelligent ways could be utilized by adversaries to launch jamming attack. Zhang et al. put forward the jamming ACK attack to disrupt the traffic flow of IEEE 802.11 networks [8]. When the receiver sent back the ACK to the sender, the jammer would send out packets to collide with the ACK packets. Law et al. put forward several jamming ways, i.e. S-MAC, LMAC, B-MAC, to attack the data link layer of the network [9]. Little efforts would be expended to launch such energy-efficient jamming attacks. A synchronization-based distributed denial of service (DDoS) attack was put forward by Subir Biswas et al. to attack the vehicular networks [10]. The jitter and background period of road-side units (RSUs) were synchronized by the attacker in order to launch a successful attack. The jammer was assumed to be equipped with learning ability by Yang et al. against low-duty-cycle networks [11]. They formulated the problem of learning and attacking for the jammers as a joint optimization problem and an iterative method was proposed to solve it.

Correspondingly, a series of detection methods have been proposed to determine the existence of the jammer. For the physical jamming attack, Xu et al. put forward several strategies to detect the jammers [12], such as distance consistency or position consistency detection. Fragkiadakis et al. combined the simple threshold algorithm and Dempster–Shafer algorithm to detect physical layer jamming attacks [13]. This collaborative detection method increased the performance by more than eighty percent. As for the intelligent jammers, simple detection method based on Signal to Noise Ratio (SNR) would fail to cope with them. In such case, more information in physical, MAC or protocol layer should be collected before conducting the detection process. The correlation between three parameters, i.e. Packets Delivery Rate (PDR), signal strength, was utilized by Sufyan et al. to detect the jamming attack in IEEE 802.11b network [14]. Different types of jamming attacks were considered when detecting the jammer. Punal et al. put forward one detection method based on machine learning for IEEE 802.11 network [15]. A helper metric including PDR, max.IT, channel time, busy time, was employed as the input of the detection system. Typical machine learning methods, i.e. C4.5 Decision Tree, AdaBoost, Support Vector Machine (SVM) and Expectation Maximization (EM) were utilized to detect the jamming attack. Zou et al. aimed to detect the CTS jamming attack by comparing the destination address in the CTS frame with the information of neighbor nodes [16]. The address should belong to the two-hop neighborhood set, or the control packets can be viewed as fabricated. The neighborhood information needed to be updated by sending periodic HELLO packets. Soryal et al. took the DoS attack on MAC layer into consideration and the variety of network throughput was utilized to detect and identify the MAC-layer jamming attack [17], which can increase the probability of successful transmitted packet. Then, the maximum throughout was derived in terms of maximum number of packets for the nodes in order to decide the baseline for the detection threshold. Li et al. also took this kind of selfish node with smaller backoff time into consideration and put forward a real-time detection scheme [18]. Instead of comparing the backoff time between nodes, the observer calculated the joint probability of the events that one node chose shorter backoff time and then, a confidence level was proposed to qualify the confidence of detection results.

2.2. Omnidirectional jammer localization

Without depending on specialized devices, Anthony D. Wood et al. have proposed a Jammed-Area Mapping Service for sensor networks to detect and map jammed areas through protocol interaction [19]. The output of jamming detection module is a JAMMED or UNJAMMED message broadcast to the node's neighbors. The boundary nodes inside the jammed area broadcast the message to the mapped nodes outside the region. The mapped nodes collaborate to map the jamming reports, then reroute traffic around

the jammed area. However, the nodes need to broadcast the message, which may bring about "Broadcasting Storm". As for the jammer localization, it makes it possible to eliminate the jammer by physical ways when the positions of the jammer have been estimated accurately. The problem of jammer localization has been brought into focus by several researchers in recent years. Generally speaking, jammer localization algorithms can be divided into range-based ones and range-free ones.

The distance between the jammer and the nodes are utilized by the range-based algorithms to locate the jammer. Through simulations, Pelechrinis et al. found that the PDR of the nodes decreases when the distance between the jammer and nodes [20] increases. On the other words, the node with minimum RDR value locates closest to the jammer. Therefore, the PDR values of the node and its neighbors can be investigated to search for the jammer iteratively. Later, this method was improved by Wang et al. and they combined this method with adaptive power adjustment [21]. The node's power was adaptively adjusted when node's PDR value decreased to be zero. Simulation results showed that the localization accuracy was improved. The jamming attack could influence the hearing range of the node and the closer between the node and the jammer, the smaller the hearing range is [22]. The radius of node's hearing range is closely related to the jamming power and the relative distance between the jammer and the node. Then, the problem of jammer localization is translated to a least squares problem. Liu et al. formulated the problem of jammer localization to a non-linear optimization problem and the genetic algorithm was employed to solve this problem [23]. Wei et al. also made the research on the collaborative mobile jammer tracking and the jammer is located based on multilateral localization method [24]. In reference [25], Fan et al. has proposed the noise node elimination method, which can be used to improve the localization accuracy.

The information of jammed nodes and jammed area are exploited by range-free algorithms to locate the jammer. CL is the basic range-free algorithm and the centroid of the jammed nodes was considered as the estimated jammer's position. VFIL was proposed by Liu et al. to locate the jammer [4]. At first, the CL was employed to locate the jammer initially. Then, the jammed area formulated by the estimated jammer's position and the real jammed area were compared in order to adjust the jammer's position iteratively. The iteration would terminated until all the jammed nodes were covered by the estimated jammer. Sun et al. adopted the convex hull that formulated by the boundary nodes to estimate the jammer's position [26]. The center of the minimum circumscribed circle was considered as the jammer's position. α-hull was employed by Zhang et al. to minimize the circumscribed circle of the jammed nodes [27], and the least squares circle was adopted to search for the jammer's position iteratively.

2.3. Multiple jammers localization

For multi-jammers scenario, researchers have put forward several multi-jammers localization algorithms to locate the jammers. The topology formulated by the jammed nodes were employed to estimate the jammers' positions respectively [28]. At first, the jammed area was investigated to estimate the jammer's number. The, M-clusters and X-ray were adopted to estimate jammers' positions respectively [28]. Liu et al. also took the multi-jammers scenario into consideration and investigated different network topology formulated by the jammed nodes [29]. At first, the minimum spanning tree was utilized to identify and separate the jammed region and boundary region. For the sequential opening multi-jammers scenario, the mirroring algorithm and Least-square (LSQ) algorithm were adopted to locate the jammers. For the scenario where multi-jammers are turned on simultaneously, the Gauss–Newton Searching algorithm was adopted to search for the jammers' positions. k-mean cluster was employed by Wang et al. to locate the multi-jammers [30]. At first, the neighbor nodes' information was collected by the nodes whose power could adjust adaptively. Then, k-mean cluster algorithm was adopted based on the nodes' positions to estimate the number and positions of the jammers.

From the above analysis, it is obvious that existing localization algorithms are conducted under the assumption that the jammer is equipped with omnidirectional antenna. However, the directional jammer scenario invalidates this assumption and makes these localization algorithms fail. Several range-based algorithms (e.g. multilateral localization) may be utilized to locate the directional jammer, but their localization results are inaccurate. Moreover, no paper takes the identification of jammer's antenna type into consideration. Therefore, this paper aims to put forward an adaptive jammer localization algorithm that consists of two steps: identification and localization. Specially, a novel IGSA-based jammer localization algorithm is proposed to locate the directional jammer.

3. Problem formulation

To facilitate problem formulation, network, jamming and wireless channel models are introduced firstly. Then, the problem of jammer localization is formulated.

3.1. Models and assumptions

3.1.1. Network model

The characteristics of the wireless networks investigated in this paper mainly include:

- **Stationary**. The nodes are distributed randomly in the deployed area and communicate with each other through wireless channel in ad hoc mode. Once deployed, the nodes' positions remain unchanged in the localization process.
- **Neighbors awareness**. The nodes are equipped with omnidirectional antenna and their transmission powers are assumed to be constant. Therefore, the communication distance of all the nodes is similar and nodes within each other's transmission range will be neighbor nodes. Each node's routing table is maintained and updated regularly.
- **Location awareness**. The nodes' positions can be acquired by GPS devices or some kind of localization algorithms. This can be accomplished in the initial deployment process.

3.1.2. Jamming model

The jammer can have omnidirectional or direction antennas and this knowledge is not available in advance for the localization algorithm. The transmission power of the jammer is assumed to be stable in the localization process.

3.1.3. Wireless channel model

Typical wireless channel models include free-space propagation model, shadow-fading model, exponential-fading model etc. The shadow-fading model is used to model the small scale fading circumstance, which would be employed in this paper. If the receiver located at the distance d from the transmitter, the received power $PL(d)$ is:

$$PL(d) = PL(d_0) - 10\eta \log\left(\frac{d}{d_0}\right) + X_\sigma \tag{1}$$

where $PL(d_0)$ represents the received power at specific distance d_0 and η is the fading index. X_σ is the Gauss random variable with zero mean and variance σ^2.

3.2. Impact of jamming attack

The jamming attack can be launched from physical or MAC layer. The difference between jamming attacks initiated from physical and MAC layers is that the physical jamming attack would emit the signals independent of wireless nodes while the MAC-layer jamming attack needs to sense the state of the communication nodes before emitting jamming packets. The objective of both methods is to decrease the SNR of the receiver in order to make the legal packets be decoded incorrectly. In order to guarantee that the signal can be decoded correctly by the receiver, the received SNR should exceed a certain threshold. Besides, some more intelligent jammers would disguised as a legal node to send back the acknowledgement messages, such as the CTS jammer proposed in reference [16]. In this case, the jamming attack needs to be identified by exchange of MAC layer information.

According to the network protocols, each node determines its neighbors based on the received SNR and certain topology control protocols. Generally speaking, when all the neighbor nodes of an observer remain silent, the received noises mainly include background noise and jamming signal. However, the interference from other senders cannot be avoided. Thus, as for any pair of transmitter node i and receiver node j, the received SNR at node j is:

$$SNR_{ij} = \frac{P_{ij}}{P_N + P_J + \sum\limits_{k \in v(j)} P_{kj}} \tag{2}$$

where P_J is the received jamming power from the jammer. P_{ij} is the received signal's power from node i and P_N is the background noise. P_{kj} is the signal received by node j from neighbor k and $v(j)$ represents the neighbors set of node j except node i. Then, the state of communication link l_{ij} between node i and node j can be presented as:

$$l_{ij} = \begin{cases} 0, & SNR_{ij} \le \gamma_0 \\ 1, & SNR_{ij} > \gamma_0 \end{cases} \tag{3}$$

where γ_0 is the threshold of received SNR. $l_{ij} = 1$ represents that the normal communication between node i and node j is not affected by the jammer while $l_{ij} = 0$ represents the communication between them is disrupted.

According to the change of neighbor list, the nodes in the jammed wireless network can be divided into three categories:

- **Unaffected node**. A node is unaffected if it can receive packets from most of its neighbors after the appearance of the jammer while its received SNRs from most of its neighbors do not decrease remarkably.
- **Jammed node**. A node is considered to be a boundary node if it lost b percentage of neighbors or its received SNRs from some of its neighbors drop significantly.
- **Boundary node**. Besides unaffected and jammed nodes, other nodes in the network are boundary nodes, which usually could still communicate with a few neighbors with acceptable SNRs.

Note that our definition here is different from the one made in reference [31]. This is due to the following two reasons. Firstly, a node can judge its own status based on its own information in our judgement method while it has to contact nodes with unknown status to decide its own status in reference [31]. Secondly, in reference [31], to decide a jammed node's status, it has to try to communicate with all the unaffected nodes. This is hard to achieve or at least very time-consuming since it has to firstly determine which nodes are unaffected nodes.

3.3. Problem formulation

Under the above network settings, we aim at locating the jammer without any prior information of its antenna type. This process is usually conducted after jamming detection. We have summarized the related work on physical and MAC-layer jamming attack and corresponding detection methods. Many network metrics, including SNR, carrier-sensing time, collision probability, can be utilized to detect different kinds of jamming attacks. In this section, we mainly pay our attention to the localization process. To achieve this goal, we want to figure out the main challenges in this process and present our options and basic ideas here.

In order to locate the jammer, a locating node is selected by the unaffected nodes through some kind of voting algorithm which will be in charge of information collection and jammer localization. The locating node now has three main tasks to conduct, i.e. information collection, antenna type identification and jammer localization.

3.3.1. Information collection

The information we need to collect should fulfill two main criteria. To begin with, it can be easily measured by any wireless nodes including the simplest sensors in Wireless Sensor Network (WSN). Moreover, it can be used to detect the existence of jamming attacks. Based on these requirements, RJSS is the most suitable metric. Note that even packet or message collision among normal nodes is unavoidable, there are still some effective methods to distinguish these particular scenarios from jamming attacks [31]. Therefore, it is still possible to derive the RJSS in this circumstance.

As for physical jamming attacks, the jammers would emit useless signals to the wireless channel without considering the nodes' state. The nodes can collect the jamming signals without any extra equipment. However, this collection process is more complex for MAC-layer jamming attack. Generally speaking, the challenges of MAC-layer jamming attack mainly include:

- The jammer only attacks the wireless channel while sensing the normal communications among nodes. Therefore, the "quiet" mode can not be adopted to collect the jamming information.
- The jamming signal is mixed with the normal signals for MAC-layer jamming attack. Thus, an efficient method is needed to extract the RJSS from the mixed signals.
- Intelligent jammers may disguise as legitimate node and act with similar network protocols. The nodes need to distinguish the signal type, i.e. jamming signal or normal signal.

In order to estimate the RJSS of each node accurately, the method proposed in reference [31] is employed.

- As for the physical jamming attack, the sink node can send "quiet" command to legitimate nodes after detecting the jamming attack. In the "quiet" periods, the nodes can collect the RJSS.
- As for the MAC-layer jamming attack, each node will conduct the sampling process of ambient noise in the wireless channel. Considering the behavior of jamming attack and wireless nodes, the collected ambient noise can be divided into two categories: one is the noise consists of jamming signal and background noise and the other is the noise consists of jamming signal, background noise and signals from other senders. The variance of measurements is used to distinguish the two cases and the RJSS is estimated by the average of all measurements.
- As for the intelligent jammers, the collection process can be conducted with the detection process synchronously. Take the CTS jamming attack proposed in reference [16] as an example, the nodes can identify the jamming signal by exchange the information of CTS frame with neighbor nodes. If there is no corresponding address in the CTS frame, the signal will be considered as the jamming signal. At the same time, the recorded signal strength can be viewed as the RJSS.

One in particular is the MAC-layer jamming attack proposed in reference [10]. Though both of the jitter and backoff period have been synchronized, the jamming signal may exist in the wireless channel independently because the nodes will choose the backoff value in the backoff period randomly, which cannot be estimated accurately by jammers in advance. Thus, the duration of jamming signal is biased with that of legitimate signal.

In addition, the nodes that reporting these information to the locating node should be chosen delicately. According to the nodes' categories, it is easy to see that the unaffected nodes' RJSS values are too small (comparing to the background noise) and the jammed nodes that lost all their neighbor nodes cannot reports their information to the locating node. Thus, the boundary nodes whose RJSS values are high enough and could be able to communicate with unaffected nodes are the possible ones that satisfy our needs. Moreover, the locating node needs to design a collecting protocol to gather these RJSS values. To reduce the transmission overhead introduced by the localization process, the boundary nodes will send their RJSS values and positions to the locating node.

3.3.2. Identification of jammer's antenna type

After collecting RJSS values and positions of the boundary nodes, the locating node needs to estimate the jammer's antenna type before conducting localization algorithm. For a directional jammer equipped with an ideal directional antenna, its gain in the transmission angle is the same and zero otherwise. The gain of directional antenna can be expressed as:

$$G(\theta) = \begin{cases} C_1 & \theta \in [\theta_m - \theta_B \quad \theta_m + \theta_B] \\ 0 & otherwise \end{cases} \tag{4}$$

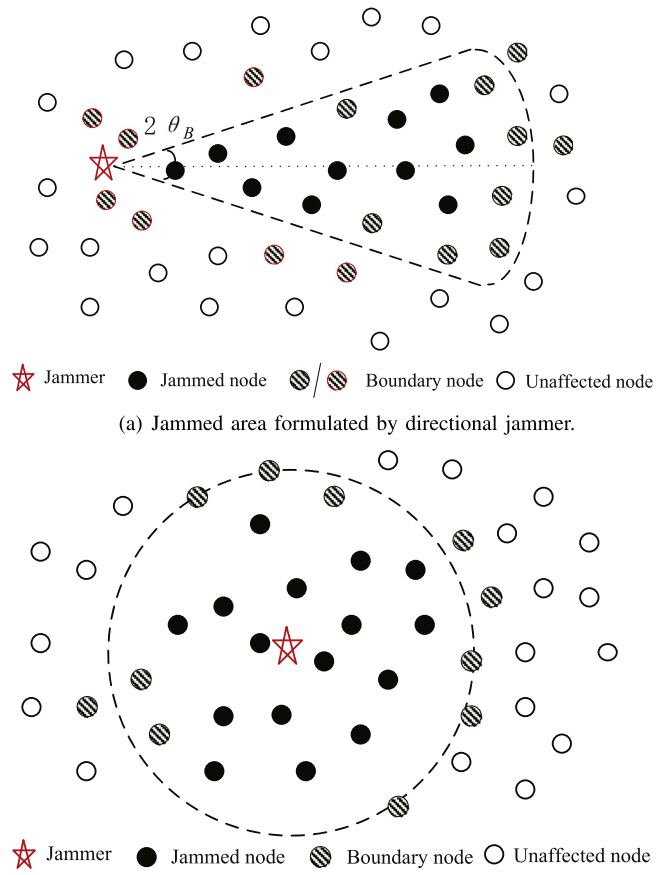
where $2\theta_B$ is transmission angle. θ_m is the beam width of the directional antenna.

Fig. 1(a) shows a typical scenario of a directional jammer's influenced area. The white circles represent the unaffected nodes, the shadow circles represent the boundary nodes and the black dots represent the jammed nodes. The jammer is denoted by the pentagram. It is clear that the jammed area is irregular and the jammer is not located at the center of the area. In contrast, the influenced area of the omnidirectional jammer is roughly a circle in Fig. 1(b). This means different localization strategies should be applied to these different jamming models. Therefore, our localization method should firstly decide the located jammer's antenna type.

To tackle this problem, we adopt a topology-based strategy. The basis of this strategy is that the jammed areas caused by different jamming models are different. To be specific, the jammed area formulated by omnidirectional jammer is approximately a circle while that formulated by directional jammer is irregular. Therefore, the collected information of boundary nodes can be utilized to reflect the topology of jammed area. If the jammed area is similar to a circle, then it can be determined the existence of an omnidirectional jammer. Section 4 will illustrate this strategy in detail.

3.3.3. Localization algorithms design

In order to locate different jammers, we define a jammer's Localization Nodes Set (LNS) as the set of nodes which are used to locate the jammer. Generally speaking, nodes in a jammer's LNS are usually selected from those nodes which are influenced by it, i.e., the jammed nodes or boundary nodes. For the omnidirectional jammer, a few existing localization algorithms are available to accomplish the localization task. However, there is no localization algorithm that aims at locating the directional jammer to this end. Some range-free localization algorithms can be adopted to locate the directional jammer, but the performance is poor.



(a) Jammed area formulated by directional jammer.

(b) Jammed area formulated by omnidirectional jammer.

Fig. 1. Jammed area formulated by different jammer.

To realize directional jammer localization, we employ IGSA in this paper. GSA is a heuristic algorithm that has been widely used in many fields and achieves good performance. However, GSA is sensitive to the distribution of initial particles and cannot be employed to locate the directional jammer directly. Based on the observation that the directional jammer locates at the edge of the jammed area, we need to refine the original GSA algorithm to increase the randomness of particles's positions in order to prevent the GSA from falling into local convergence.

4. Algorithm design

The adaptive localization process of jammer (omnidirectional jammer or directional jammer) can be divided into three steps: information collection, identification of jammer's antenna and design of localization algorithms.

4.1. Preliminary

An important preliminary for jammer localization is that affected nodes can detect the jamming attack. As a burgeoning research topic in wireless network security, it has attracted much attention in recent years. In the perspective of multiple network layers, a few jammer detection schemes have been put forward. For the elementary and intelligent jammers, there exist several detection schemes. As for physical layer, a series of detection models have been presented, such as signal strength measurement, carrier sensing time, PDR measurement, consistency checks [32]. Moreover, several proposals have been put forward to detect MAC layer selfish, DoS or misbehavior [33–35]. Recently there has been an in-

Node_ID	*Destination_node_ID*	*RJSS*	*Node_position*

Fig. 2. Frame structure of information collection.

creased interest in network layer intrusion detection to find those attacks target at route interaction and topology construction. However, jamming detection is out of the scope of this work as we focus on jammer localization and assume that wireless nodes can correctly detect the existence of the jammer.

After jamming detection, each node has to determine its state (be an unaffected node, boundary node or jammed node) as defined in Section 3.2. To achieve this goal, each node maintains its neighbor number and records each neighbor's SNR. Then, it will determine its own state according to the Algorithm 1 , where parameters *b, c* and *d* are determined by specific network condition.

Algorithm 1: Node state determination algorithm.

```
if a node does not detect jamming attack then
    This node is an unaffected node;
else
    if this node does not lose any neighbor and its received SNRs
    from most of its neighbors' do not decrease more than a
    percent then
        This node is an unaffected node;
    else
        if this node loses more than b percent of its neighbors
        then
            This node is a jammed node;
        else
            if more than c percent of its neighbors' SNRs decrease
            more than d percent then
                This node is a jammed node;
            else
                This node is a boundary node;
            end
        end
    end
end
```

4.2. Information collection

Based on the analysis in the above section, we have to design an information collection protocol to gather necessary RJSS values and other related information from nodes in the jammer's LNS while reducing the transmission overhead introduced by this process since normal communication among the network may be disrupted by the jammers. To achieve this goal, a node is required to report its RJSS values to the locating node if and only if it is judged to be a boundary node.

The boundary nodes will send the collected RJSS values and their own positions to the locating node. The basic structure of the reporting message is shown in Fig. 2, where *Node_ID* and *Destination_node_ID* represent the IDs of the boundary node and locating node respectively. *RJSS* and *Node_position* are the reported RJSS value and the coordinate of the boundary node respectively.

4.3. Identification of jammer's antenna type

Generally speaking, the jammed area of an omnidirectional jammer is approximately a circle while that of a directional jammer is irregular. Therefore, the locating node needs to judge the shape of the jammed area based on its collected RJSS values.

In omnidirectional jammer scenario where the distances between the jammer and boundary nodes are similar, boundary nodes tend to have similar RJSS values considering that the fading index of the wireless channel of all the nodes is assumed to be the same. On the other hand, this observation conclusion is not hold for the directional jammer scenario since the boundary nodes usually have different distances to the jammer due to the irregular of the jammed area. Therefore, the variance of the boundary nodes' RJSS values can be utilized to derive the jammer's antenna type.

Assume the RJSS value of boundary node i is R_i and the total number of boundary nodes is S. Then, the variance σ can be calculated by:

$$\sigma = \sum_{i=1}^{S} \left(R_i - \overline{R_i}\right)^2 \tag{5}$$

$$\overline{R_i} = \frac{1}{S} \sum_{i=1}^{S} R_i \tag{6}$$

where $\overline{R_i}$ is the average RJSS value of the all the boundary nodes. If σ exceeds a certain threshold η, the jammer is judged to be a directional jammer. Otherwise, the jammer is an omnidirectional jammer.

4.4. Localization algorithm design

If the jammer is determined to be an omnidirectional jammer, many existing algorithms can be utilized to accomplish the localization task, like CL, WCL, VFIL, etc. For the directional jammer, we design an IGSA-based jammer localization algorithm here.

GSA is a heuristic algorithm proposed by Esmat in 2009 [36]. In the GSA, several particles are deployed in the specific area randomly at the initial step and each particle represents a solution of the problem. The particle's mass is calculated by the fitness function, which is related to the particle's position. Under the action of universal gravitational forces among particles, the acceleration, velocities and positions of particles are updated after each iteration. In the moving process, the particles will move toward the particle with larger mass, whose value of fitness function is higher. At last, the position of particle with largest mass will be considered as the best solution. At present, GSA has been widely used in data mining, parameters identification, multi-objective decision, etc. At the problem of jammer localization, the objective is to estimate the jammer's position, which may be mapped to the particles' positions. Therefore, it is feasible to adopt the GSA to solve the problem of jammer localization.

The process of proposed IGSA can be divided into four steps: *1*) select the initial particles; *2*) determine the fitness function; *3*) calculate the resultant forces; *4*) update the acceleration, velocities and positions of particles. Furthermore, we have improved the randomness of the particles (in subsection to improve the accuracy of localization.

4.4.1. Select the initial particles

Generally speaking, IGSA is sensitive to the distribution of initial particles and the initial particle should cover the entire solution space as much as possible. Therefore, the initial particles' positions should be chosen evenly from the jammer's possible positions. In order to improve the uniformity of the initial particles' distribution, a backward learning method is adopted to promote the performance of the algorithm.

N particles are deployed in a D-dimensional space, the position of particle i is represented as $X_i = \left\{x_i^1, x_i^2, \ldots, x_i^k, \ldots, x_i^D\right\}$, where

$x_i^k \in [a_k, b_k]$, a_k and b_k are the lower bound and upper bound in the kth dimension. Then, the opposite particle of x_i^k is $\tilde{x}_i^k = a_k + b_k - x_i^k$. At last, the initial particles and opposite particles are all added to the final particle set.

Here, the particles are deployed in a 2-D space and a_k and b_k are determined by the area determined by the boundary nodes. Initially, the acceleration and velocity of the particles are set to be zero at the first iteration.

4.4.2. Determine the fitness function

The fitness values of particles are defined to evaluate the closeness of each particles's position to the jammer's real location. If a particle's position is close to the jammer's location, the boundary nodes will receive similar RJSS values when the jammer initiates attack in the particle's position. Given some particular transmission power of the jammer and a particle's position, we can calculate the RJSS value at each boundary node's location according to Formula (1). Then, this value can be compared with the boundary node's reported RJSS value. The difference between estimated RJSS value and reported RJSS value can be employed to reflect the particle's estimation accuracy. The closer a particle's position to the jammer's location, the smaller the difference between the estimated RJSS value and the reported value at each boundary node in the jammer's LNS. Note that we adopt the jamming power estimation method illustrated in reference [31] and will discuss it in Section 4.5. Based on this principle, we can calculate each particle's fitness value as follows.

To be specific, the fitness value $fit_i(t)$ of particle i at the t iteration can be defined as:

$$fit_i(t) = \frac{1}{M}\sum_{j=1}^{M}\left(\Delta R_j(t) - \Delta\overline{R_j}(t)\right)^2 \tag{7}$$

$$\Delta R_j(t) = R_j(t) - \widehat{R}_j(t) \tag{8}$$

$$\Delta\overline{R_j}(t) = \frac{1}{M}\sum_{j=1}^{M}\left(\Delta R_j(t)\right) \tag{9}$$

where M is the number of nodes in the jammer's LNS. $R_j(t)$ and $\widehat{R}_j(t)$ are the reported and estimated RJSS value of node j at tth iteration. $\Delta R_j(t)$ and $\Delta\overline{R_j}(t)$ are the error and average error of the RJSS values respectively.

4.4.3. Calculate the resultant forces

The Gravitational forces among particles are calculated based on the their mass. In IGSA, the mass of each particle is determined by its fitness function:

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \tag{10}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \tag{11}$$

where $m_i(t)$ and $M_i(t)$ are the mass and normalized mass for particle i at the tth iteration. $fit_i(t)$ represents the fitness value for particle i at the tth iteration. $best(t)$ and $worst(t)$ are the best value and worst value for all the particles at the tth iteration.

The gravitational force between particle i and particle j in the dth dimension is:

$$F_{ij}^d = G(t)\frac{M_i(t) \times M_j(t)}{d_{ij}^2(t) + \varepsilon}\left(x_j^d(t) - x_i^d(t)\right) \tag{12}$$

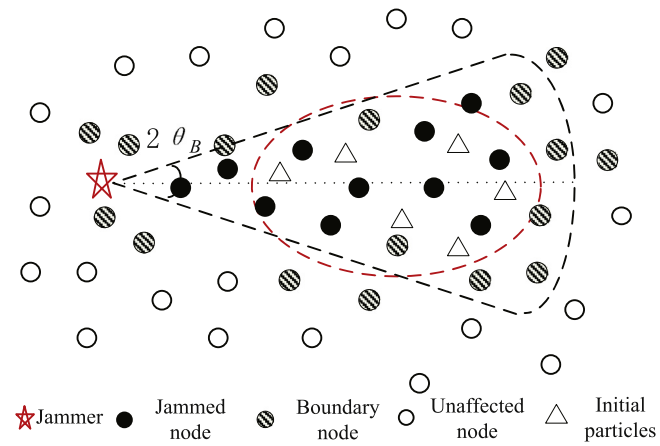


Fig. 3. Distribution of initial particles.

where $x_i^d(t)$ and $x_j^d(t)$ represent the position in dth dimension of particle i and j at the tth iteration respectively. $d_{ij}(t)$ is the Euclidean distance between these two particles. ε is a small constant and $G(t)$ is the gravity coefficient. The resultant force $F_i^d(t)$ of particle i in dth dimension is:

$$F_i^d(t) = \sum_{j=1, j\neq i}^{N} rand_j F_{ij}^d(t) \tag{13}$$

where N is the number of particles and $rand_j$ is a random number in [0,1].

4.4.4. Update the acceleration, velocities and positions of particles

The gravity coefficient $G(t)$ varies with the time and the calculation of $G(t)$ is:

$$G(t) = G_0 \times e^{-\gamma\frac{t}{T}} \tag{14}$$

where γ is the time constant, T is the maximum number of iterations and G_0 is the initial value of gravity coefficient.

The acceleration of particle i in dth dimension at the tth iteration is:

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} \tag{15}$$

where $a_i^d(t)$ is the acceleration of particle i in dth dimension at tth iteration. The velocity and position of particle i can be updated by:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \tag{16}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \tag{17}$$

where $v_i^d(t)$ and $x_i^d(t)$ are the velocity and position of particle i in the dth dimension at the tth iteration.

The IGSA will terminate until the iterations reaches to the maximum number. Otherwise, the mass, velocity and position for each particle will be updated at each iteration.

4.4.5. Improved strategies

As shown in Fig. 3, the deployment of initial particles may be concentrated at the center of the jammed area. In this case, the real position of the jammer is not covered by the particles and the localization results will be biased according to the searching result of IGSA. In order to overcome this weak point, the strategy increasing the randomness of the particles is designed in this subsection. To be specific, the particles with smaller mass (0.1*N) will choose new positions at each iteration. In other words, 0.1*N particles will be redistributed in the jammed area at each iteration.

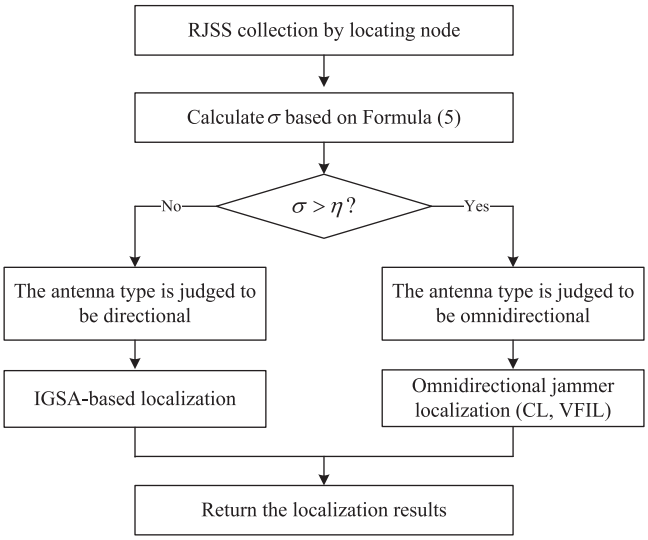


Fig. 4. Adaptive jammer localization.

In addition, we add the memorability and interconnection of Particle Swarm Optimization (PSO) algorithm to improve the performance of IGSA. In the IGSA, the velocities of particles update at each iteration according to the new formulation, which is represented as:

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) + rand_j \times \left(p_{best}^d - x_i^d(t)\right) + rand_k \times \left(g_{best}^d - x_i^d(t)\right) \quad (18)$$

where $rand_i$, $rand_j$ and $rand_k$ are the random variable in [0,1]. p_{best}^d and g_{best}^d are the best position of particle i and best position of all the particles through all the iterations respectively.

The flow chart of our proposed AJLA is illustrated in Fig. 4. Firstly, the RJSS values of boundary nodes are reported to the locating node. Then, the locating node identifies the jammer's antenna type based on the collected RJSS values according to the proposed identification strategy. If the antenna type is determined to be omnidirectional, traditional localization algorithms (eg. CL, VFIL) will be adopted to locate the jammer. Otherwise, the antenna type is determined to be directional and the designed IGSA-based method will be adopted to obtain the jammer's position.

4.5. Discussions

4.5.1. Jamming power estimation

In the IGSA-based localization algorithm, the fading index of wireless channel can be estimated by physical method during the transmission among unaffected nodes. However, the jamming power cannot be derived directly and thus the estimated RJSS values cannot be calculated in Section 4.4.2. In order to solve this problem, the jamming power is assumed to be chosen from a series of discrete values as assumed in reference [31] since simple jammers only have limited choices in their transmission powers. IGSA-based localization algorithm will be conducted under these discrete jamming powers and the best position with optimal fitness value under each jamming power will be recorded. At last, the global optimal value chosen from all the optimal values of different power levels will be determined and its corresponding position will be treated as the jammer's estimated position.

4.5.2. Determination of η

η is the threshold that determines the jammer's antenna type, which is related to the nodes' density, jamming power and background noise. For the omnidirectional jammer, σ is mainly determined by the background noise. As for the directional jammer, it is closely related to the jamming power, nodes' density and is hard to estimated accurately. Here, η is decided according to the background noise and assumed to be five times the mean noise power.

4.5.3. Determination of LNS

In our proposed algorithm, the boundary nodes are the potential localization nodes, but not all the boundary nodes can be added to the jammer's LNS. The boundary nodes influenced by the directional jammer is illustrated in Fig. 1(a). In the figure, the boundary nodes can be divided into two categories. The first category is the boundary nodes that enclosed by red circles. These nodes are determined to be the boundary nodes due to the lose of some neighbor nodes, but their RJSS values are similar to the background noise. In other words, these nodes are not affected by the jamming attack and cannot reflect the impact of the jammer. Thus, they cannot be added to the jammer's LNS. Therefore, when determining the jammer's LNS, we take the RJSS values of boundary nodes into consideration. A boundary nodes will be added to the jammer's LNS if and only if its RJSS value exceeds a certain threshold β.

4.5.4. Applying IGSA to omnidirectional jammer localization

The proposed IGSA-based directional jammer localization can also be utilized to locate the omnidirectional jammer. The time complexity of the proposed IGSA-based directional jammer localization is $O(TN^2)$. T is the number of iterations, and N is the number of particles. However, though the IGSA-based algorithm can also achieve good performance in locating omnidirectional jammer, there is a tradeoff between the localization accuracy and time complexity. Thus, we need to select proper algorithm when locating the omnidirectional jammer.

4.6. Pseudo-code of adaptive jammer localization algorithm

The pseudo code of adaptive jammer localization algorithm is illustrated in Algorithm 2 . At first, the RJSS values of the boundary nodes are collected by the locating node in Step 1. σ is calculated based on Formula (5) in Step 2, and if it exceeds η, the jammer is judged to be an omnidirectional jammer and typical omnidirectional localization algorithms is employed to conduct the localization task in Step 4. Otherwise, the jammer is a directional jammer and IGSA-based localization algorithm is employed to locate the jammer in Step 6 - Step 16. N particles are deployed according to the backward learning method in the jammed area in Step 8. The fitness values of all the particles are calculated in Step 9, and the best value need to be updated in each iteration in Step 10. Step 11 calculates the mass and normalized mass of the particles, and Step 12 calculates the resultant force of particles according to Newton's second law. The gravity coefficient and acceleration, velocity, position of the particle will be updated in Step 13. $0.1*N$ particles with lower mass are redeployed in the jammed area in Step 14. The best value of this power level and its corresponding particle's position are stored in Step 16. At last, the global best value chosen from all the best values of different power levels will be determined, and the corresponding particle's position is considered as the jammer's estimated position in Step 18.

5. Simulation and results

In this section, a series of experiments are conducted to evaluate the performance of AJLA.

Algorithm 2: Adaptive jammer localization algorithm.

Input: number of iterations (T), number of particles(N), number of power levels (p), acceleration, velocityand position of the particle, gravity coefficient and time constant α

Output: the estimated jammer's position

```
Collect the RJSS values and positions of boundary nodes based on the collecting protocol;
Calculate σ according to Formula (5);
if σ exceeds η then
    Typical omnidirectional jammer localization algorithm is adopted to locate the jammer;
else
    for k = 1 : p do
        for i = 1 : T do
            Select N particles according to backward learning method in the jammed area;
            Calculate the fitness function for each particle;
            Save the best value at this iteration and update the best value;
            Calculate the mass and normalized mass for each particle;
            Calculate the resultant force in each dimension for all the particles;
            Update the gravity coefficient and acceleration, velocity, position of the particle;
            0.1*N particles with lower mass are redeployed in the jammed area;¡¡
        end
        Store the best value and the corresponding particle's position;
    end
    The position of particle with global best value among all the power levels is the estimated jammer's position.
end
```

5.1. Experimental settings

5.1.1. Experimental environment

M nodes are deployed in a L^*L square area randomly. The jammer, which is equipped with an omnidirectional or directional antenna, is located at (400,400). Other related parameters are shown in Table 1.

5.1.2. Comparison benchmark and evaluation metrics

The Euclidean distance between estimated jammer's position and the jammer's real position is employed to evaluate the accuracy of different localization algorithms. The mean error E_m is defined as:

$$E_m = \frac{1}{Q}\sum_{i=1}^{M}\left(\left\|X_i - \widehat{X}_i\right\|\right) \tag{19}$$

where Q is the simulation times and X_i and $\widehat{X}_i$ are the ith estimated position and real position of the jammer respectively.

In our proposed AJLA, CL, VFIL are adopted to locate the omnidirectional jammer while IGSA (defined as AJLA-IGSA) is employed to locate the directional jammer. In the section of simulations results, we mainly employs some typical methods to evaluate the performance of IGSA for directional jammer. The typical methods include Multilateral Localization (ML) and GSA-based localization (AJLA-GSA). Multilateral localization is a typical range-based method that can be utilized to locate the directional jammer. In AJLA-GSA, we take no modification of GSA into consideration when locating the directional jammer.

Table 1
Experimental Parameters .

Parameter	Meaning	Value
M	Number of nodes	800
L	Range of the network	800 m
N_J	Number of jammers	1
α	Threshold of received SNR	5 dB
β	Threshold of RJSS value	10^{-5}
Q	Simulation times	200
P_J	Transmission power of jammer	14 mW
P_T	Transmission power of node	10 mW
P_N	Power of background noise	−60 dBm
n	Fading index of channel	2
G_r	Gain of transmitting antenna	1
G_t	Gain of receiving antenna	1
η	Threshold for the determination of jammer's antenna	5×10^{-6}
$2\theta_B$	Transmission angle	60
G_0	Initial value of gravity coefficient	100
T	Number of iteration	100
N	Number of particles	20
γ	Time constant	20
b	Parameter 1 of nodes classification	0.85
c	Parameter 2 of nodes classification	0.9
d	Parameter 3 of nodes classification	0.95

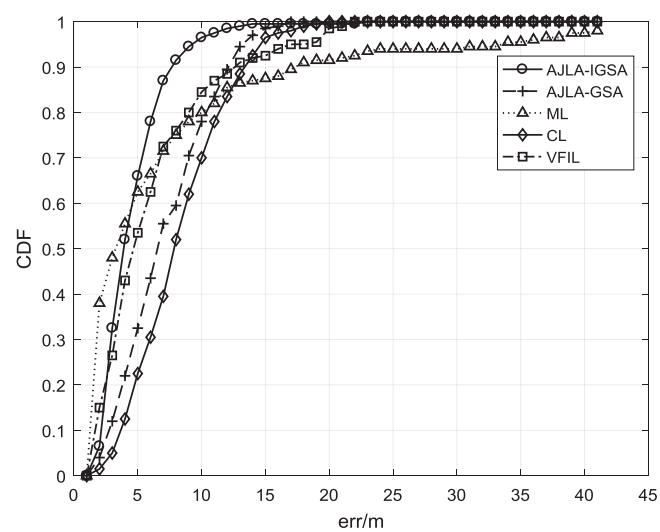


Fig. 5. CDF of localization errors of different algorithms for omnidirectional jammer.

5.2. Experimental results

5.2.1. Comparison of localization errors for omnidirectional jammer

The AJLA-IGSA, AJLA-GSA, CL, VFIL and ML are employed to locate the omnidirectional jammer respectively. For each method, we have conducted the simulations for 200 times independently. The Cumulative Distribution Function (CDF) of localization errors and mean errors are illustrated in Figs. 5 and 6 respectively. From the figures, it can be concluded that the proposed AJLA-IGSA and AJLA-GSA can be utilized to locate the omnidirectional jammer and achieve good performance. The mean localization errors of AJLA-IGSA, AJLA-GSA, ML, CL and VFIL are 3.6 m, 6.1 m, 7.8 m, 7.2 m and 5.1 m respectively. The localization accuracy of them are similar, but the time complexity of AJLA-IGSA or AJLA-GSA is higher than other three algorithms. Thus, when locating omnidirectional jammer, we prefer to select existing algorithms (eg. CL, VFIL) in order to decrease the time complexity while ensuring localization accuracy.

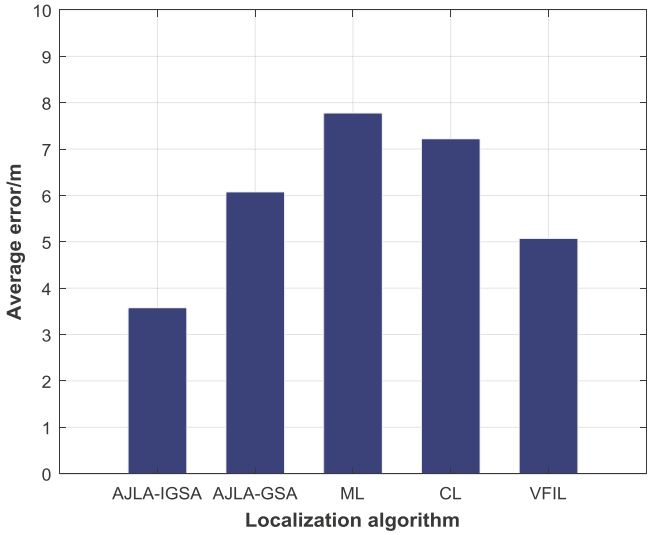


Fig. 6. Mean localization errors of different algorithms for omnidirectional jammer.

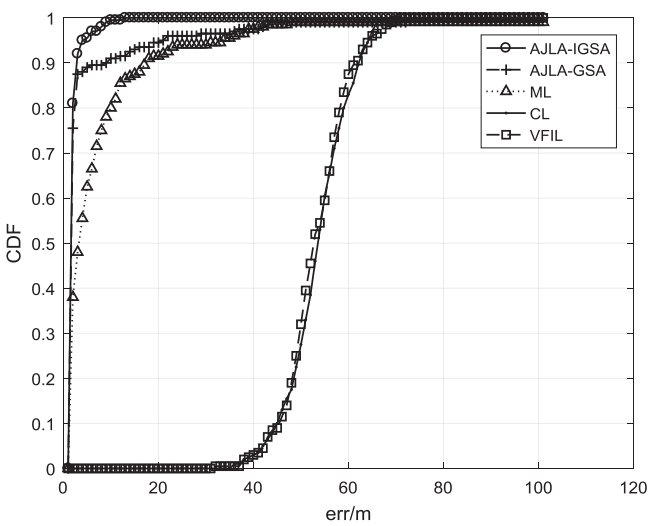


Fig. 7. CDF of localization errors of different algorithms.

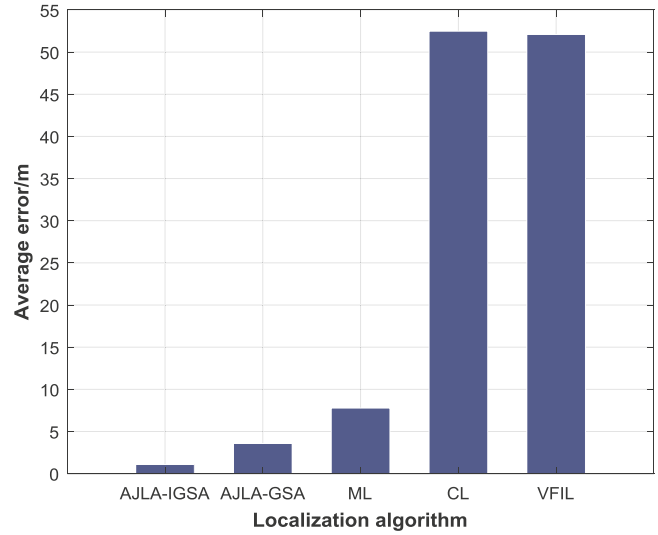


Fig. 8. Mean localization errors of different algorithms.

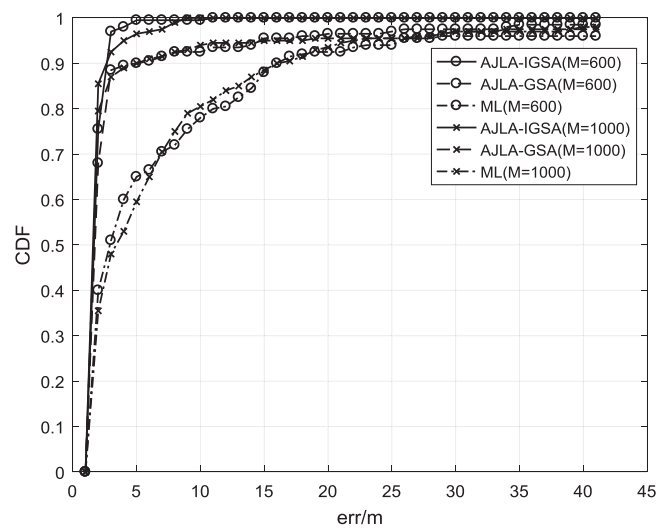


Fig. 9. Localization errors for different nodes' number.

5.2.2. Comparison of localization errors for directional jammer

The AJLA-IGSA, AJLA-GSA, CL, VFIL and ML are employed to locate the directional jammer respectively. The CDF of localization errors and mean errors are illustrated in Figs. 7 and 8 respectively. From the figures, it is obvious that the localization results of CL and VFIL for directional jammer is inaccurate with mean localization errors of 52.5 m and 52.1 m and AJLA-IGSA achieves the best performance among these three algorithms. Besides, the localization errors of AJLA-IGSA, AJLA-GSA are smaller than that of ML. The mean errors for the AJLA-IGSA, AJLA-GSA and ML are 1.9 m, 3.6 m and 7.8 m respectively. The simulation results demonstrate that the proposed algorithm are suitable for the localization of directional jammer and the improved strategies have promoted the localization performance. The simulation results show that typical omnidirectional jammer localization algorithms cannot be utilized to locate directional jammer directly and demonstrate the effectiveness of the proposed AJLA-IGSA.

5.2.3. Impact of nodes' number

The localization performance of different algorithms is compared for the impact of node density. In the simulations, the area of the network remains unchanged and the nodes varies with 600, 800 and 1000. The CDFs of localization errors when the nodes' number is 600 and 1000 are illustrated in Fig. 9 and the mean errors for different algorithms are calculated and shown in Fig. 10. From the figures, we can see that AJLA-IGSA achieves the best performance among these three algorithms. Besides, the mean error of ML decreases as the number of nodes increases, but the mean errors of AJLA-IGSA and AJLA-GSA increases a little. This is due to that as the number of nodes increases, the number of neighbor nodes for each node will increase and thus, the number of nodes that changed to be boundary nodes will increase. Though the number of nodes in the jammer's LNS increases as the number of boundary nodes increases, but the jammed area where the particles are deployed increases. The performance of GSA decreases with the distributed region increases when the number of particles remains unchanged. There is a tradeoff between the number of jammer's LNS and jammed area and the localization errors increase a little for the AJLA-IGSA and AJLA-GSA.

5.2.4. Impact of jamming power

The localization performance of different algorithms is compared for the impact of jamming power. In the simulations, the power of the jammers is set as 10 mW, 14 mW and 25 mW respectively. The CDFs of localization errors when the jamming power is

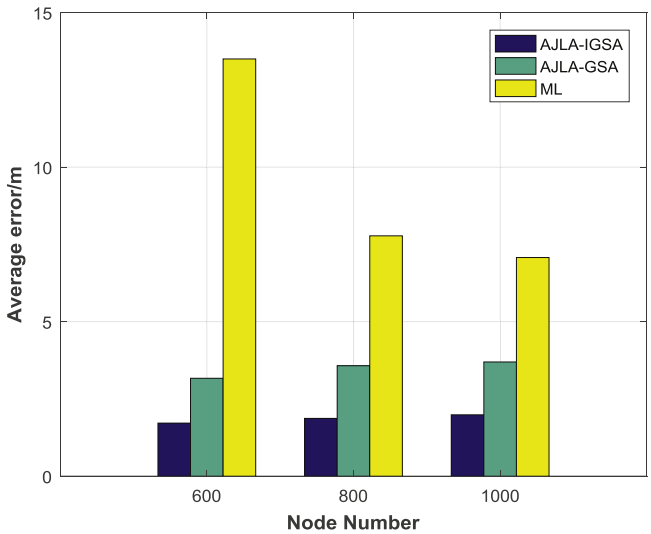


Fig. 10. Mean localization errors for different nodes' number.

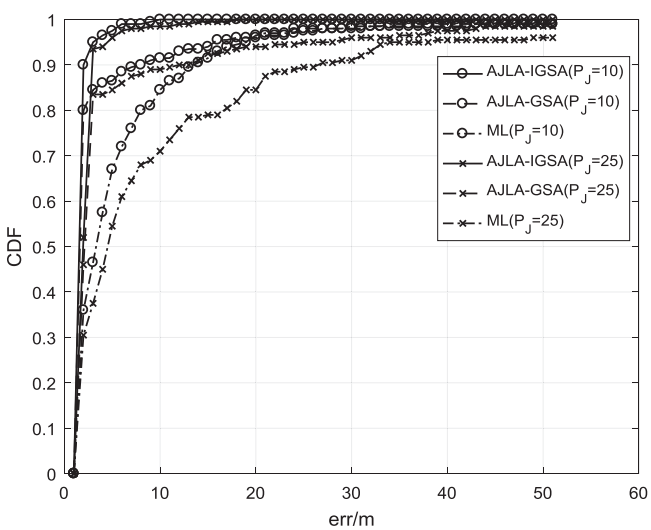


Fig. 11. localization errors for different jamming power.

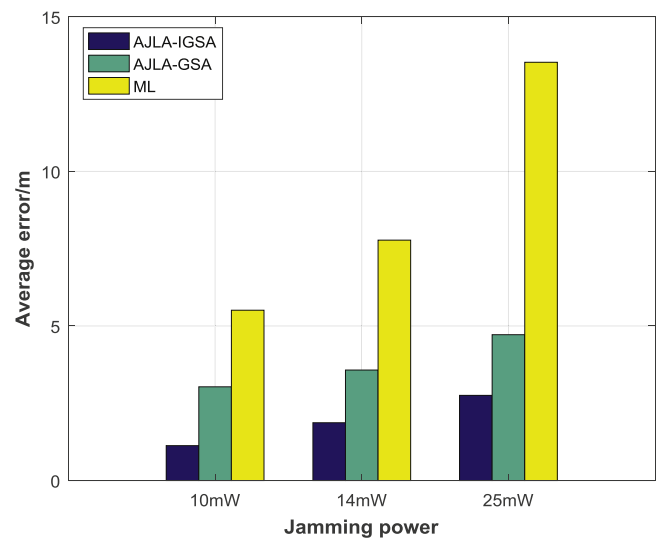
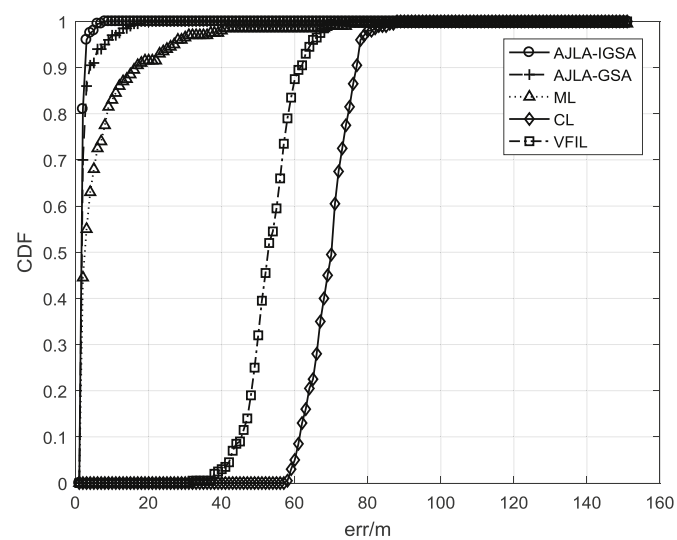
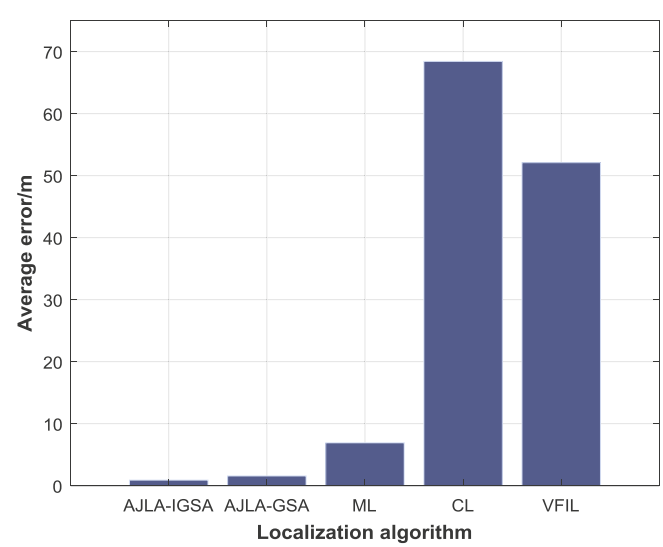


Fig. 12. Mean localization errors for different jamming power.



(a) CDF of localization errors for different jammer's position.



(b) mean localization errors for different jammer's position.

Fig. 13. localization errors for different jammer's position.

10 mW and 25 mW are illustrated in Fig. 11 and the mean errors for different algorithms are calculated and shown in Fig. 12. From the figures, we can see that AJLA-IGSA achieves the best performance among these three algorithms. Besides, the mean errors of these three algorithms increase as jamming power increases. This is due to that as the jamming power increases, the jammed area where the particles are deployed increases. GSA is sensitive to the distribution of the particles and as the distributed region increases, the performance of GSA decrease with the number of particles remains unchanged.

5.3. Impact of parameters

5.3.1. Impact of jammer's position

The localization performance of different algorithms is compared for the impact of jammer's position. In the experiment, the jammer is deployed in the location randomly chosen from [100,700] for both x and y. The CDF of localization errors and mean errors are illustrated in Fig. 13(a) and (b) respectively. From the figures, it can be concluded that the proposed algorithm is independent of the jammer's position. As the jammer deployed in the wireless network randomly, the proposed AJLA-IGSA achieves

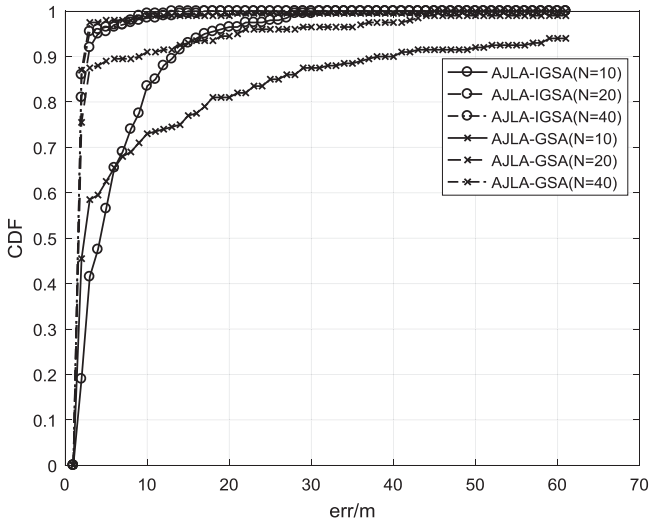


Fig. 14. localization errors for different particle number.

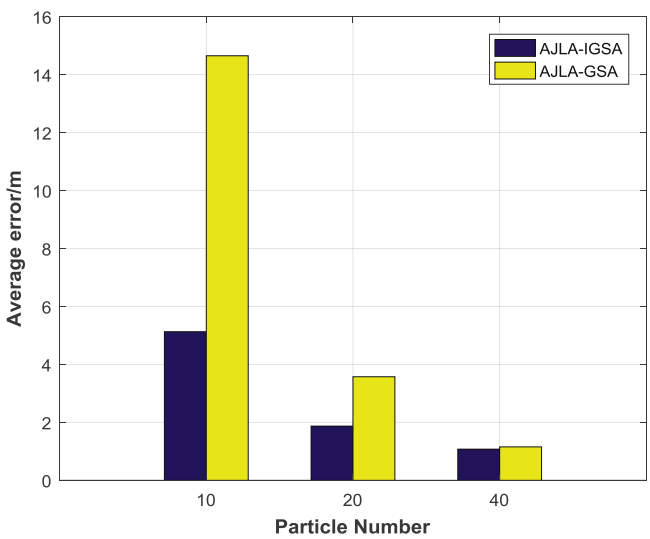


Fig. 15. Mean localization errors for different particle number.

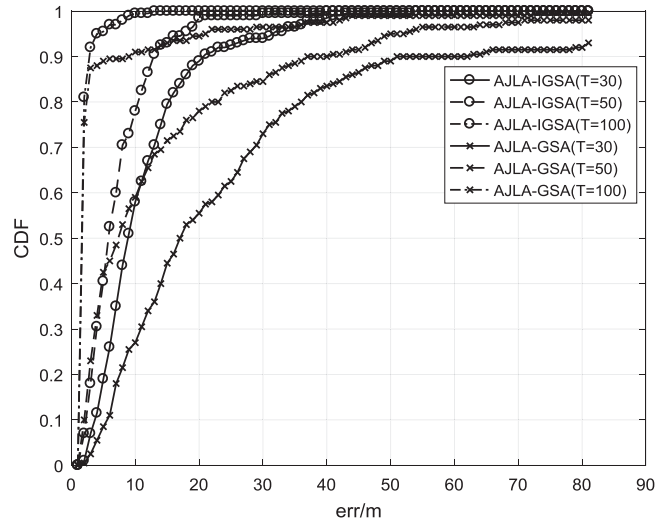


Fig. 16. localization errors for different iterations.

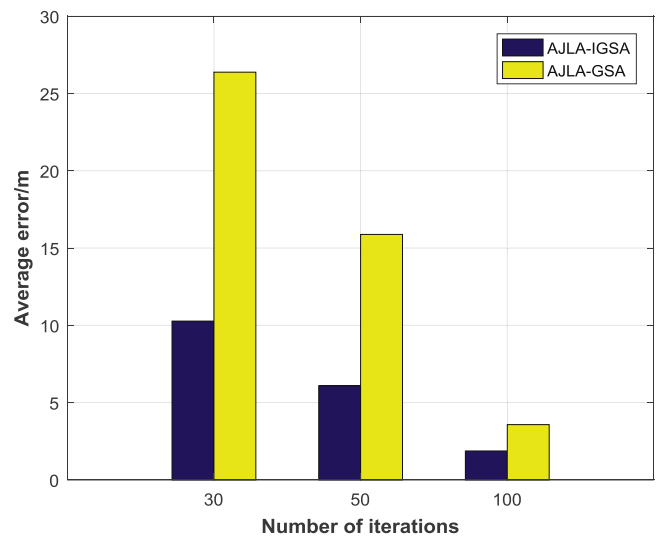


Fig. 17. Mean localization errors for different iterations.

the best performance of these algorithms when locating directional jammer.

5.3.2. Impact of particles' number

The localization performance of different algorithms is compared for the impact of particle' number. In the simulations, the number of particles is set as 10, 20 and 40 respectively. The CDFs of localization errors for AJLA-IGSA and AJLA-GSA when the number of particles is 10 and 40 are illustrated in Fig. 14 and the mean errors for AJLA-IGSA and AJLA-GSA are calculated and shown in Fig. 15. From the figures, we can see that as the number of particles increases, the localization errors of AJLA-IGSA and AJLA-GSA decrease and the performance of AJLA-IGSA is better than that of AJLA-GSA for different particles' number. This is due to that GSA is sensitive to the number of particles. When the particles' number increases, the performance of GSA is improved. Thus, the AJLA-IGSA and AJLA-GSA obtain better results as the number of particles increases.

5.3.3. Impact of iterations

The localization performance of different algorithms is compared for the impact of iterations. In the simulations, the iterations is set as 30, 50 and 100 respectively. The CDFs of localization errors for AJLA-IGSA and AJLA-GSA when the iterations is 30 and 50 are illustrated in Fig. 16 and the mean errors for AJLA-IGSA and AJLA-GSA are calculated and shown in Fig. 17. From the figures, we can see that as the iterations increases, the localization errors of AJLA-IGSA and AJLA-GSA decrease and the performance of AJLA-IGSA is better than that of AJLA-GSA for different iterations. This is due to that as the iterations increases, the GSA will converge continuously. When the iterations increases, the performance of GSA is improved. Thus, the AJLA-IGSA and AJLA-GSA obtain better results as the iterations increases.

5.3.4. Complexity analysis

In order to evaluate the complexity of the proposed algorithm, the delays of information collection and localization process are analyzed respectively. The same scenario is conducted in NS3 in order to calculate the delay of information collection and the transmission rate of data is assumed to be 1 Mbps. Through our experiments, we have conducted the simulations for 200 times independently and the locating node is randomly selected from unaffected nodes for each time. The average delay of information collection is 3.2 s. Based on the collected information, we carry out different localization algorithms in Matlab. The mean delay and average localization errors of different algorithms is shown in Table 2. Though

Table 2
Complexity analysis.

Algorithm	CL	VFIL	ML	IGSA-S10	IGSA-S20	IGSA-S40
Time complexity	3 μs	20 μs	30 ms	40 ms	60 ms	0.14 s
Average error	52.5 m	52.1 m	7.78 m	5.13 m	1.96 m	1.08 m

the time complexity of CL and VFIL is lower than others, the average error can be hardly accepted. The time complexity of IGSA-based localization algorithm increases with the number of particles. Considering the backoff between time complexity and average error, the number of particles is assumed to be 20. Based on this assumption, the time complexity can be acceptable and the localization error is small enough.

6. Conclusion and future work

In order to locate the jammer with no prior knowledge about the type of jammer's antenna, this paper puts forward an Adaptive Jammer Localization Algorithm (AJLA). At first, a collection protocol is designed to collect the information of boundary nodes. Then, an identification method based on the topology of jammed area is employed to derive the type of jammer's antenna. When the jammer is determined to be omnidirectional jammer, typical existing algorithm, i.e., Centroid Localization (CL), Virtual Force Iteration Localization (VFIL) can be adopted to locate the jammer directly. When the jammer is equipped with directional antenna, an Improved Gravitational Search Algorithm (IGSA) is designed to estimate the jammer's position. At last, a series of simulation experiments are conducted to evaluate the correctness and effectiveness of the proposed algorithm. Experimental results show that AJLA-IGSA could locate the directional jammer efficiently and accurately.

In the future, we will take different directional antenna types into consideration, including flat antenna, sampling function antenna etc. The transmission pattern of different directional antenna is various, which increases the challenges for range-based localization method.

Acknowledgments

This research was supported in part by the National Natural Science Foundation of China under Grant No. 61402521 and No. 61471392, Jiangsu Province Natural Science Foundation of China under Grant No. BK20140068, No. BK20150201 and No. BK20140070, and Major State Basic Research Development Program of China (973 Program) No. 2012CB315806.

References

[1] K. Pelechrinis, M. Iliofotou, S.V. Krishnamurthy, Denial of service attacks in wireless networks: the case of jammers, IEEE Commun. Surv. Tut. 13 (13) (2011) 245–257.
[2] X. Wei, Q. Wang, T. Wang, J. Fan, Jammer localization in multi-hop wireless network: a comprehensive survey, IEEE Commun. Surv. Tut. 19 (2) (2016) 765–799.
[3] Y. Zou, J. Zhu, X. Wang, L. Hanzo, A survey on wireless security: technical challenges, recent advances, and future trends (2015).
[4] H. Liu, X. Wenyuan, Y. Chen, Z. Liu, Localizing jammers in wireless networks, in: IEEE International Conference on Pervasive Computing and Communications, 2009, pp. 1–6.
[5] T. Cheng, P. Li, S. Zhu, An algorithm for jammer localization in wireless sensor networks, in: IEEE International Conference on Advanced Information NETWORKING and Applications, 2012, pp. 724–731.
[6] L. Li, S. Zhu, D. Torrieri, S. Jajodia, Self-healing wireless networks under insider jamming attacks, in: Communications and Network Security (CNS), 2014 IEEE Conference on, IEEE, 2014, pp. 220–228.
[7] W. Xu, W. Trappe, Y. Zhang, T. Wood, The feasibility of launching and detecting jamming attacks in wireless networks, in: International Symposium on Mobile Ad Hoc Networking and Computing, 2005, pp. 46–57.
[8] Z. Zhang, J. Wu, J. Deng, M. Qiu, Jamming ack attack to wireless networks and a mitigation approach, in: Global Telecommunications Conference, 2008. IEEE GLOBECOM, 2008, pp. 1–5.
[9] Y.W. Law, L.V. Hoesel, J. Doumen, P. Hartel, P. Havinga, Energy-efficient link-layer jamming attacks against wireless sensor network MAC protocols, in: ACM Workshop on Security of Ad Hoc and Sensor Networks, Sasn 2005, Alexandria, VA, USA, November, 2005, pp. 76–88.
[10] S. Biswas, J. Miic, V. Miic, DDoS attack on WAVE-enabled VANET through synchronization, in: Global Communications Conference, 2012, pp. 1079–1084.
[11] Z. Yang, P. Cheng, J. Chen, Learning-based jamming attack against low-duty-cycle networks, IEEE Trans. Depend. Secure Comput. (99) (2015). 1–1
[12] W. Xu, K. Ma, W. Trappe, Y. Zhang, Jamming sensor networks: attack and defense strategies, IEEE Netw. 20 (3) (2006) 41–47.
[13] A.G. Fragkiadakis, V.A. Siris, N.E. Petroulakis, A.P. Traganitis, Anomaly-based intrusion detection of jamming attacks, local versus collaborative detection, Wireless Commun. Mobile Comput. 15 (2) (2015) 276–294.
[14] N. Sufyan, N.A. Saqib, M. Zia, Detection of jamming attacks in 802.11b wireless networks, Eurasip J. Wireless Commun. Networking 2013 (1) (2013) 208.
[15] O. Punal, I. Aktas, C.J. Schnelke, G. Abidin, K. Wehrle, J. Gross, Machine learning-based jamming detection for IEEE 802.11: design and experimental evaluation, in: IEEE International Symposium on A World of Wireless, Mobile and Multimedia Networks, 2014, pp. 1–10.
[16] X. Zou, J. Deng, Detection of Fabricated CTS Packet Attacks in Wireless LANs, Springer Berlin Heidelberg, 2010.
[17] J. Soryal, T. Saadawi, IEEE 802.11 denial of service attack detection in manet, in: Wireless Telecommunications Symposium, 2012, pp. 1–8.
[18] M. Li, S. Salinas, P. Li, J. Sun, X. Huang, Mac-layer selfish misbehavior in ieee 802.11 ad hoc networks: detection and defense, IEEE Trans. Mob. Comput. 14 (6) (2015) 1203–1217.
[19] A.D. Wood, J.A. Stankovic, S.H. Son, Jam: a jammed-area mapping service for sensor networks, in: Real-Time Systems Symposium, 2003. RTSS 2003. 24th IEEE, IEEE, 2003, pp. 286–297.
[20] K. Pelechrinis, I. Koutsopoulos, I. Broustis, S.V. Krishnamurthy, Lightweight jammer localization in wireless networks: System design and implementation, in: Global Telecommunications Conference, 2009. GLOBECOM, 2010, pp. 1–6.
[21] Q. Wang, X. Wei, J. Fan, T. Wang, Q. Sun, A step further of PDR-based jammer localization through dynamic power adaptation, in: Wireless Communications, Networking and Mobile Computing (WiCOM 2015), 11th International Conference on, IET, 2015, pp. 1–6.
[22] Z. Liu, H. Liu, W. Xu, Y. Chen, Exploiting jamming-caused neighbor changes for jammer localization, IEEE Trans. Parallel Distrib. Syst. 23 (3) (2012) 547–555.
[23] Z. Liu, H. Liu, W. Xu, Y. Chen, An error-minimizing framework for localizing jammers in wireless networks, IEEE Trans. Parallel Distrib. Syst. 25 (2) (2013) 508–517.
[24] X. Wei, T. Wang, C. Tang, J. Fan, Collaborative mobile jammer tracking in multi-hop wireless network, Fut. Gen. Comput. Syst. 78 (2018) 1027–1039.
[25] J. Fan, Q. Wang, X. Wei, T. Wang, Jammer location-oriented noise node elimination method for MHWN, Int. J. Mobile Comput. Multimedia Commun. 6 (4) (2014) 1–19.
[26] Y.Q. Sun, X.D. Wang, X.M. Zhou, Geometry-covering based localization for jamming attack in wireless sensor networks, J. Commun. 31 (11) (2010) 10–16.
[27] J. Zhang, L. Xu, Q. Shen, X. Ji, Localization for Jamming Attack in Wireless Sensor Networks, Springer International Publishing, 2015.
[28] T. Cheng, P. Li, S. Zhu, D. Torrieri, M-cluster and x-ray: two methods for multi--jammer localization in wireless sensor networks, Int. Comput. Aided Eng. 21 (1) (2014) 19–34.
[29] H. Liu, Z. Liu, Y. Chen, W. Xu, Localizing multiple jamming attackers in wireless networks 6574 (4) (2011) 517–528.
[30] Q. Wang, J. Fan, X. Wei, T. Wang, Multi-jammers localization for multi-hop wireless network, J. Commun. 37 (13) (2016) 176–186.
[31] Z. Liu, H. Liu, W. Xu, Y. Chen, Error minimizing jammer localization through smart estimation of ambient noise, in: 2012 IEEE 9th International Conference on Mobile Ad-Hoc and Sensor Systems (MASS 2012), 2012, pp. 308–316.
[32] W. Xu, W. Trappe, Y. Zhang, T. Wood, The feasibility of launching and detecting jamming attacks in wireless networks, in: Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM, 2005, pp. 46–57.
[33] K. Pelechrinis, G. Yan, S. Eidenbenz, S.V. Krishnamurthy, Detecting selfish exploitation of carrier sensing in 802.11 networks, in: INFOCOM 2009, IEEE, IEEE, 2009, pp. 657–665.
[34] L. Mokdad, J. Ben-Othman, A.T. Nguyen, Djavan: detecting jamming attacks in vehicle ad hoc networks, Perform. Eval. 87 (2015) 47–59.
[35] N. Lyamin, A.V. Vinel, M. Jonsson, J. Loo, Real-time detection of denial-of-service attacks in ieee 802.11 p vehicular networks., IEEE Commun. Lett. 18 (1) (2014) 110–113.
[36] E. Rashedi, H. Nezamabadi-Pour, S. Saryazdi, GSA: A gravitational search algorithm, Inf. Sci. 179 (13) (2009) 2232–2248.

Tongxiang Wang was born in Shangdong province, China. He received a Bachelor's degree from the PLA University of Science and Technology, Nanjing, China in 2012. He is now a postgraduate in Army Engineering University. His research interests include Wireless Network Security and Mobile Cloud Computing.

Xianglin Wei was born in Anhui province, China. He received a Bachelor's degree from the Nanjing University of Aeronautics and Astronautics, Nanjing, China in 2007, and his Ph.D. degree in PLA University of Science and Technology, Nanjing, China. He is now working as a researcher at Nanjing Telecommunication Technology Research Institute, Nanjing, China. His research interests include Wireless Network Security, Mobile Cloud Computing, Peer-to-Peer network and distributed system design and optimization. He has served as editorial member of many international journals and TPC member of a number of international conferences.

Jianhua Fan was born in Anhui province, China. He received the Ph.D. degree from Institute of Communication Engineering, Nanjing, China in 1999. Now he is working as a senior researcher at Nanjing Telecommunication Technology Research Institute, Nanjing, China. His research interests include wireless ad hoc network, cognitive network and delay/disruption tolerant network.

Tao Liang received the Ph.D. degree in computer science and technology from the Nanjing Institute of Communications Engineering, Nanjing, China, in 1998. He is currently a Research Fellow of Nanjing Telecommunication Technology Research Institute. His research interests include wireless network protocol, satellite communication technologies and signal processing in communications, physical layer security, cooperative communication, and cognitive network. He is the Senior Member of China Institute of Communications and Electronics.