Defending Trace-Back Attack in 3D Wireless Internet of Things

Junsong Fu, Na Wang, Leyao Nie, Baojiang Cui, and Bharat K. Bhargava, *Life Fellow, IEEE*

***Abstract*— With the development of 5G, it is unsurprising that most of the smart devices in the Internet of Things (IoT) will be wirelessly connected with each other in the near future. This kind of lightweight, scalable and green network architecture will be well-received. In a wide variety of IoT application scenarios, sensor nodes deployed in a local space, such as a multistory building, automatically form a distributed 3D wireless IoT and it can be employed to collect and analyze environmental information. Source-location privacy protection is of great importance in these networks and however, most existing schemes focus on only planar distributed networks which are not suitable for the 3D networks. In this paper, we consider a novel trace-back attack for 3D wireless IoT and then design a source-location privacy protection scheme, named DMR-3D, to defend this kind of novel attacks. In DMR-3D, the source node first selects a set of virtual locations to indirectly choose a set of agent nodes based on the cold start sphere structure and the ellipsoid communication pipeline. Then, a sophisticated mechanism is designed based on both the connected graph and Multiple Delaunay Triangulation (MDT) structure of the network to deliver packets from the source node to the destination node via these agent nodes in a relay manner. Analysis and simulation results illustrate that the proposed scheme can effectively protect source-location privacy with a moderate increment of path stretch, time delay and data transmission amount.**

***Index Terms*— Source-location privacy protection, 3D wireless networks, trace-back attack, geographic routing.**

I. Introduction

WITH the proliferation of Internet of Things (IoT) and 5G, 3D wireless IoT will be widespread in the near future. In fact, quite a few smart wireless networks have been deployed in 3D scenarios in real life, such as in the sky [1], [2], underwater [3]–[5], [51], [52] and underground [6], [7], to perform various tasks. Generally speaking, a local wireless IoT comprises a large number of smart nodes to collect information of surrounding environments and these nodes cooperate with each other to transmit messages in a multi-hop manner. Each node in the network is usually limited in resources such as power, computing and communication, and hence most nodes keep silent (except the basic heartbeat packets) unless they detect events or they are requested to relay packets. Meanwhile, some stronger nodes named sink nodes or gateway nodes may exist in a local network to manage the other common nodes and act as bridges to the remote network operators. Under this circumstance, wireless distributed networks suffer many security threats [45]–[47] and we will discuss it in Section II. Though numerous approaches have been designed to defend against these attacks [48], [49], some contextual-information-based attacks still exist.

In a wireless IoT, a source node is defined as the initial node that generates data packets of events. Note that, the source nodes are different from relay nodes that receives and retransmits packets subsequently. As discussed in [8], [9], and [31], the locations of source nodes are of great physical significance in a wide range of applications considering that the source nodes are naturally close to the event locations. For example, in a wild animal monitor network [8], the adversary can easily locate the animals once they find the source nodes; in a 3D unmanned aerial vehicle (UAV) network [2], the source nodes are usually near to the important targets and emergencies. Events may include locating endangered animals or observing emergencies, and the events information need to be reported to sink nodes in time.

In data collection process, most data generated by the source nodes are delivered to sink nodes and hence they are the destinations of a huge amount of data packets. For a set of neighbor smart nodes, their data delivery paths to the sink node trend to aggregate with each other in most existing routing algorithms [22], [44]. This can be explained by the fact that the next hop of a packet is always selected by a constant rule, such as minimal time delay and minimal hops. Consider geographic routing algorithms in both 2D and 3D distributed networks as examples. These schemes have been attracting a lot of attentions, because they are distributed, lightweight and of high scalability. In these algorithms, the nodes select the next hop of a packet mainly based on the local decisions, such as choosing the nearest neighbor node to the sink node, and the rule is deterministic in general. Therefore, the routing paths with similar sources gather together and more importantly, this leads to the imbalance of traffic amount in different regions of the networks.

The constant data collection pattern and imbalanced traffic distribution bring a potential threat to the source-location privacy and they make it possible for the adversary to locate the source node based on traffic analysis. In real networks, even it is difficult to totally monitor the traffic of a network, it is likely that the adversary can find some valuable clues about the source nodes. In fact, several attacks, such as

Manuscript received 11 June 2020; revised 26 March 2021 and 10 December 2021; accepted 2 February 2022; approved by IEEE/ACM TRANSACTIONS ON NETWORKING Editor K. Ren. Date of publication 10 February 2022; date of current version 18 August 2022. This work was supported by the National Natural Science Foundation of China under Grant 62001055 and Grant 62102017. *(Corresponding authors: Junsong Fu; Na Wang.)*

Junsong Fu, Leyao Nie, and Baojiang Cui are with the School of Cyberspace Security, Beijing University of Posts and Telecommunications, Beijing 100876, China (e-mail: fujs@bupt.edu.cn; nieleyao@bupt.edu.cn; cuibj@bupt.edu.cn).

Na Wang is with the School of Cyber Science and Technology, Beihang University, Beijing 100191, China (e-mail: nawang@buaa.edu.cn).

Bharat K. Bhargava is with the Department of Computer Science, Purdue University, West Lafayette, IN 47906 USA (e-mail: bbshail@purdue.edu).

Digital Object Identifier 10.1109/TNET.2022.3149293

HotSpot Locating [8] and Enhanced HotSpot Locating [10], have been specifically proposed for the adversary to locate the source nodes in planar wireless networks and meanwhile the adversary can locate the targets which trigger the events. This phenomenon is defined as the source-location privacy leakage problem and it is crucial to design proper approaches to strictly protect the privacy of source nodes' locations. These above attack models are designed for planar networks and they cannot be directly employed in 3D networks. The attack models in 2D networks and that in 3D networks are quite different with each other in terms of packet pipeline monitoring, boundary detection and inside trace back pattern. More details about the attack model in 3D networks will be provided in Section IV.A.

In planar distributed networks, many approaches have been proposed [8]–[12], [14] to hide source-location privacy. Two adversary models, i.e., global adversary model and local adversary model, have been widely employed in existing schemes. In global-adversary-based schemes, all the nodes periodically send dummy packets to the sink nodes to confuse the adversary even they did not realize any event. These schemes have two obvious defects. First, they incur a much larger data transmission amount because an extra amount of redundant packets will be extensively generated and transmitted in the network. This will greatly shorten the lifetime of networks. Second, the time delay of information is enlarged because the data can be only transmitted during specified time slices rather than any time.

Local-adversary-based schemes are the mainstream and we roughly divide them into two categories, i.e., random-routing-based schemes and cloud-based schemes. The former ones protect source-location privacy by diversifying routing paths and making it difficult for the adversary to trace back. However, the paths usually cannot be controlled by the source node and the effectiveness of privacy protection can be further improved. For example, in the phantom routing algorithm [11], the fake source nodes are always about k hops far from the real source node and this property makes it easy for the adversary to locate the suspicious region of the source node. Cloud-based schemes [8], [10] first construct an anonymous cloud around the source node and then the nodes on the border of the cloud act as agent nodes to retransmit the packets. In fact, these schemes are the combination of global-adversary-based schemes and random-routing-based schemes. Therefore, they inherit both advantages and disadvantages from these two types of schemes. They can be further improved in terms of energy-efficiency and effectiveness.

Note that, all the above privacy protection schemes are designed for planar networks and they cannot be directly employed in the 3D wireless networks. This can be explained by the fact that 2D and 3D networks are different in terms of network topology, routing algorithms, data collection patterns, *et al.* As a consequence, the privacy protection schemes are also different with each other. For example, the sector-based directed random walk pattern in phantom routing algorithm [11] does not suit the 3D networks; the merge phase of the cloud in [8] cannot used in 3D networks; the data transmission method in [10] between agent nodes does not work in 3D networks. Moreover, even we stiffly extend these schemes into 3D scenario, these privacy protection schemes need to be improved in defending the new attack model proposed in this paper. To the best of our knowledge, our scheme is the first attempt to solve the source-location privacy protection problem specially for 3D wireless networks.

In this paper, we first extend the HotSpot Locating attack to 3D wireless networks based on the new properties of 3D communication pipelines. The new trace-back attack model comprises two modes, including surface trace-back pattern and inside trace-back pattern. The adversary nodes in surface tracing mode always monitor the surface of the packet pipeline. Meanwhile, the nodes in inside tracing mode is responsible of calculating the number of packets go through the cross-section of the pipeline. By cooperating with each other, the adversary nodes together decide the best direction of trace back. Moreover, we design a set of quantitative measurements related to the security of source-location privacy. More details about the attack model will be discussed in Section IV.

To defend against the new trace-back attack, we first assume that proper encryption techniques have been employed in wireless networks [8], [20], [21]. Based on these techniques, we assume that the describing information of an event in a data packet is encrypted by the source node and the adversary cannot decrypt it in time. Considering that these schemes are mature and have been widely researched in a variety of scenarios, we do not detailedly introduce them in this paper for the sake of simplicity.

Based on the network and attack models, we design a novel source-location privacy protection scheme, named DMR-3D, based on the geographic information of smart nodes for 3D wireless IoT. We first design an optimal set of geometry paths between a source node and the sink node without considering the locations of sensor nodes. Then, we will discuss how to deliver data packets strictly along with the geometry paths in a totally distributed manner. In this way, the delivery paths of data packets are carefully designed and they are of high untraceability.

To design the optimal geometry paths between a source node and the sink node, we first propose four basic principles in Section V. Then, a cold-start sphere structure and an ellipsoid-based communication pipeline are constructed. To send a packet to the sink node, the source node $source$ first selects one or two virtual locations for different scenarios and they indirectly decide one or two agent nodes. Then, a packet generated by the $source$ will not be directly delivered to the sink node $dest$ and instead, it is transmitted in a relay manner by the agent nodes. In this way, the virtual locations can roughly decide the shape of a routing path and they are intensely related to performance of DMR-3D in defending trace-back attacks.

In Section VI, we first define the agent nodes as the entity nodes nearest to the virtual locations. Because the source node does not know the exact location of the agent node, a great challenge is how to transmit a packet from the source node to the agent node. In this paper, we design a distributed algorithm to properly deliver a packet to an agent node based on the Multiple Delaunay Triangulation (MDT) structure. Though

only one or two agent node are employed in this paper, our scheme can be easily enhanced by using more agent nodes. In fact, the source node $source$ can select a set of virtual locations $\{l_1, l_2, \cdots, l_m\}$ and they indirectly define a set of ordered agent nodes $\{a_1, a_2, \cdots, a_m\}$. Then, a packet can be transmitted in a relay manner, i.e., $source(a_0) \dashrightarrow a_1 \dashrightarrow a_2 \dashrightarrow \cdots \dashrightarrow a_m \dashrightarrow dest$.

DMR-3D can securely transmit packets from the source nodes to the sink node in an untraceable way. Even for the same source node and sink node, the packet routing paths are totally different with each other. In this way, the packet density of whole network is extremely low and the adversary cannot trace back to the source node easily. Moreover, all the routing paths can be strictly controlled by the source node and hence we can dynamically optimize the routing paths in response to the trace-back strategy of the adversary. This will be discussed in our future work.

The main contributions of this paper are presented as follows:

- We extend HotSpot Locating attack to 3D wireless networks which can easily trace back to the source nodes if the network employs existing routing algorithms.
- A set of atomic factors related to source location privacy protection are summarized. Two comprehensive measurements in terms of a routing algorithm's effectiveness and the security of a real network are also provided.
- A mechanism is designed based on ellipsoid structure to construct the optimized shape of the routing paths corresponding to the attack model.
- We propose a sophisticated algorithm based on Multiple Delaunay Triangulation structure to deliver a packet to an agent node based on a virtual location only.
- A series of experiments are conducted to evaluate the performance of our scheme in terms of packet density, source-location privacy security, routing path stretch, time delay and data transmission amount.

The rest of this paper is organized as follows. In Section II, we first review different threats to the networks. Then we summarize the related studies in source-location privacy protection schemes for planar networks and geographic routing algorithms for 3D networks. Network model and trace-back attack are presented in Section III and IV, respectively. Section V discusses how to construct the shape of the routing paths to diversify them. In Section VI, we introduce the process of packet delivery from source node to sink node in detail. Theoretical analysis of path extension and security of DMR-3D is discussed in Section VII. We evaluate the performance of the proposed scheme in Section VIII. At last, Section IX concludes this paper.

II. Related Work

A. Security Threats to the Distributed Wireless Networks

Distributed wireless networks are vulnerable to many threats and we can divide the security challenges into two categories including content security and contextual security [8]. We briefly summarize existing threats in the following.

Data security is one of the most important consideration in distributed networks and many schemes are designed to improve content security of the networks. Lou *et al.* [47] improve data confidentiality by mapping the message to a set of shares based on secret sharing schemes and delivering the shares to the sink node through independent routing paths. Liu *et al.* [48] and Mahmoud *et al.* [49] also propose secure data delivery approaches to improve data security of networks.

In the field of contextual security, Fan *et al.* [46] propose a privacy-preserving scheme against traffic analysis attack. By employing homomorphic encryption operation on global encoding vectors, the proposed scheme offers two significant privacy-preserving features, i.e., packet flow untraceability and message content confidentiality. Obviously, we can divide the source-location privacy security to the category of contextual security.

Moreover, node security has been also widely researched. As an example, clone detection is a promising technique to efficiently eliminate the malicious nodes and improve node security. Zheng *et al.* [45] proposed an energy-efficient location-aware clone detection protocol to located the compromised nodes. Specifically, the scheme randomly selects a set of witnesses located in a ring area to verify the legitimacy of sensors. Even 10 percent of the nodes are compromised, the clone detection probability still approaches 98 percent.

B. Source-Location Privacy Protection in Distributed Networks

Recently, most source location privacy protection approaches [27]–[30] are designed for 2D distributed wireless networks. We mainly focus on the local-adversary-based schemes which share similar adversary model with our scheme.

Phantom routing algorithm [11] is the most popular random-routing-based source-location privacy protection scheme and it contains two phases. In the first phase, the source node sends the packet to any neighbor node and the packet is randomly delivered for k steps based on random walk model. In the second phase, the node that receives the packet after k steps of walk is denoted as the fake source node and it is responsible for sending the packet to the sink node. In fact, the fake source node can employ any routing algorithm to deliver the packet and hence the adversary can locate a set of fake sources. Considering that all the fake source nodes are about k steps far from the source node, the adversary can easily find the source node.

To defend HotSpot Locating attack, Mahmoud and Shen [8] propose a cloud-based scheme that builds a cloud with irregular shape to hide the real source nodes. In its delivery process, each packet is updated at each node by encrypting the packet with different secret keys. In this way, the adversary cannot distinguish whether two packets are the same with each other. The most important disadvantage of this scheme is that a large data transmission amount is consumed in the cloud. Moreover, if the adversary can successfully outline the cloud, he can finally locate the source nodes by global adversary model.

Wang *et al.* [10] propose a novel scheme named SPAC and improve the cloud-based scheme [8] in all-around way. SPAC seamlessly integrates a lightweight secret sharing scheme into the whole approach and the original messages are mapped to

shares which are much shorter in length. To decrease data transmission amount, an anonymous cloud around the source node is constructed based on the shares rather than the original messages. All the behaviors of nodes in the cloud are carefully designed to make them indistinguishable and hence the source node is hidden. Moreover, simulation results show that the secret sharing scheme also improves the reliability of data delivery process.

C. Geographic Routing Algorithms in 3D Wireless Networks

Source-location privacy leakage is strongly related to the distributed data collection schemes in which the packages are transmitted in a multi-hop manner. In existing schemes, geographic routing algorithms have been attracting a lot of attentions because of their scalability and conciseness [32]–[35]. Most schemes include a greedy-based packet forwarding mode and the main difference among them is how to overcome the local minima. We summarize the representative algorithms in the following.

Liu and Wu [18] proposed the localized partial unit Delaunay triangulation (PUDT) algorithm in a distributed way. The whole network space is divided into a set of tetrahedrons and irregular polyhedrons. They then propose greedy-hull-greedy (GHG) routing algorithm which recovers from local minima using a hull-based, depth-first search strategy to forward packets. To recover a local minimum, only the nodes on the corresponding polyhedron rather than all the nodes are visited and this improves the search efficiency.

Zhou *et al.* [16] extend the Greedy Distributed Spanning Tree Routing (GDSTR) to three-dimension version, GDSTR-3D, by projecting nodes from 3D space onto orthogonal 2D planes to achieve a balance between computing consumption and performance. GDSTR-3D recovers local minima by travelling on hull routing subtree which contain the reachable destination node. However, the researchers are discontented with GHG's unrealistic assumptions of unit ball graph.

Lam and Qian [15] propose Multihop Delaunay Triangulation (MDT) algorithm that leverages the guaranteed delivery property of Delaunay triangulation graphs and use only 1-hop greedy forwarding to get better resilience and reliability. Moreover, the guaranteed delivery property holds even the coordinates of the nodes are inaccurate. Simulation results illustrate that MDT provides a low routing stretch.

Sarkar *et al.* [26] avoid to design recovery mechanisms for greedy forwarding and instead they "reform" the network to make it suitable for greedy forwarding mode. In particular, they first extract a planar triangulation of the network graph with a set of holes. Then, they construct a conformal map based on Ricci flow such that all the holes are mapped to perfect circles. In this case, the greedy forwarding will never get stuck at a node.

It can be observed that none of the above data collection algorithms take source location privacy into consideration. When computing routing paths, all the schemes produce the routing paths based on determinate inbuilt algorithms. This kind of determination leads to considerable vulnerability in source location privacy.

III. Network Model and Adversary Model

A. Network Model

Without loss of generality, we take the underwater wireless networks [3]–[5], [51], [52] as examples to illustrate our scheme. Underwater networks can be employed in a wide range of applications, such as resource exploration, pollution monitoring, wild animal monitoring, tactical surveillance and *et al.* For convenience, we consider a huge network which is used to monitor and trace the endangered wild animals (e.g., the tunas). Once a tuna is detected, the source nodes send information to the sink node through multi-hop manner. Each node in the network is strictly limited in resources such as capabilities of computing, storage, communication and power. However, we assume that they can properly execute all the pre-stored instructions. All the nodes are homogeneous and the communication radius of the nodes is assumed to be R. The nodes in the network can locate themselves by proper methods [41]–[43] and each node consists of a 3-tuple vector to indicate its location.

We assume that only one sink node exists in the network and it locates in the center of the network space. The sink node is assumed to be much stronger than the common nodes in the network and it can communicate with remote network users by telecommunicating. By broadcasting the related information after network deployment, we assume that the location of the sink node is known to all the nodes in the network. All the information generated by a source node is encrypted based on the pairwise key with the sink node and the packets cannot be decrypted by the adversary. Note that, the location of the destination sink node is stored in plaintext because all the relay nodes need this common information. Therefore, the adversary can also get the destination of a packet using an analyzer. However, the information about the agent nodes need to be encrypted in the packets by a pair of neighbor nodes in the delivery process. The pairwise key negotiation algorithm has been widely discussed in distributed networks [8], [10] and any proper algorithm can be employed in the proposed source-location privacy protection scheme.

In the new trace-back attack model which will be discussed in Section IV, the adversary starts to trace back from the sink node. Therefore, if the source node is very close to the sink node, the adversary can easily locate the monitored targets. In this case, locating the source node in a random manner is unavoidable and fortunately this is of a low probability in a large network. In this paper, we assume that the distance between the source node and sink node is larger than $2R_s$, where R_s is the radius of the cold start sphere which will be discussed in Section V.B.

B. Adversary Model

In this paper, we assume that the adversary can detect the data transmission behaviors of a sensor node if he is close to the node with a distance not larger than R. For a large 3D wireless network, it is extremely difficult for an adversary to

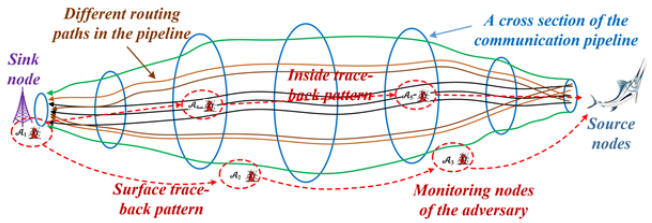

Fig. 1. Trace-back attack in 3D wireless networks.

search the tunas in a random manner. A more efficient way is locating the source node by the trace-back attack and then search the tunas around the source nodes. We assume that the adversary employs a set of monitoring devices to execute trace-back attack. The monitoring devices have the following three important characteristics.

Passive. We assume that the goal of the adversary is passively locating the source node rather than actively attacking the network. This is reasonable considering that the adversary will lose the chance to locate the tuna once his behaviors are detected by the network operators [8], [9].

Well-equipped. The monitoring devices of the adversary is assumed to be well-equipped with antennas, spectrum analyzers and mobile modules. Then, the adversary can receive local packets and locate their senders. Moreover, the devices can freely move in the network to trace the packets.

Collaborative. Each monitoring device can communicate with the other nearby devices. Moreover, they can share the collected information in real time to yield the optimal strategy of trace back in a collaborative manner.

IV. Trace-Back Attack Based on Traffic Analysis in 3D Wireless IoT

A. Trace-Back Attack in 3D Networks Based on Traffic Analysis

As shown in Fig. 1, we take the tuna monitor network as an example to discuss the trace-back attack model. Once a set of smart nodes find a tuna that moves around, they begin to continuously generate and send packets to the sink node. Though the locations of source nodes are slightly different, their routing paths to the sink node are close with each other for general algorithms and this will be illustrated by experiments in Section VIII.B. For any routing algorithm, though the paths in the start and middle phases between source nodes and sink node may diffuse with each other, they trend to gather together while they move closer and closer to the sink node as shown in Fig. 1. Without loss of generality, we named the smallest 3D irregular pipeline that encases all the routing paths as the communication pipeline. For a stable data stream, data sending rate in the pipeline significantly larger than that of the outside. We denote the boundary between the inside and outside of the pipeline as the surface.

In the simplified version of trace-back attack which has been widely employed in existing schemes [9], [11], the adversary first stays around the sink node and waits until his monitoring devices probes a signal which indicates that a packet has been sent to the sink node. By analyzing the source of the signal, he moves to the sender node, sn, and waits until another related packet is sent to node sn. By iterating the above process, the adversary can find the source nodes at last.

To improve the successful rate of trace-back, a more complicated attack model is considered based on the capabilities of monitoring devices. The new attack is conducted by analyzing the traffic in the pipeline rather than staring at the behaviors of several local nodes. The adversary can successfully attack the network with a high probability if the tuna stays around a field for a period of time. The workflow of the novel trace-back attack in a 3D wireless network is shown in Fig. 1. It can be observed that the nodes in the new trace-back attack comprises two patterns, i.e., the surface trace-back pattern and the inside trace-back pattern. These two patterns are presented as follows.

Surface trace-back pattern. A set of adversary nodes on the surface together can sketch the shape of cross-sections as shown as the blue circles in Fig. 1 and this information is shared with the nodes in inside trace-back pattern. The adversary can easily identify the surface of a communication pipeline by observing the large difference in packet sending rate of the nodes in the two sides of the surface. In trace-back process, the monitoring nodes move on the surface step by step until they gather together around the source nodes.

Inside trace-back pattern. In this pattern, all the nodes locate inside the pipeline to monitor all the packets that walk through the cross-sections. The adversary follows the high data transmission rate of the relay nodes. The monitoring nodes repeatedly moves to the source node step by step until the source nodes are located. Note that, the direction of trace-back is opposite to the direction of packet delivery.

The monitoring nodes on the surface of the communication pipeline and the nodes inside the pipeline jointly monitor the whole cross-section of the pipeline as the blue circles shown in Fig. 1. The surface trace-back pattern decides the area that the inside trace-back needs to monitor. Meanwhile, the inside trace-back pattern decides the direction of trace back. In this way, the adversary can move to the source node step by step. The back tracing process terminates when the adversary nodes gather together and the packet sending amount through the cross-section decreases sharply. We say that the trace-back attack is successful when at least one source node is located, i.e., the distance between a monitoring device and a source node is smaller than R.

Theoretically, the adversary can always trace back to the source nodes based on the proposed trace-back attack if he has enough monitoring devices and the packet stream is stable. However, even the adversary has few monitoring devices and the stream is slightly dynamic, he still can locate the source node with a high probability for most existing data collection algorithms. For example, in the shortest path routing algorithm or the directed diffusion algorithm, the routing paths gather so fast that the cross-sections can be totally monitored with only several monitoring nodes.

B. Quantization of Source-Location Privacy Security Under the Proposed Trace-Back Attack Model

By analyzing the proposed attack model, we can infer that whether the trace-back attack works depends on many related

factors. In this section, we summarize four main atomic factors related with the effect of trace-back attack, i.e., *the size*, S, *of the cross-section*, *the rate*, F, *of packet flow, the time period*, T, *of the target stays around an area* and *the number*, N_{md}, *of the adversary's monitoring devices*. We discuss these four related atomic factors in detail as follows.

Size of the cross-section $\boldsymbol{S}$. It can be observed from Fig. 1 that the difficulty of trace back is strongly related with the sizes of pipe's cross-sections. For a larger cross-section, more monitoring devices are needed to totally monitor it. If the adversary obtains only a part of the information, it is challenging for the adversary to construct the optimal trace back strategy.

Rate of packet flow $\boldsymbol{F}$. This factor is defined as the number of packets that are delivered from the source nodes to the sink node in a period of time and it decides the number of packets that can be monitored by the adversary. More packets indicate a clearer direction of trace back and meanwhile the adversary can walk more steps towards the source node.

Time period of a tuna stays in an area $\boldsymbol{T}$. This factor is quite straightforward. A larger T leads to a stable packet stream and this decreases the difficulty of trace back.

Number of monitoring devices $\boldsymbol{N_{md}}$. From the perspective of the adversary, a straightforward strategy to improve the success rate of trace-back attack is employing more monitoring devices. In the extreme case, if the adversary has enough monitoring nodes, he becomes a global adversary and any random routing algorithm cannot defend his attack.

Based on the definitions of the above four atomic factors, we can observe that the first factor is an inherent property of a random routing algorithm and the last three factors are related with the network user, monitored target and adversary, respectively. Considering that the atomic factors are trivial, we design another two comprehensive measurements to evaluate the security of source-location privacy.

Packet density $\boldsymbol{Den}$. By integrating the first three atomic factors, *Packet density* D is proposed to evaluate the effectiveness of a routing algorithm in terms of source location privacy protection. We define *packet density* as follows:

$$Den = F * T / \int_{source}^{dest} S, \tag{1}$$

where $source$ is the source node, $dest$ is the destination node and the distance between them is $2d$. We will provide the details of calculating Den in Section IV.C. Note that, a lower packet density means an outstanding performance of a routing algorithm. In the extreme case, if the density generated by a routing algorithm can be ignored compared with the heart-beat packets density in the network, the source node is completely secure in terms of trace-back attack.

Source node privacy security $\boldsymbol{Sec}$. The security of the source node in a real network is quantized by *Source node privacy security* Sec, which is a combination of all the four atomic factors. By combing packet density and number of monitoring devices, the security of a source node privacy in a real network can be calculated as follows:

$$Sec = 1/(N_{md} * Den). \tag{2}$$

It can be easily inferred from equation (2) that the larger of Sec, the securer of the network.

Though all the atomic factors are related with the security of source-location privacy in a routing algorithm, it is unwise to decrease the rate of packet flow, F, considering that we need to guarantee the quality of network service. Parameter T and N_{md} also cannot be controlled by the network operators. Consequently, a good choice of protecting source-location privacy is enlarging the size of the cross-sections of a routing algorithm and this is the main idea of our scheme.

C. Calculation of Packet Density

Though packet density is clearly defined in equation (1), it is extremely difficult to accurately calculate Den. To get the packet density of a routing algorithm Alg in simulation, we first select a set of nearby source nodes $source$ and a destination node $dest$. Then, we employ Alg to generate a set of routing paths $Path_1, Path_2, \cdots, Path_k$. Each path, $Path_i$, comprises a set of ordered relay nodes $\langle source, n_1, n_2, \cdots, dest\rangle$ which can be treated as a set of ordered points. For the sake of convenience, we build a new coordinate system with $source$ as the original point and source-destination line as an axis. The other two axes can be randomly selected as long as all the three axes are perpendicular with each other. Then, all the locations of nodes in $Path_i$ are transferred to new points in the new coordinate system.

We need to first eliminate the outliers of the routing paths before calculating packet density and then only the mainstream routing paths are employed to calculate Den. This is reasonable considering that a small set of outliers can greatly decrease the packet density and meanwhile, they cannot significantly improve the security of source-location privacy. In this paper, we used DBSCAN clustering algorithm [39] to detect the outliers of the routing paths. A great challenge is how to defined the pairwise distances of the paths. Considering that a routing path stretchable and compressible, we employ the classic Dynamic Time Warping (DTW) [38] to calculate the distance between each pair of routing paths. For a pair of paths, $Path_i$ and $Path_j$, with length n and m, which consist of a series of three-dimensional points $x_1 \ldots x_n \; and \; y_1 \ldots y_m$, their DTW distance is denoted as $D(n, m)$, and it can be calculated in a dynamic programming approach [40]:

$$D(i,j) = min \left\{ \begin{array}{c} D(i, j-1) \\ D(i-1, j) \\ D(i-1, j-1) \end{array} \right\} + d(x_i, y_j). \tag{3}$$

Given a set of selected routing paths without outliers, we calculate $\int_{source}^{dest} S$ in a discrete manner. Specifically, we uniformly choose t cross-sections of the paths and the area of a cross-section is defined as the area of the convex hull of the points on the cross-section. Then, $\int_{source}^{dest} S$ can be approximately calculated as follows:

$$\int_{scr}^{des} S = \sum\nolimits_{i=1}^{t} S_i * \frac{2d}{t} \tag{4}$$

where S_i is the area of i-th cross-section, $2d$ is the distance between $source$ and $dest$. The calculation results of packet

densities of DMR-3D and existing algorithms will be presented and analyzed in Section VIII.B.

V. Construction of Optimal Packet Delivery Geometry Paths Between Source Node and Sink Node

A. Basic Principles of Designing Optimal Geometry Paths

By analyzing the steps of trace-back attack proposed in this paper, we employ four basic design principles of geometry paths and they are summarized as follows:

Stifling trace-back attack in the cradle. The packet delivery pattern around the sink node decides the difficulty of trace-back attack in the initial phase. A good start can greatly improve the success probability of locating the source node. Therefore, the optimal set of paths should conceal all the clues about the direction of the source node around the sink node. The cold start structure will be discussed in the following.

Enlarging the cross-section of communication pipelines. In trace-back attack, the adversary needs to monitor the cross-sections of the pipeline. A large cross-section leads to a large number of needed monitoring devices and hence it greatly increases the difficulty of attack. As a consequence, we should expand the communication pipeline as large as possible.

Diversifying the routing paths. A stable data stream between the source node and the sink node makes it easy to trace back. On one hand, even for the same source node and sink node, the paths are should be very different with each other. On the other hand, the paths in a short period a time should be also different.

Limiting the maximum length of routing paths. Apparently, both enlarging the cross-sections and diversifying the routing paths trend to increase the lengths of the routing paths. However, a longer path means a larger amount of energy consumption which is another big concern in resource-limited networks. In this paper, we assume that the maximal length of a path cannot beyond $l_{max} = \rho * 2d$, where $2d$ is the Euclidean distance between the source node and sink node, and $\rho(\rho \geq 1)$ is a preset parameter.

The randomness and untraceability of packet delivery paths decides the difficulty of back tracing. Based on these above design principles, we design a novel geometry structure of the routing paths and it is discussed in the following.

B. Cold Start Path Structure

As discussed in Section IV.A, the adversary initially stays around the sink node and he traces back to the source nodes step by step based on continuous packet flows. To stop the trace-back attack in the initial, we construct a sphere in the surrounding region of the sink node, i.e., the black sphere in Fig. 2. As shown of the blue lines, all the packets are first uniformly transmitted to the surface of the sphere and then they are sent to the sink node from the surface. From the adversary's perspective, the packets are uniformly transmitted from all the directions and he cannot extract any valuable information about the direction of the source nodes. In this way, we stifle the trace-back attack in the cradle. The effectiveness of this structure is related with the radius, R_s, of the sphere. The larger of R_s, the more difficult for the adversary to trace back. However, a larger R_s increases the average length of the packet delivery paths and more energy are consumed. Considering that the maximum length of routing paths is limited, when we set parameter R_s, a proper balance between source-location privacy and energy consumption of the whole network needs to be considered.

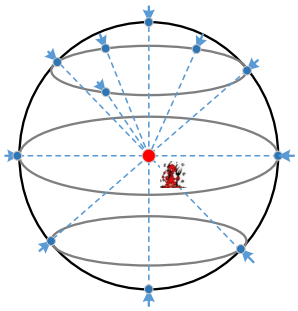

Fig. 2. Cold start path structure of trace-back attack in 3D wireless networks.

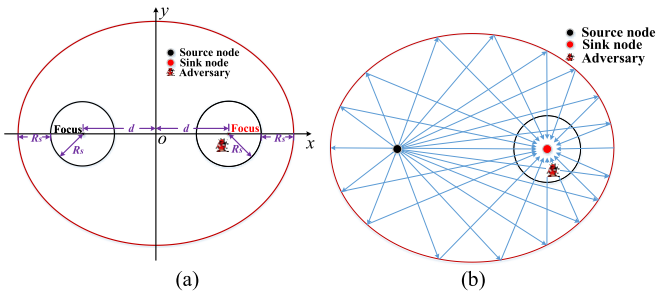

Fig. 3. Ellipsoid-based routing path construction. (a) Cross-section of ellipsoid communication pipeline. (b) Different routing paths from the source node to the sink node.

C. Ellipsoid-Based Routing Path Construction

To maximize the cross-section of communication pipeline under the assumption that the lengths of all the geometry paths are limited by l_{max}, we can infer that the agent nodes must locate on a ellipsoid with the source node and sink node as two focuses. Obviously, the ellipsoid structure needs to be integrated with the cold start path structure to form a completed path between the source node and sink node. Moreover, the ellipsoid cannot be randomly selected, because it must completely contain the cold start sphere. The construction process of the ellipsoid is discussed as follows.

After choosing a preset radius R_s of the cold start sphere, we need to design the shape of routing paths given a source node and a sink node with distance $2d(d \geq R_s)$. For simplicity sake, we present a cross-section of the ellipsoid as shown in Fig. 3(a). The source node first builds a coordinate system based on the location of itself and the sink node. The left focus and right focus of the ellipsoid are defined as the source node and sink node, respectively. Based on the definition of a ellipsoid, we know that the sum distance between each focus to any point on the ellipsoid is constant and it is denoted as $2a$. In this paper, we set $a = d + 2R_s$ and it is straightforward to prove that the cold-start sphere around the sink node is completely surrounded by the ellipsoid.

Based on the ellipsoid, we can refract a packet from the source node to the sink node uniformly as shown in Fig. 3(b). For each path from the source node to the sink node, at least

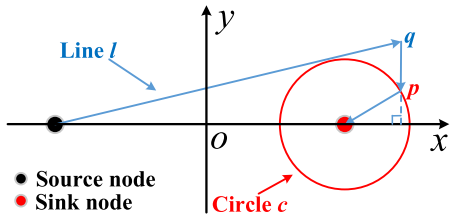

Fig. 4. Packet delivery with two agent nodes.

one virtual location on the ellipse, corresponding to an agent node, is selected by the source node. Then, the packets can be sent to the sink node from all the directions in a relay manner. Note that, the selected virtual locations are not uniformly distributed on the ellipsoid and instead, the points of intersection between the paths and the cloud-start sphere should be uniformly scattered on the sphere. In this way, from the adversary's perspective, the packets are delivered to the sink node from all the directions uniformly.

D. Packet Delivery to the Dark Side of the Cold-Start Sphere

A drawback of the scheme in Section V.C is that some paths between the source node and agent nodes cross the cold-start sphere around the sink node as shown as the yellow paths in Fig. 3(b). This may leak some information about the source node to the adversary. As shown in Fig. 4, we design a supplementary method to deliver a packet from the source node to the sink node by two agent nodes. Assume that a packet should be transmitted to the sink node from the direction of p which cannot be refracted by a node on the ellipsoid. Then, the source node first sends the packet along line l which is tangent to circle c until it reaches point q which is right above point p. Point q acts as an agent node and sends the packet to point p where the packet is sent to the sink node finally. By combining these two cases, we conclude that the source node can always send a packet to the sink node from any direction without crossing the cold-start sphere.

E. Selection of Agent Nodes' Location Based on Geometry Path

It can be observed that the complete path between the source node and sink node comprises several line segments and some turning points exist on the geometry path. To lead data packets walking along the designed paths, intuitively, we can deploy a set of entity agent nodes on the turning points. However, this is impractical in real networks and hence we propose another scheme. Specifically, we set the locations of turning points as virtual locations and then the data packets are delivered to a node near to the virtual location. In a network with densely deployed nodes, this mechanism is a very good choice. The details of transmitting a packet from the source node to the sink node strictly along the designed geometry paths will be discussed in Section VI.

VI. Packets Delivery Between the Source Node and the Sink Node Along With the Geometry Paths

A. Framework of Packet Delivery

In this section, we discuss how to deliver a packet along with the optimized path designed in Section V. Apparently,

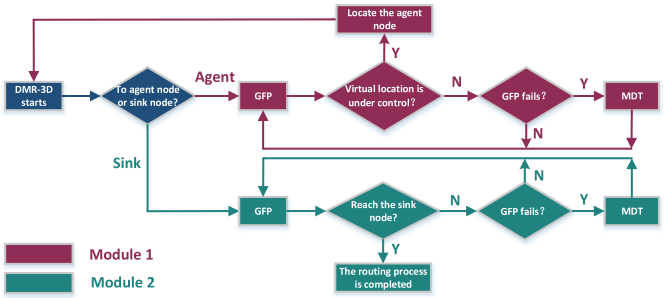

Fig. 5. Framework of packet delivery between source node and sink node in DMR-3D.

the virtual locations are not entity smart nodes of the network and they cannot act as agent nodes. Consequently, we first define an agent node as follows.

Definition 1 (An Agent Node): The *agent node* a_i corresponding to a virtual location l_i is defined as the entity smart node closest to l_i in the whole network.

With the help of agent nodes, a packet will be delivered from source node $source$ to the sink node $dest$ in the following path:

$$source(a_0) \dashrightarrow a_1 \dashrightarrow a_2 \dashrightarrow \cdots \dashrightarrow a_m \dashrightarrow dest,$$

where $\dashrightarrow$ means delivering a packet in a multi-hop manner. In the process of delivering a packet from a_i to $a_{i+1}(0 \leq i \leq m-1)$, the packet needs to store two destinations, i.e., the virtual location l_{i+1} corresponding to node a_{i+1} and location of the sink node. The whole data delivery process can be decomposed to two modes, i.e., delivering a packet to an agent node based on a virtual location and delivering a packet to the sink node.

As shown in Fig. 5, the framework of packet delivery process mainly comprises two modules according to the destination of a packet. In Module 1, the destination of the packet is an agent node and on the contrary the destination in Module 2 is the sink node. Module 1 is repeatedly executed until the packet reaches the final agent node, a_m, since which Module 2 is employed. These two modules are slightly different with each other because the agent node is indeterminate in advance and the sink node is always determinate.

In both Module 1 and Module 2, greedy forwarding pattern (GFP) and Multiple Delaunay Triangulation Pattern (MDT) are always two important data delivery patterns. GFP is of great efficiency in moving packets toward the destination node and we discuss it in Section VI.B. We say that the GFP of a packet fails on a node if the node cannot directly communicate with the destination node and it has no neighbor that is closer to the destination node.

In Module 1, the Greedy mode fails in two cases. In case 1, the packet is always stuck at the nearest node to virtual location l_{i+1}. This is reasonable considering that there is no entity node on l_{i+1}. In this case, the agent node corresponding to l_i is located successfully and a new data delivery process starts. Locating the agent node based on the virtual location is the main challenge in module 1. Mathematically speaking, a node a_i is an agent node to a virtual location l_i if and only if the Voronoi polyhedron of a_i covers l_i. Based on the property of Voronoi polyhedron [36], [37], we can infer

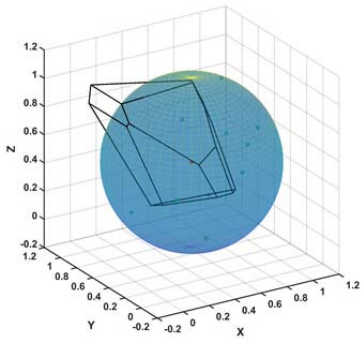

Fig. 6. Voronoi polyhedron of the red node and a local minimum occurs at the red node when des locates in the polyhedron and outside the sphere.

that if l_i locates out of the polyhedron, a_i can always find a neighbor (a physical neighbor or a MDT neighbor) closer to l_i. We will discuss how to locate an agent node based on a virtual location in Section VI.C. In case 2, the packet is stuck in a local optimized result between two agent node. In this case, we recover GFP by MDT and this will be discussed in Section VI.D.

Compared with Module 1, Module 2 is quite straightforward. In Module 2, the Greedy mode can fail only in one case, i.e., case 2 in module 1, and the solution to the problem is similar to that in Module 1 which will be discussed in Section VI.D.

B. Greedy Forward Pattern

In packet delivery process, the destination of a packet is denoted as a position $(x_{dest}, y_{dest}, z_{dest})$. In packet delivery process, the next hop is locally decided by the node and the choice is directly related with $(x_{dest}, y_{dest}, z_{dest})$. In greedy forward pattern (GFP), node n, with position (x_n, y_n, z_n), always transmit the packet to the neighbor node nei, with position $(x_{nei}, y_{nei}, z_{nei})$ that is closest to the sink node. Note that, the choice should also meet the following criteria:

$$\begin{aligned}&(x_{nei} - x_{dest})^2 + (y_{nei} - y_{dest})^2 + (z_{nei} - z_{dest})^2 \\ &\quad < (x_n - x_{dest})^2 + (y_n - y_{dest})^2 + (z_n - z_{dest})^2 \end{aligned} \quad (5)$$

In this way, we can guarantee that the packet continuously walks closer and closer to the sink node. In most cases, the packet can be successfully delivered to the sink node in only the Greedy mode [22]. Once there is no neighbor node of n satisfies inequation (5), we say that GFP fails and we need to process it with different methods based on different cases.

C. Locating the Agent Nodes Based on Voronoi Polyhedron and 3D Delaunay Triangulation

We first discuss how to process GFP failure in case 1. Consider a set of nodes in a 3D space, each node n_i corresponds to a Voronoi polyhedron P_i and, compared with all the other nodes, n_i is closest to any point in P_i. As an example shown in Fig. 6, the Voronoi polyhedron P_i of the red node is constrained by the black edges. We say that a virtual location is under control of a node if the virtual location locates in the Voronoi polyhedron of the node. Meanwhile, we can infer that the node is just the agent node corresponding to the virtual location based on definition 1. The communication range of the red node is represented by the orange sphere. Each face of P_i is the perpendicular bisector of n_i and another node in the network. By connecting all the pairs of nodes if they correspond to a face in a Voronoi polyhedron, the 3D Delaunay Triangulation (DT) of the nodes are constructed successfully [50].

By combining the network connectivity graph and 3D DT, we get the Multiple Delaunay Triangulation (MDT) graph as shown in Fig. 7. In Fig. 7(a), a pair of nodes is connected in the network connectivity graph if the nodes can directly communicate with each other, i.e., the distance between them is smaller than R. Fig. 7(b) presents the 3D DT structure of all the nodes in the network. The 3D DT structure divides the network space into a set of disjoint tetrahedrons and the circumcircle of each tetrahedron contains no other nodes of the network except for the four vertexes. Then, as shown in Fig. 7(c), A pair of nodes is connected in MDT graph if they are connected in the network connectivity graph or 3D DT graph. In the following, we call the neighbor nodes in connectivity graph, DT graph as physical neighbors and DT neighbors, respectively. Moreover, we call the neighbors exist in MDT graph and do not exist in connectivity graph as MDT neighbors.

It has been proved that GFP never fails on MDT graph [36], [37] and hence we can always delivery a packet to the destination node by employing GFP on MDT. However, if the destination node does not exist or is not connected to the network, the packet will be delivered to the nearest node to the destination node. Based on this property, we can easily locate an agent node corresponding to a virtual location by MDT. Though a pair of connected nodes in MDT may cannot directly communicate with each other, they can communicate by a multi-hops path, which is stored in the forwarding table [15]. Apparently, delivering a packet to a DT neighbor is more complex than that to a physical neighbor. As a consequence, we always execute GFP on the connectivity graph and overcomes local minima on the DT graph. Overall, our scheme is designed based on the MDT graph.

D. Local Minima Recovery Based on MDT

Compared with 2D networks, more local minima tend to occur in 3D topology. As an example shown in Fig. 6, a packet is stuck at the red node if the destination locates out of the Voronoi polyhedron and there is no neighbor node nearer to the destination. Once GFP fails in case 2, we employ MDT structure for recovery.

A node n_i in the network needs to store a forwarding table for each MDT neighbor. An entry in its forwarding table [15] is denoted as $\langle source, relay_1, relay_2, \cdots, dest\rangle$ where $source$ is n_i, $dest$ is a MDT neighbor of n_i, $relay_j$ are a set of relay nodes between $source$ and $dest$.

When a packet is stuck at node n_i, it needs to check all its MDT neighbors to find a neighbor closer to $dest$ and decides the next step based on whether the neighbor exists. If the closer neighbor to $dest$ does not exist, we can infer that n_i is the closest node to $dest$ in the network according to the property

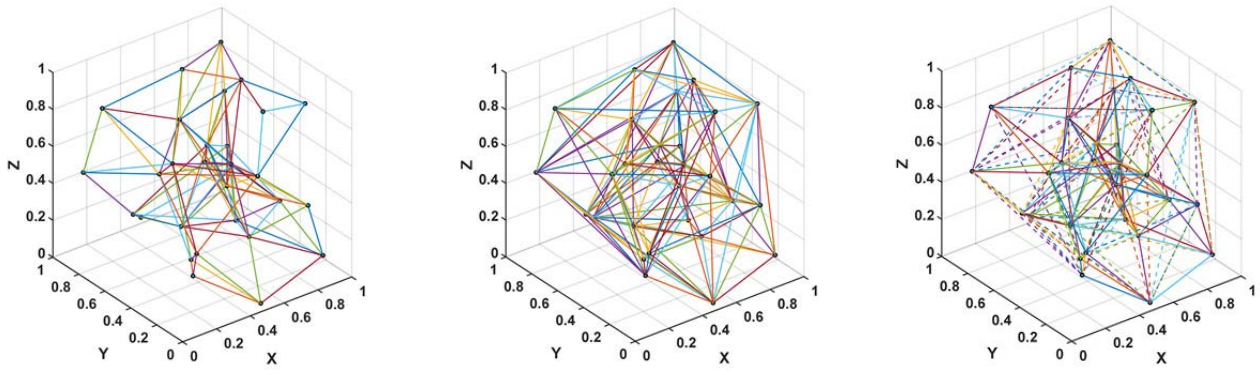

Fig. 7. (a) 3D Network Connectivity graph; (b) 3D Delaunay Triangulation graph; (c) 3D Multiple Delaunay Triangulation graph.

of MDT. According to definition 1, node n_i is the agent node in Module 1. In Module 2, if the neighbor does not exist, we can infer that $dest$ is not connected to the network and the routing process fails. If the closer neighbor to $dest$ exists, node n_i directly sends the packet to the neighbor based on the forwarding table and the packet can turn back to GFP at the neighbor node. In this case, the data delivery process can be continuously executed until the packet is delivered to an agent node or the sink node.

VII. Theoretical Analysis of the Proposed Scheme

A. Source Location Privacy Protection

In this paper, we assume that the adversary has no background knowledge about the source node and hence he first needs to analyze the traffic information around the sink node. In most existing routing algorithms, the adversary can get sustained and steady packet streams. Moreover, these streams are very thin and it is easy to totally monitor the cross-sections with only several monitoring devices. Therefore, the adversary can always trace back to the source nodes.

In DMR-3D, the designed path structure is very interesting and it increases the difficulty of trace back attack with the following properties:

- In the initial phase of trace back, the adversary cannot extract any information about source location because the packets are transmitted to the sink node from all the directions in a random manner. The trace back attack needs to solve the cold start problem.
- The pipeline between the source node and sink node is very large in size and it does not shrink in all the path (even around the source node). This makes it of great difficulty to monitor a cross-section and extract valuable information.
- The source node can totally control the shape of the paths and the delivery paths of two neighboring packets are totally different with each other. As a consequence, it is impossible for the adversary to obtain sustained packet streams.
- The source node can dynamically change the strategy of generating routing paths according to the attack strategy of the adversary. This further improves the security of source node and we will try to introduce the game theory into our scheme in the future work.

In conclusion, DMR-3D can protect source location privacy from several aspects in theory and we will verify this through simulations in Section VIII.

B. Routing Path Stretch

In DMR-3D, the source node employs one or two agent nodes to deliver packets in a relay manner. In general, the agent nodes do not locate on the line between the source node and the sink node. As a consequence, the length of routing paths increases compared with the shortest paths. In this section, we theoretically analyze the average length between the source node and sink node.

As shown in Fig. 3(b), most routing paths have one agent node and the total distance of each path is $2a = 2d + 4R_s$. Some other paths may have two agent nodes and the distance of a path must be not larger than $\frac{4d^2+2R_s^2+6dR_s}{\sqrt{4d^2-R_s^2}} + R_s$ which tends to $2a$ with the increasing of $\frac{d}{R_s}$. Considering that the length of most paths is $2a$ and quite a small portion of paths are slightly shorter or longer than $2a$, the average length of a path in the proposed scheme is approximately equals to $2a$. Specifically, when $\frac{d}{R_s}$ is selected from the set $3, 5, 10, 20$, the average length of paths are enlarged by 1.67, 1.40, 1.20, 1.10 times compared with the shortest path. It can be observed that for a constant R_s, with the increasing of d, the difficulty of trace back increases and meanwhile the routing stretch of our scheme monotonously decreases. This is reasonable considering that if the source node is close to the sink node, the trace-back difficulty is low and we need to greatly improve it though some extra prices are payed.

In conclusion, our scheme can dynamically adjust its strategy to construct the routing paths and this achieves a balance between the security of source node and packet delivery efficiency.

C. Promotion of Our Scheme

In this paper, only one or two agent nodes are employed to relay the packets from the source node to the sink node. It is unnecessary to use more agent nodes to diversify the routing paths, because simulation results show that our scheme

can properly defend against the proposed trace back attack. However, our scheme can be further improved to defend stronger adversaries. In theory, we can select any number of agent nodes in a routing path and it is quite straightforward as shown in Fig. 5.

VIII. Performance Evaluation

A. Experiment Settings

We have built up a discrete event simulator for 3D wireless smart networks based on ns3. The whole simulation is conducted on a DELL tower server with two intel CPUs and 128G memory. To thoroughly evaluate the performance of our scheme, we simulate an extremely large wireless network with $24,000$ smart nodes randomly deployed in a $2000\mathrm{m} \times 2000\mathrm{m} \times 2000\mathrm{m}$ cubic space. In simulation, the radius of the nodes is set as 120 meters and meanwhile the average degree of the nodes, i.e., the average number of neighbors of the nodes in the network, is about 20. A powerful sink node locates in the center of network and it is the destination of all the packets. For simplicity, we assume that only one target exists in the network. The initial location of the target is randomly selected in the network space and it moves based on random walk model with a speed $1\mathrm{m/s}$. Specifically, the target can move to six directions with $1\mathrm{m}$ in each step. The simulation terminates when the adversary locates the source node successfully or it lasts for 10 minutes. We say that a source node is located when the monitoring nodes of the adversary is close to the source node and their distance is smaller than R. Each simulation is executed for ten times and the average simulation results are presented and analyzed in the following.

In simulation, the smart nodes are redundantly deployed and the target is well monitored if it locates in the network space. A node in the network can detect the target if its distance to the target is not larger than 120 meters and then the node becomes a source node. All the source nodes independently collect the information of the target and then send packets to the sink node through proper routing algorithms. Each packet contains 2048 bits in which the first 96 bits are packet head (at most two agent nodes are employed in a path). The locations related with agent nodes and the sink node are stored in packet heads. In order to match the random walk model of the target, time interval of packet generation in the source nodes is set as 1 second. The radius, R_s, of the sphere around the source node is chosen from {100, 150, 200, 250, 300, 350, 400}. Meanwhile, the half distance, d, between $source$ and $dest$ is selected from {600, 700, 800, 900, 1000, 1100, 1200}. These two parameters greatly affect the performance of our scheme and we will discuss it in the following.

In simulation, we assume that one adversary attempts to locate the target through trace-back attack. The number of monitoring nodes, N_{md}, is chosen from {6, 12}. Further, we assume that all the adversary monitoring nodes form an ad-hoc network and they can share the detected information to decide the next step of trace back collaboratively.

The parameters of simulation are summarized in Table I. Based on these parameters, we evaluate the performance of DMR-3D algorithm in terms of packet density, source-location security, routing path stretch, time delay of data packets and data transmission amount. To the best of our knowledge, DMR-3D is the first scheme specially designed for source-location privacy protection in 3D wireless networks. To thoroughly evaluate our scheme, we extended the routing-based scheme [11] and cloud-based scheme [8] into 3D scenarios. Then, we compare DMR-3D with the modified schemes. In simulation, the steps of random walk in [11] and the size of the cloud in [8] are carefully preset according to the radius of sphere R_s. In this way, it is relative fair to compare them. Moreover, the shortest routing algorithm and MDT [15] algorithm are also employed as benchmarks.

TABLE I
Simulation Parameters

Network size	$2000\mathrm{m} \times 2000\mathrm{m} \times 2000\mathrm{m}$
Number of nodes	24,000
Communication radius	120m
Average number of neighbors	≈ 20
Number of sinks	1
Location of sink	Center of the network
Number of tunas	1
Source nodes	The nodes that can detect the target
Size of the packets	2048bits
Packet generation interval	1s
Number of adversaries	1
Number of monitoring devices N_{md}	{6, 12}
Radius of sphere R_s	{100, 150, 200, 250, 300, 350, 400}
Half distance, d, between $source$ and $dest$	{600, 700, 800, 900, 1000, 1100, 1200}

B. Diversity of Routing Paths and Packet Density

Packet density reflects the performance of a data collection scheme in terms of source location privacy protection. Generally speaking, if the routing paths gather together, the packet density increases. On the contrary, if the routing paths disperse with each other, the packet density decreases and it increases the difficulty of trace back. As shown in Fig. 8(a), the paths of the shortest routing algorithm gather with each other and quite a few packets are delivered on same paths. In DMR-3D, as shown in Fig. 8(b), the paths are totally different with each other with the help of a set of agent nodes. It is unlikely that two packets are delivered on a same path in DMR-3D. Therefore, it is extremely difficult for the adversary to get a complete path between the source node and the sink node. It is almost impossible for the adversary to trace back.

In the following, we calculate the packet densities of different algorithms and a discussion is also provided. As shown in Fig. 9, the packet densities of all the three algorithms monotonously decrease with the increasing of d. This is reasonable considering that $\int_{scr}^{des} S$ increases with d for all the three algorithms. The performance of MDT is very similar

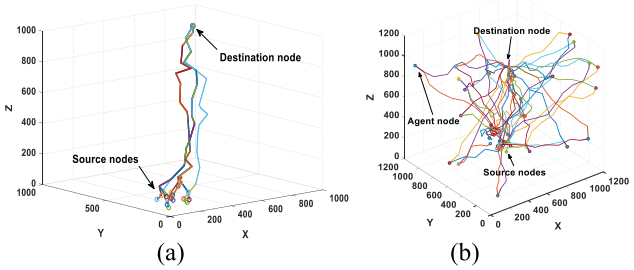

Fig. 8. The paths of (a) shortest routing algorithm gather with each other; (b) DMR-3D disperse with each other.

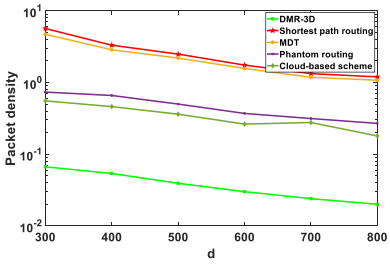

Fig. 9. Packet density Den of different schemes.

to that of the shortest routing paths, because GFP in MDT works well in most cases. The Phantom routing algorithm and the cloud-based scheme performs better. This is reasonable considering that both of them employ a random walk phase to diversify the paths. Compared with these above algorithms, DMR-3D performs much better when we set $R_s = 300$. Specifically, the packet density in DMR-3D is about 1% of that in the first two algorithms and about 10% of the following two schemes. This can be explained by the fact that the routing paths in DMR-3D are considerably dispersive with each other. Packet density is an important indicator about the security of source-location privacy and we can infer that our scheme can provide a proper protection on the locations of source nodes.

C. Source-Location Privacy Security

In this section, we employ two other measurements to evaluate the security of source-location privacy. The first measurement is Sec which has been defined in Section IV.B. The second is the locating probability of the source node and it is calculated as the number of times that the adversary locates the source node to the number of experiment runs.

In this paper, Sec is calculated as $1/(N_{md} * Den)$ and we set $R_s = 300, d = 600$. For different parameter N_{md}, simulation results are presented in Fig. 10. It can be observed that with the increasing of N_{md}, Sec of all the schemes decrease. This can be explained by the fact that the adversary can locate the source nodes with a high probability if he can control more monitor devices. However, our proposed scheme performs much better than the other schemes. This is reasonable, because for the same N_{md}, packet density of our scheme is much smaller than that of existing schemes.

We then analyze the locating probability of source node with different R_s and d, respectively. In simulation, the adversary can almost always locate at least one source node with the help of monitoring devices if the network employs the shortest path routing algorithm and MDT algorithm, as shown in

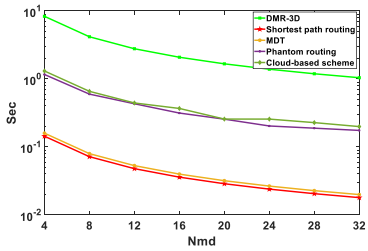

Fig. 10. Sec of different schemes.

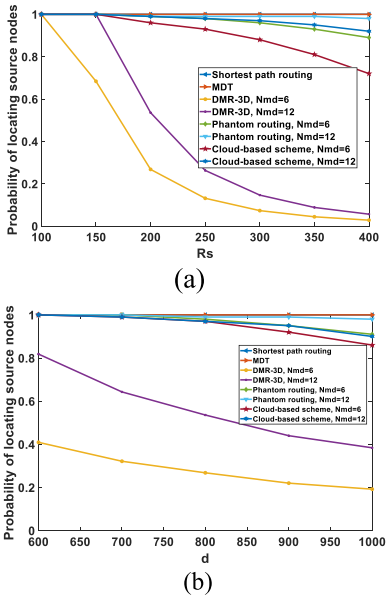

Fig. 11. Locating probability of source node (a) with different radius of sphere R_s, $d = 800\text{m}$; (b) with different distance between $source$ and $dest$, $R_s = 200$.

Fig. 11(a). This is reasonable considering that the routing paths are approximately constant when the target moves slowly in a local area. The adversary can walk to the source node closer and closer until a source node is found. The number of monitoring devices decides the needed time of trace back and the number of located source nodes. With the help of more devices, the adversary can locate more source nodes in a faster way.

In Phantom routing and cloud-based schemes, the initial phase of trace back is quite easy and the difficulty gradually increases until the adversary sketches the hot region (e.g., the cloud) around the source nodes. Once the adversary locates the nodes in the hot region, he can finally locate the source node. These two schemes greatly outperform the former ones because of the employment of random walk phase.

In DMR-3D, we first set $d = 800\text{m}$ and the probability of locating the source nodes with different R_s is presented in Fig. 11(a). With the increasing of R_s, the probability of locating source node rapidly decreases. A larger R_s means a larger communication pipeline between $source$ and $dest$. Consequently, the packet density also decreases and the security of source location privacy increases. For a constant R_s, the adversary can improve the success probability by employing more monitoring devices. However, for a large R_s, the adversary performs badly even with the help of quite a few devices.

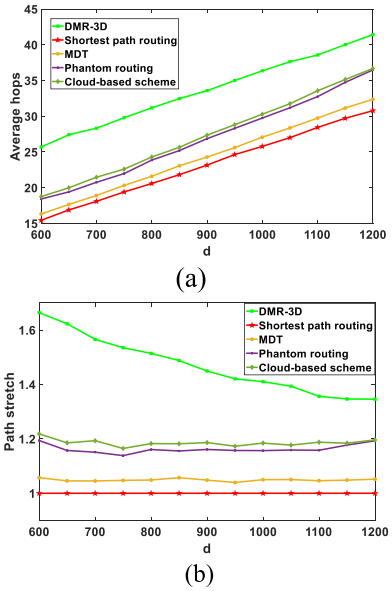

Fig. 12. Routing path stretch. (a) Average hops between $source$ and $dest$; (b) Path stretch compared with the shortest paths.

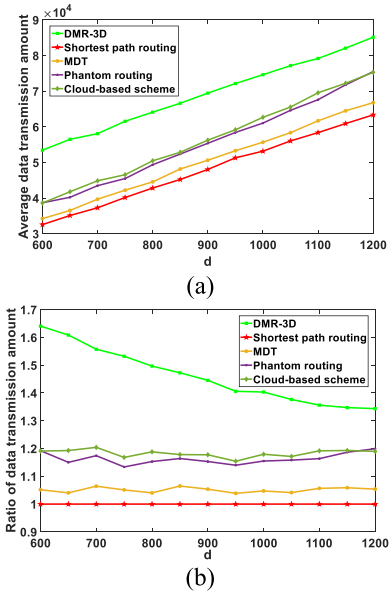

Fig. 13. Time delay of data packets (a) Average time delay; (b) Ratio of time delay compared with the shortest path routing.

When presenting the affection of d on source location security, we set $R_s = 200\text{m}$ and simulation result is presented in Fig. 11(b). The success rate of the adversary moderately decreases with the increasing of d. This is reasonable considering that more steps are needed to trace back for a larger d.

D. Routing Path Stretch

In this section, we verify the average hops between the source node and sink node in the networks for different algorithms. In this section, we set $R_s = 200$ and select d from 600 to 1200. Simulation results are presented in Fig. 12.

It can be observed from Fig. 12(a) that the average hops from $source$ to $dest$ of all three algorithms monotonously increase in an approximate linear manner. The average distances between neighbors are similar with each other for different networks with similar density of nodes. Then, the average distances of a hop are similar with each other. As a consequence, the hops are approximately linear with the distance between $source$ and $dest$.

For a constant d, the average hop of MDT, Phantom and cloud-based scheme is larger than that of the shortest path routing algorithm. This can be explained by the fact that some extra paths are employed in all the three schemes. The average hop of DMR-3D is much larger than that of the other schemes. This is inevitable, because a set of agent nodes are employed to disperse the paths and meanwhile the paths are extended. However, with the increasing of d, the extra price in terms of path stretch monotonously decreases as shown in Fig. 11(b). When we set $d = 1200$, the paths are stretched by 1.33 times in average compared with the shortest paths and this is the extra price of the source-location privacy protection.

E. Time Delay of Data Packets

As shown in Fig. 13(a), the shortest path routing and MDT are of similar and the best performance in time delay. This is straightforward considering that they always select the straight path between the source nodes and the sink node. The time delays of Phantom routing algorithms and cloud-based scheme are slightly larger than the first two schemes because of the random walk phase and the cloud construction phase. The employment of agent nodes in our scheme increases the average length of the paths and apparently the average time delay of data packets also increases. However, the ratio of extra price of DMR-3D decreases with the increasing of d as shown in Fig. 13(b).

F. Data Transmission Amount

As shown in Fig. 14(a), the average data transmission amounts of all these algorithms increases with the increasing of d. This is reasonable considering that a longer path leads to more hops of packet delivery. The performance of the shortest path routing, MDT, and Phantom routing algorithm are similar with each other because the routing paths are short and no dummy packets are transmitted in the network. As the routing paths of DMR-3D are larger than that of existing routing algorithms, the data transmission amount of DMR-3D also increases. However, similar to the routing path stretch, with the increasing of d, the ratio of extra price in terms of data transmission decreases as shown in Fig. 14(b). We can infer that DMR-3D performs better in large wireless networks. Compared with the routing-based schemes, the cloud-based scheme performs much worse. This is quite reasonable considering that a large number of dummy data packets are generated and transmitted in the cloud.

G. Performance Discussion

We thoroughly evaluate the performance of DMR-3D in terms of effectiveness and efficiency. Simulation results illustrate that existing routing algorithms cannot defend the trace-back attack at all. This can be explained by the fact that the objective goals of them are being robust and efficient. Source location privacy is not taken into consideration in the

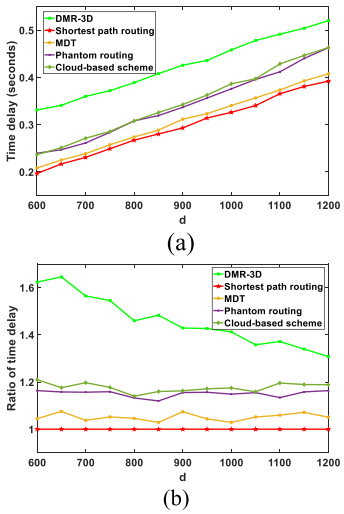

Fig. 14. Data transmission amount (a) Average data transmission amount; (b) Ratio of data transmission amount.

design process. Though the Phantom routing algorithm and cloud-based schemes can be stiffly extended into 3D scenario, their performance can be further improved. DMR-3D can strongly protect the source location privacy because of the low packet density. The price of DMR-3D is that the average length of paths, average time delay of packets and data transmission amount are improved. Fortunately, the prices are under control and they are acceptable in most cases. In conclusion, DMR-3D reaches a good balance between source location privacy and energy efficiency.

IX. Conclusion

In this paper, we propose a novel scheme for 3D wireless IoT to protect the source-location privacy by diversifying the transmission paths of packets. A novel trace-back attack model with two modes is specially designed for 3D wireless network. Meanwhile, a set of quantitative measurements is prosed to evaluate the performance of schemes. To defend the attack, an ellipsoid-based geometry path structure is first constructed between the source node and sink node to improve the difficulty of trace-back. Then, a sophisticated algorithm is designed to deliver the packets strictly along a geometry path. Our scheme reaches a perfect balance between the controllability of a single path and the randomness of a set of routing paths. Simulation results illustrate that our proposed scheme can effectively protect the source-location privacy with a controllable routing stretch, time delay and increasing of data transmission amount.

As future work, we plan to improve DMR-3D in two aspects. First, the MDT structure in this paper cannot be constructed totally in a distributed manner and a challenging problem is designing a novel distributed structure to replace MDT. Second, the communication radii of the nodes are all the same in this paper and we will study new schemes for heterogeneous networks in which the communication radii of different devices can be totally different with each other.

References

[1] Y. Yang, Z. Zheng, K. Bian, L. Song, and Z. Han, "Real-time profiling of fine-grained air quality index distribution using UAV sensing," *IEEE Internet Things J.*, vol. 5, no. 1, pp. 186–198, Feb. 2018.

[2] H. Sedjelmaci, S. M. Senouci, and N. Ansari, "Intrusion detection and ejection framework against lethal attacks in UAV-aided networks: A Bayesian game-theoretic methodology," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1143–1153, May 2017.

[3] D. Pompili, T. Melodia, and I. F. Akyildiz, "Routing algorithms for delay-insensitive and delay-sensitive applications in underwater sensor networks," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2006, pp. 298–309.

[4] J. Yan, X. Yang, X. Luo, and C. Chen, "Energy-efficient data collection over AUV-assisted underwater acoustic sensor network," *IEEE Syst. J.*, vol. 12, no. 4, pp. 3519–3530, Dec. 2018.

[5] P. Braca, R. Goldhahn, G. Ferri, and K. D. LePage, "Distributed information fusion in multistatic sensor networks for underwater surveillance," *IEEE Sensors J.*, vol. 16, no. 11, pp. 4003–4014, Jun. 2016.

[6] U. I. Minhas, I. H. Naqvi, S. Qaisar, K. Ali, S. Shahid, and M. A. Aslam, "A WSN for monitoring and event reporting in underground mine environments," *IEEE Syst. J.*, vol. 12, no. 1, pp. 485–496, Mar. 2018.

[7] M. C. Vuran, A. Salam, R. Wong, and S. Irmak, "Internet of underground things: Sensing and communications on the field for precision agriculture," in *Proc. IEEE 4th World Forum Internet Things (WF-IoT)*, Feb. 2018, pp. 586–591.

[8] M. M. E. A. Mahmoud and X. Shen, "A cloud-based scheme for protecting source-location privacy against hotspot-locating attack in wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, no. 10, pp. 1805–1818, Oct. 2012.

[9] B. Alomair, A. Clark, J. Cuellar, and R. Poovendran, "Toward a statistical framework for source anonymity in sensor networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 2, pp. 248–260, Feb. 2013.

[10] N. Wang, J. Fu, J. Li, and B. K. Bhargava, "Source-location privacy protection based on anonymity cloud in wireless sensor networks," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 100–114, 2020.

[11] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," in *Proc. 25th IEEE Int. Conf. Distrib. Comput. Syst. (ICDCS)*, Jun. 2005, pp. 599–608.

[12] K. Bicakci, H. Gultekin, and B. Tavli, "Maximizing lifetime of event-unobservable wireless sensor networks," *Comput. Standards Interface*, vol. 33, no. 4, pp. 401–410, 2011.

[13] J. Fu, B. Cui, N. Wang, and X. Liu, "A distributed position-based routing algorithm in 3-D wireless industrial Internet of Things," *IEEE Trans. Ind. Informat.*, vol. 15, no. 10, pp. 5664–5673, Oct. 2019.

[14] N. Wang, J. Fu, J. Zeng, and B. K. Bhargava, "Source-location privacy full protection in wireless sensor networks," *Inf. Sci.*, vol. 444, pp. 105–121, May 2018.

[15] S. S. Lam and C. Qian, "Geographic routing in d-dimensional spaces with guaranteed delivery and low stretch," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 663–677, Apr. 2013.

[16] J. Zhou, Y. Chen, B. Leong, and P. S. Sundaramoorthy, "Practical 3D geographic routing for wireless sensor networks," in *Proc. 8th ACM Conf. Embedded Networked Sensor Syst. (SenSys)*, 2010, pp. 337–350.

[17] S. Durocher, D. Kirkpatrick, and L. Narayanan, "On routing with guaranteed delivery in three-dimensional ad hoc wireless networks," *Wireless Netw.*, vol. 16, no. 1, pp. 227–235, Jan. 2010.

[18] C. Liu and J. Wu, "Efficient geometric routing in three dimensional ad hoc networks," in *Proc. IEEE INFOCOM 28th Conf. Comput. Commun.*, Apr. 2009, pp. 2751–2755.

[19] R. Flury and R. Wattenhofer, "Randomized 3D geographic routing," in *Proc. IEEE INFOCOM 27th Conf. Comput. Commun.*, Apr. 2008, pp. 834–842.

[20] M. E. A. Mahmoud, X. Lin, and X. Shen, "Secure and reliable routing protocols for heterogeneous multihop wireless networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 1140–1153, Apr. 2015.

[21] S. Ozdemir and H. Cam, "Integration of false data detection with data aggregation and confidential transmission in wireless sensor networks," *IEEE/ACM Trans. Netw.*, vol. 18, no. 3, pp. 736–749, Jun. 2010.

[22] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw.*, 2000, pp. 243–254.

[23] I. Stojmenovic, M. Russell, and B. Vukojevic, "Depth first search and location based localized routing and QoS routing in wireless networks," *Comput. Informat.*, vol. 21, no. 2, pp. 149–165, 2002.

[24] C. Liu and J. Wu, "Efficient geometric routing in three dimensional ad hoc networks," in *Proc. IEEE INFOCOM 28th Conf. Comput. Commun.*, Apr. 2009, pp. 2751–2755.

[25] S. Fortune, J. E. Goodman, and J. O'Rourke, Eds., "Voronoi diagrams and Delaunay triangulations," in *Handbook of Discrete and Computational Geometry*, 2nd ed. Boca Raton, FL, USA: CRC Press, 2004.

[26] R. Sarkar, X. Yin, J. Gao, F. Luo, and X. D. Gu, "Greedy routing with guaranteed delivery using Ricci flows," in *Proc. Int. Conf. Inf. Process. Sensor Netw.*, Apr. 2009, pp. 121–132.
[27] M. Shao, Y. Yang, S. Zhu, and G. Cao, "Towards statistically strong source anonymity for sensor networks," in *Proc. IEEE INFOCOM 27th Conf. Comput. Commun.*, Apr. 2008, pp. 51–55.
[28] R. Lu, X. Lin, H. Zhu, and X. Shen, "TESP2: Timed efficient source privacy preservation scheme for wireless sensor networks," in *Proc. IEEE Int. Conf. Commun.*, May 2010, pp. 1–6.
[29] K. Mehta, D. Liu, and M. Wright, "Protecting location privacy in sensor networks against a global eavesdropper," *IEEE Trans. Mobile Comput.*, vol. 11, no. 2, pp. 320–336, Feb. 2012.
[30] K. Ren, W. Lou, and Y. Zhang, "LEDS: Providing location-aware end-to-end data security in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 7, no. 5, pp. 585–598, May 2008.
[31] Y. Liu, J.-S. Fu, and Z. Zhang, "K-nearest neighbors tracking in wireless sensor networks with coverage holes," *Pers. Ubiquitous Comput.*, vol. 20, no. 3, pp. 431–446, Jun. 2016.
[32] L. Zhu *et al.*, "On stochastic analysis of greedy routing in vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 6, pp. 3353–3366, Dec. 2015.
[33] Z. Mohseni and M. Reshadi, "A deadlock-free routing algorithm for irregular 3D network-on-chips with wireless links," *J. Supercomput.*, vol. 74, no. 2, pp. 953–969, Feb. 2018.
[34] H. Kun, S. Haifeng, and L. Yonglei, "Integrating localization and energy-awareness: A novel geographic routing protocol for underwater wireless sensor networks," *Mobile Netw. Appl.*, vol. 23, no. 5, pp. 1–9, 2018.
[35] C. Chen, L. Liu, and T. Qiu, "ASGR: An artificial spider-Web based geographic routing in heterogeneous vehicular networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 5, pp. 1604–1620, May 2019.
[36] P. Bose and P. Morin, "Online routing in triangulations," *SIAM J. Comput.*, vol. 33, no. 4, pp. 937–951, Jan. 2004.
[37] D.-Y. Lee and S. S. Lam, "Protocol design for dynamic Delaunay triangulation," Dept. Comput. Sci., Univ. Texas Austin, Austin, TX, USA, Tech. Rep. TR-06-48, Dec. 2006.
[38] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 1994, pp. 359–370.
[39] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "Density-based spatial clustering of applications with noise," in *Proc. Int. Conf. Knowl. Discovery Data Mining*, 1996, pp. 6–13.
[40] T. M. Rath and R. Manmatha, "Word image matching using dynamic time warping," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2003, pp. II-521–II-527.
[41] K.-F. Ssu, C.-H. Ou, and H. C. Jiau, "Localization with mobile anchor points in wireless sensor networks," *IEEE Trans. Veh. Technol.*, vol. 54, no. 3, pp. 1187–1197, May 2005.
[42] J. J. Cho, Y. Ding, Y. Chen, and J. Tang, "Robust calibration for localization in clustered wireless sensor networks," *IEEE Trans. Autom. Sci. Eng.*, vol. 7, no. 1, pp. 81–95, Jan. 2010.
[43] Y. He, Y. Liu, X. Shen, L. Mo, and G. Dai, "Noninteractive localization of wireless camera sensors with mobile beacon," *IEEE Trans. Mobile Comput.*, vol. 12, no. 2, pp. 333–345, Feb. 2013.
[44] C. Intanagonwiwat, R. Govindan, D. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," *IEEE/ACM Trans. Netw.*, vol. 11, no. 1, pp. 2–16, Feb. 2003.
[45] Z. Zheng, A. Liu, L. X. Cai, Z. Chen, and X. Shen, "Energy and memory efficient clone detection in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 15, no. 5, pp. 1130–1143, May 2016.
[46] Y. Fan, Y. Jiang, H. Zhu, and X. Shen, "An efficient privacy-preserving scheme against traffic analysis attacks in network coding," in *Proc. IEEE INFOCOM - 28th Conf. Comput. Commun.*, Apr. 2009, pp. 2213–2221.
[47] W. Lou, W. Liu, and Y. Fang, "SPREAD: Enhancing data confidentiality in mobile ad hoc networks," in *Proc. INFOCOM*, vol. 4, Mar. 2004, pp. 2404–2413.
[48] A. Liu, Z. Zheng, C. Zhang, Z. Chen, and X. Shen, "Secure and energy-efficient disjoint multipath routing for WSNs," *IEEE Trans. Veh. Technol.*, vol. 61, no. 7, pp. 3255–3265, Sep. 2012.
[49] M. E. A. Mahmoud, X. Lin, and X. Shen, "Secure and reliable routing protocols for heterogeneous multihop wireless networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 4, pp. 1140–1153, Apr. 2015.
[50] M. D. Berg, *Computational Geometry: Algorithms and Applications*. Cham, Switzerland: Springer, 2000.
[51] L. Liu, Y. Liu, and N. Zhang, "A complex network approach to topology control problem in underwater acoustic sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 25, no. 12, pp. 3046–3055, Dec. 2014.
[52] L. Liu, R. Wang, and F. Xiao, "Topology control algorithm for underwater wireless sensor networks using GPS-free mobile sensor nodes," *J. Netw. Comput. Appl.*, vol. 35, no. 6, pp. 1953–1963, Nov. 2012.

Junsong Fu received the Ph.D. degree in communication and information systems from Beijing Jiaotong University in 2018. He is an Assistant Professor with the School of Cyberspace Security, Beijing University of Posts and Telecommunications. He is interested in network security and information privacy protection.

Na Wang received the Ph.D. degree from the School of Mathematical Sciences, Xiamen University, in 2018. She is a Post-Doctoral Fellow with the School of Computer Science, Beijing University of Posts and Telecommunications. Her research interests include cryptography, message sharing, and information security issues in distributed systems and cloud systems.

Leyao Nie received the bachelor's degree from the Beijing University of Posts and Telecommunications in 2017, where she is currently pursuing the undergraduate degree with the School of Cyberspace Security. Currently, her synthesizing grade ranks first among 90 competitors. She is passionately interested in scientific research fields, including information security and privacy, risk management, and the Internet of Things.

Baojiang Cui received the Ph.D. degree in control theory and control engineering from Naikai University, China, in 2004. He is currently a Professor with the School of Computer Science, Beijing University of Posts and Telecommunications, China. His main research interests include the detection of software, cloud computing, and the Internet of Things.

Bharat K. Bhargava (Life Fellow, IEEE) is a Professor of computer science with Purdue University. He has published more than 100 research papers. He is the Founder of the IEEE Symposium on Reliable and Distributed Systems, IEEE Conference on Digital Library, and the ACM Conference on Information and Knowledge Management. He has won five best paper awards in addition to the Technical Achievement Award and the Golden Core Award from IEEE. He is the Editor-in-Chief of four journals and serves on over ten editorial boards of international journals.