LSCD : A Low-Storage Clone Detection Protocol for Cyber-Physical Systems

LSCD: A Low-Storage Clone Detection Protocol for Cyber-Physical Systems

Mianxiong Dong, Kaoru Ota, Laurence T. Yang, Anfeng Liu, Minyi Guo

***Abstract*— Cyber-Physical Systems (CPSs) have recently become an important research field not only because of their important and varied application scenarios, including transportation systems, smart homes, surveillance systems, and wearable devices, but also because the fundamental infrastructure has yet to be well addressed. Wireless Sensor Networks (WSNs), as a type of supporting infrastructure, play an irreplaceable role in CPS design. Specifically, secure communication in WSNs is vital because information transferred in the networks can be easily stolen or replaced. Therefore, this paper presents a novel distributed *L*ow-*S*torage *C*lone *D*etection protocol (LSCD) for WSNs. We first design a detection route along the perpendicular direction of a witness path with witness nodes deployed in a ring path. This ensures that the detection route must encounter the witness path because the distance between any two detection routes must be smaller than the witness path length. In the LSCD protocol, clone detection is processed in a non-hotspot region where a large amount of energy remains, which can improve energy efficiency as well as network lifetime. Extensive simulations demonstrate that the lifetime, storage requirements, and detection probability of our protocol are substantially improved over competing solutions from the literature.**

***Index Terms*—wireless sensor network security, clone detection, network lifetime, distributed route**

I. Introduction

Wireless Sensor Networks (WSNs) are one of the most important compositions of Cyber-Physical Systems (CPSs) [1], [2], which monitor physical phenomena from surrounding environments to dynamically interact with human activities and/or machine systems such as surveillance systems, body health monitoring systems, and intelligent transportation systems. Secure communication in WSNs is vital because information transferred through such networks can be easily stolen or replaced [1]-[3]. For instance, an adversary could capture sensor nodes and acquire all the information stored therein—the nodes are commonly assumed to not be tamper proof. Therefore, an adversary may replicate captured nodes and deploy them in the network to perform a variety of malicious activities [2], [4]-[8]. This type of attack is referred to as a clone attack [4]-[6]. A cloned node, because it has legitimate information, may participate in network operations in the same manner as a non-compromised node, and thus, the cloned node can launch a variety of attacks [5]-[7]. It is not difficult to imagine that this may present a huge risk to users in many types of WSN applications (e.g., flood monitoring). Therefore, early detection and recognition of cloned nodes has important significance to network security [4], [7], [8].

Witness-based clone detection methods that allow resource-constrained sensor nodes to mitigate node capture and clone attacks have been developed [5]-[9]. In such methods, each node probabilistically forwards its identity (ID) to a set of coordinates that act as witness nodes. Such methods use the fact that clones have the same ID as the captured node but are at different locations. Hence, a clone is detected when two nodes report the same ID but different locations.

Recently, various distributed witness-based solutions for addressing this fundamental problem have been proposed. However, these solutions are not satisfactory due to the following three reasons. First, the storage capacities of sensor nodes are limited. Due to the very limited nodal storage capacity, protocols at the $O(\sqrt{n})$ storage level often cannot be used in practice [5], [9]. Second, the probability of clone detection is quite low. In distributed clone detection protocols, because witness and detection routes are distributed, ensuring that detection routes encounter witness nodes is challenging. Hence, in current research, the clone detection probability is often 60% [4], [6], [8], which cannot be applied to applications with higher security requirements [10], [11]. Third, the network lifetime is not considered. To achieve a higher clone detection probability in distributed networks, a system must suffer higher communication costs, which results in the consumption of the limited battery power of sensor nodes and decreased network lifetime [2], [4], [9].

In this paper, we propose a novel clone detection protocol, named the Low-Storage Clone Detecting protocol (LSCD). The most obvious feature is that the LSCD protocol is not affected by the network scale or number of nodes so that the storage requirement remains at a small constant level. Moreover, the LSCD protocol has a very high detection probability, high security performance, and improved energy efficiency. The main innovations of this paper are as follows:

(1) The LSCD has low storage requirements.

To the best of our knowledge, the LSCD protocol is the first clone detection protocol that achieves storage at small constant

Manuscript received 1 July 2015; revised ****. 2015

This work is supported by JSPS KAKENHI grant numbers 26730056 and 15K15976, the A3 Foresight Program, the National Natural Science Foundation of China (61379110, 61073104, 61261160502, 61272099), and the National Basic Research Program of China (973 Program) (2014CB046305, 2015CB352403), and the Scientific Innovation Act of STCSM (13511504200).

Mianxiong Dong and Kaoru Ota are with the Department of Information and Electronic Engineering, Muroran Institute of Technology, Japan. E-mail: {mx.dong, ota}@csse.muroran-it.ac.jp.

Laurence T. Yang is with the Department of Computer Science, St. Francis Xavier University, Canada. (e-mail: ltyang@stfx.ca).

Anfeng Liu is with the School of Information Science and Engineering, Central South University, Changsha, China. (e-mail: afengliu@mail.csu.edu.cn).

M. Guo is with the Department of Computer Science, Shanghai Jiao Tong University, China. (e-mail: guo-my@cs.sjtu.edu.cn).

Φ levels. There is a tradeoff between storage capacity and energy consumption, namely, more detection routes can ensure a higher clone **detection** probability with decreased number of witnesses. Meanwhile, we found that, due to the "energy hole" phenomenon in WSNs, the remaining energy is as high as 90% under the premature death of the network [12]-[15]. Therefore, the LSCD protocol fully utilizes the remaining energy to create as many detection routes as possible to reduce the storage requirements of the node and achieves a small constant storage requirement.

(2) The LSCD protocol has fully distributed characteristics and provides strong protection against attacks and a high detection probability.

In the LSCD protocol, witness nodes form route paths along circles, with a sink serving as the center, because clone detection is processed along the centrifugal (or centripetal) direction, and the distance between any two detection routes is shorter than the witness path length. Thus, the witness path must encounter the detection route, ensuring that the LSCD theoretically has a 100% clone detection probability. Moreover, witness routes and clone detection routes are randomly generated. Thus, even if the adversary knows the LSCD algorithm, the locations of witness nodes and detection route information cannot be obtained. Therefore, the LSCD protocol has fully distributed characteristics and strong robustness to compromise attacks.

(3) The LSCD protocol prolongs the network lifetime.

The LSCD protocol inactively performs clone detection in hotspots to reduce energy consumption. Meanwhile, in regions with abundant energy, clone detection is performed as aggressively as possible to ensure a higher detection probability, which effectively improves network lifetime.

The remainder of this paper is organized as follows: In section Ⅱ, related works are reviewed. The system model and the problem statement are described in section III. The novel LSCD protocol is presented in section IV. A security performance analysis is provided in section V. Section VI presents the experimental results and comparison. Finally, we conclude this paper in section VII.

II. Related Work

A clone attack proceeds as follows [2], [5], [6], [8], [9]. First, an adversary simply needs to physically capture one node from the network; obtain its security credentials, such as codes and cryptographic materials; and, if necessary, reprogram the node to modify its behavior. Then, with little effort, the adversary replicates the node to create an army of malicious nodes at its command. Finally, it deploys these replicas back into the WSN, possibly at strategic positions, to launch a variety of insidious insider attacks to undermine the network protocol [2], [9]. The goal of clone detection is to detect cloned nodes with high probability [2], [9]. Detection protocols can be divided into centric and distributed algorithms according to their applied control modes.

Centric protocols require all nodes to send their neighbors' location information to the base station for detection [16]. Other solutions rely on local detection, such as voting mechanisms, which use nodes within a neighborhood to confirm the legitimacy of a given node [15], [17]. A method called Area-Based Clustering Detection (ABCD) detects clones based on two or more different locations to avoid the shortcomings of single base station detection [18]. However, the setback of centric protocols is that if the adversary intercepts the base station message or interferes with base station communications, centric detection will fail. Moreover, nodes near the base station are required to forward substantially more packets than other nodes, which seriously decreases the network lifetime; thus, distributed protocols have seen higher preference [9].

Distributed detection protocols can be divided into the following categories according to the generation mode of witness nodes:

(1) Randomly generated witness mode. This protocol maps a node's ID to a random witness node set [9]. Because the witness is randomly generated, an adversary cannot determine the corresponding witness in advance even if it obtains the map function; therefore, it cannot compromise witnesses, thereby achieving better security. However, due to the randomness in witness generation, it is difficult to ensure that a message from the same node ID has the same witness; thus, the clone detection probability is low. Moreover, the communication and storage requirements are also relatively large. Research on this type of protocol includes Non-Deterministic and Fully Distributed (NDFD) protocols [5]. As discussed in [9], Randomized Multicast (RM) is a distributed algorithm for detecting node replications in which $\sqrt{n}$ witness nodes are randomly selected among $\boldsymbol{n}$ nodes in the network. When detecting clones, each node's neighbors probabilistically forward claims to a randomly selected set of witness nodes. The disadvantage of the RM protocol lies in the communication cost, which is $O(n^2)$, and its comparatively low detection probability. The Line-Selected Multicast (LSM) protocol in [9] decreases communication costs by increasing storage, and its core idea is as follows. Considering a nodal degree of d, node a randomly selects g nodes as its witness with probability p. Thus, there are gpd witnesses, and then, a's neighbor forwards a claim of a's ID and location to these gpd witnesses and stores the claim in the route. During clone detection, the node can select a certain number of nodes as the route destination and check whether the detection route will encounter a witness; thus, the detection probability is improved but at a cost of larger storage requirements of $O(n)$.

(2) Deterministic witness generation mode. This protocol deterministically maps the nodal ID to a set of witnesses. The RED protocol has received the most attention under this mode [2]. This protocol's core idea is that a set of witnesses can be computed by a map function pseudo rand (rand, ID_x, g). Under clone detection, detected information is sent to the witness. This protocol has high detection probability and low communication cost but suffers from a low robustness against compromise attacks. Once an adversary captures a node, it can obtain the map function and thus becomes aware of this witness set. Then, it can compromise this witness set, resulting in detection failure [10].

(3) Randomly deterministic witness generation mode. In this mode, a node obtains only the witness location region through the map function, and witness nodes remain unknown; therefore, it is difficult for the adversary to compromise all

witnesses of this region. Moreover, this mode represents a tradeoff between randomness and determinism, and thus, its storage requirements, communication costs and robustness to compromise attacks are between the other above two modes. Related research was proposed in [10]: (*Single Deterministic Cell* (**SDC**) and *Parallel Multiple Probabilistic Cells* (**P-MPC**). A replicated node detection approach called SET was proposed in [4].

In [19], we proposed an energy-efficient ring-based clone detection (ERCD) protocol that achieved a high detection probability with random witness selection while ensuring normal network operation and satisfactory WSN lifetime. In the ERCD protocol, a witness is selected in a ring path around the sink; however, the location of the ring is randomly generated. Therefore, the attacker cannot easily determine the location; this provides the system with high robustness to attack. Clone detection can meet the rings of witnesses by sending one concentric (centrifugal) route; therefore, the probability of clone detection is high. The disadvantage suffered by this scheme is that it is difficult to form a ring. However, the LSCD scheme only requires a small ring routing to store witnesses; thus, it not only achieves stronger practical performance but also reduces the storage requirements of the nodes.

For mobile node clone detection, Zhu *et al.* proposed a distributed solution called NBDS [17], which considers node mobility and allows for occasional movement. Related research can be found in [6], [20].

III. The System Model and Problem Statement

A. The System Model

Network model

(1) We consider n sensor nodes, uniformly and randomly scattered in a circular network; a nodal transmission radius r; a network radius $R = \hbar r$; and a nodal density ρ [3], [21]. Data are periodically collected, and nodes send sensed data to the sink via multi-hop during each data collection round [13], [21], [22].

(2) We assume that link-level security has been established through a conventional cryptography-based bootstrapping algorithm. We also assume that a link key is safe unless the adversary physically compromises either side of the link. We also assume that there exists a trusted and powerful sink that will never be compromised [23].

(3) The sensor nodes are assumed to know their relative locations, the sink node location and the hops to the sink. We also assume that each sensor node has knowledge of its adjacent ***neighboring*** nodes. The information about the relative location of the sensor ***domain*** may also be broadcast through this network to facilitate updating of the routing information [2], [9]. Thus, because of the lack of GPS requirements, the LSCD protocol lowers demands on the system.

The Adversary Model

An adversary can compromise a set of network nodes and extract their information. Using this extracted data, the adversary can then fabricate exact functional copies of captured nodes (clones) and deploy the clones back into the network [2], [9]. We consider that the adversary is powerful and can subvert a limited number of legitimate sensor nodes at unpredictable locations simultaneously [2].

We assume that the adversary operates in a stealthy manner, therein attempting to avoid detection [2], and the adversary cannot readily create new IDs for nodes [2]. Newsome et al. described several techniques for preventing an adversary from deploying nodes with arbitrary IDs [5]-[10].

B. Energy Consumption Model and Related Definitions

$$\begin{aligned} E_t &= lE_{elec} + l\varepsilon_{fs}d^2, \quad if\ \ d < d_0 \\ E_t &= lE_{elec} + l\varepsilon_{amp}d^4, \quad if\ \ d > d_0 \end{aligned} \tag{1}$$

Adopting a typical energy consumption model [12], [21], [24], [25], the energy consumption for transmission is given by Formula 1. Because the energy consumption for receiving is relatively low compared with that for transmission, in this paper, we consider only transmission energy consumption. E_{elec} represents the transmitting circuit loss. Both the free space (d^2 power loss) and the multi-path fading (d^4 power loss) channel models are used in the model, depending on the distance between the transmitter and receiver. ε_{fs} and ε_{amp} are the amounts of energy required to perform power amplification in the two models. l denotes the number of data bits. The problem statement is as follows:

The optimization goal of this paper is as follows:

(1) Network lifetime maximization. As in [3], [12], the network lifetime is defined as the time to the death of the first node. We consider e_i^1 as the energy consumption of node i for regular data collection, and e_i^2 is that for clone detection. Then, the maximization of the network lifetime can be expressed as follows:

$$max(T) = min \max_{0<i\le n}(e_i^1 + e_i^2) \tag{2}$$

(2) High clone detection probability. For any node a, if an adversary clones nodes such as a', a''…., and if the detection probability for node a can be expressed as $P_d(a)$, then the maximization of $P_d(a)$ is as follows:

$$\max(\mathrm{P}) = \max(P_d(a)) \tag{3}$$

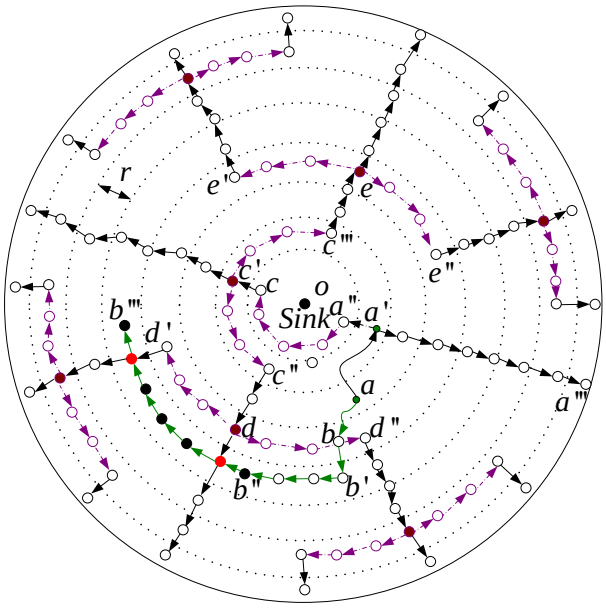

Fig. 1 Illustrate of the low storage clone detecting protocol.

(3) Smaller storage requirement. Generally, the nodal storage capacity is relatively small, often on the order of a few kilobytes [8]. Therefore, the third optimization goal of our protocol is to minimize the storage requirement, namely,

$$\min(\mathrm{M})\,|\mathrm{M} = \min((\sum_{i=1}^{n} n_i.m_e)/\mathrm{n}) \wedge (\forall n_i m_e \le \tau) \qquad (4)$$

$n_i.m_e$ denotes the required clone detection storage capacity for node n_i, and τ denotes the upper bound of the allowed storage.

In summary, the optimization goal of this paper is

$$\begin{cases} max(T) = min \max_{0<i\le n}(e_i^1 + e_i^2) \\ \max(\mathrm{P}) = \max(P_d(\varepsilon)) \\ \min(\mathrm{M})\,|\mathrm{M} = \min((\sum_{i=1}^{n} n_i.m_e)/\mathrm{n}) \wedge (\forall n_i.m_e \le \tau) \end{cases} \qquad (5)$$

The optimization also includes the distributed characteristics, robustness against compromise attacks and system demands, which will be discussed in the performance analysis section.

To allow the readers to better understand this paper, the main notations introduced in this paper are listed in Table I.

TABLE I
NOTATIONS

R	The network radius
r	The transmission range of a node
Ψ	The length of a witness path
$\hbar$	The max hop distance from the Sink
h	The current node distance from the Sink in hops
n	The number of nodes in the network
ID_a	The identity information of node a
l_a	The location node a claims to occupy
X_a	The message including node a's private information
λ	The data collection frequency
λ'	The clone detection frequency
λ''	The witness construction frequency
σ	The ratio of data packet length over detection packet length
d	Average degree of each node
p	The probability a neighbor will forward location information
g	The number of forwarding nodes selected by each neighbor
$\varepsilon_2, \varepsilon_2, \varepsilon_3$	The random walk route hop for clone detection and constructing witness paths

IV. LSCD DESIGN

A. Overview of the Proposed LSCD Protocol

As shown in Fig. 1, the LSCD protocol consists of two components: the witness building stage and the clone detection stage.

(1) The witness building stage can be divided into two stages. Stage 1: Let $\mathcal{F}(ID, location,\ h)$ be the map function, which can set the node ID and location map to an arbitrary integer among $[2, \hbar\,]$; however, for the same ID and location, the calculated results are not always the same when using $\mathcal{F}(ID, location,\ h)$. An example of such a map function can be easily found in [2], [19]. For node a, k= $\mathcal{F}(a.ID, a.location,\ h)$ is the hop count between the node a and the Sink (see Fig. 1), that is, the hop count from the witness of node a to the Sink is k. Then, node a must route his ID and $location$ to the witness nodes, which are k hops away from the Sink. However, to confuse the adversary, in the LSCD protocol, node a does not directly route data to the nodes, which are k hops away from the Sink; rather, the following methods are used to enhance the security of the protocol. After node a is randomly routed a given distance using the random routing method, node a routes data to the nodes that are k hops away from the Sink using a directional routing method and then form an arc path around the sink using the jumping routing method. The specific process is as follows: node a randomly routes c =rand($\mathcal{J}$) hops to node b carrying its (ID, location), and then, b compares its hop count to the sink ($b.hop$) with k to make a decision as to whether to route centrifugally or centripetally. This process is repeated until reaching node b', which is k hops to the sink. Then, the first stage ends. In this stage, each node in the routing path does not need to store the ID and $location$ of node a.

Stage 2: The main task of stage 2 is to form a continuous length Ψ to the nodes that are k hops away from the Sink. Similarly, to confuse the adversary, a routing path that cannot store the witness is formed at the beginning of the second stage; however, the true witness is stored in the node passed by in the last routing of the second stage. The process is as follows: node b' randomly chooses the left hand (or right hand) direction for the same-hop routing (i.e., each node selects the next-hop nodes that are the same hop counts to the Sink) until node b' routes to node b''. Then, node b' stores (ID, location) of node a on every node in the later route path until the route path length is set as a length Ψ, namely, $\overline{b'b''} = \Psi$. After this stage, each node's witness must be in the arc centered on the sink, which is k hops to the sink and has length Ψ. Because k and the arc are randomly generated, the adversary cannot calculate their value even though it captures the random function. Therefore, it cannot compromise these witnesses in advance, thus achieving high security.

(2) The clone detection stage: In the LSCD protocol, the main innovation is that the witness of each node is stored in a route whose length is Ψ. Considering that the transmission radius of the node is r, the number of storing witnesses of each node is $\lceil \Psi/r \rceil$, that is, the required storage space for storing witnesses in the LSCD protocol is unrelated to the number of nodes and network size: it is a constant. To the best of our knowledge, in most previous studies, the storage space required for storing the witnesses of nodes is related to the number of nodes n [2], [5], [19], and the storage space for storing the witnesses of nodes in few studies is related to the network size $\hbar$. Therefore, it is difficult to research the storage space requirements and costs of the nodes under large network scenarios. However, the LSCD protocol can overcome those shortcomings and thus is very suitable considering the minimal storage space provided by the nodes in WSNs. The LSCD protocol produces a constant storage space of the node because (1) the witnesses of nodes are along an arc route whose length is Ψ, and the center is the sink. Therefore, when performing clone detection, if the system creates many centrifugal clone detection routing paths departing from the sink, and if the distance among any two clone detection routes is less than Ψ, there will be a clone detection route and witness path that

intersect, which can achieve the goal of clone detection. Note that the witness of the nodes is in the arc route whose length is Ψ; however, the location of the arc route is arbitrary. Therefore, theoretically, the adversary does not obtain the position of where the witnesses are stored. (2) Moreover, the above shows that the distance between any two clone detections is less than Ψ in the LSCD protocol; therefore, the number of clone detection routes is related to the length Ψ of the stored witness. The relationship is as follows: if the length Ψ increases, the number of required clone detection routes decreases, whereas if Ψ decreases, the number of required clone detection routes increases. In addition, a higher Ψ results in greater storage space requirements for the nodes, and the number of clone detection routes is linearly proportional to the energy consumption of the network. This represents a tradeoff between the required storage space of nodes and the network lifetime. For example, if Ψ is a ring with the Sink as the center, only a clone detection routing can achieve the goal of clone detection. At this time, the energy consumption of the nodes required by clone detection is minimized; thus, the network lifetime is maximized, and the maximum length of the routing path of the witness is $2\pi\hbar r$. Therefore, the storage space requirement for storing the witness is not constant; it is related to the network scale. Based on the above analysis, the LSCD protocol adopts a different scheme compared to previous clone detection methods to ensure that the required storage space of the nodes is a small constant and that the network lifetime is equal to the network lifetime with only one clone detection route. The length of the required ring in the area near the Sink is small; therefore, the number of required clone detection routes in the area near the sink is small. However, in the area far from the sink, the number of clone detection routes increases. Thus, the LSCD protocol creates few clone detection routes in areas near the Sink and creates numerous clone detection routes in areas far from the Sink. This can ensure that the distance between any two clone detection routes is less than Ψ. Moreover, note that, in WSNs, because the nodes near the Sink must forward the data of nodes far from the Sink, the energy consumption is much higher than that in the area far from the Sink. Thus, an "energy hole" is formed, resulting in premature network death. According to previous studies, more than 90% of the energy in the network cannot be used; therefore, the LSCD protocol makes fully utilizes the energy resources of WSNs. The generated clone detection routes in different areas can reduce the energy consumption in the hotspot region and fully utilize the energy in the area far from the Sink to create numerous clone detection routes. This can ensure network security and maximize network lifetime and network energy utilization. The method for establishing clone detection routing is discussed in the following.

Clone detection is performed as shown in Fig. 2. In Fig. 2, the witness path is randomly distributed across the network, and the length of the witness path of each node is Ψ. The purpose of clone detection is that each witness path can meet at least one clone detection route. For security reasons, if the clone detection routing is always the same routing path, the adversary can intercept the detection route, thus invalidating clone detection, which is often overlooked in research on the original clone detection protocol. Therefore, in the LSCD protocol, for any node a, the node first randomly walks and routes to node a'. The location of node a' is not predetermined, and the routing sponsor node also does not know the location; therefore, the adversary cannot cut its route. Then, node a' routes to node a'', which is located in the 2nd ring (i.e., it is 2 hops to the sink) along the centripetal direction. In addition, node a' routes to node a''', who is $\hbar$ hops to the Sink along the centrifugal direction. Then, path a''' becomes the first clone detection routing. The clone detection routing is composed of the following process:

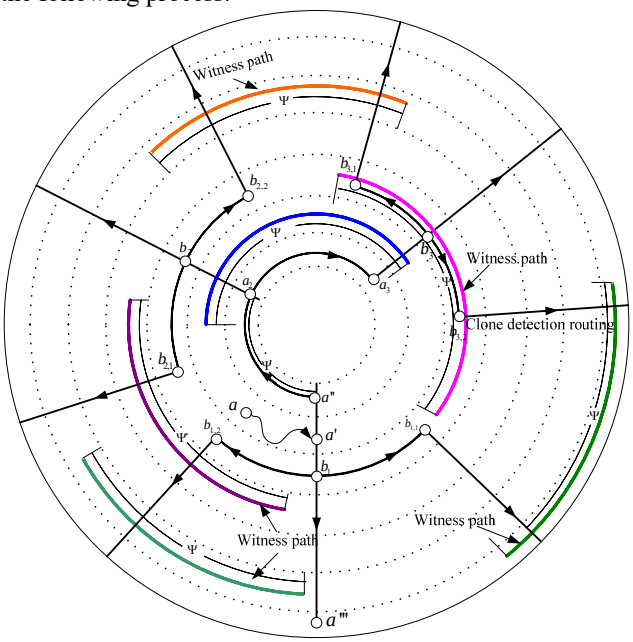

Fig. 2: The clone detection process in the LSCD protocol.

(a) The first same-hop routing. Considering that node a'' starts performing clone detection, node a'' routes to node a_2, which must add the detection centrifugal routing path using a same-hop routing approach. The length of $\overline{a''a_2}$ =ξΨ, where ξ is a number in the range (0,1). Because node a'' knows that the hop count to Sink is k, the number of generated centrifugal routing paths under same-hop routing can be obtained: $\mathcal{L}_k = \lceil (2\pi kr)/(\xi\Psi) \rceil$. However, node a'' creates a centrifugal routing path when the length under same-hop routing is $\lceil (2\pi kr)/\mathcal{L}_k \rceil$ until $\mathcal{L}_k$ centrifugal routing paths are created. This will stop some-hop routing. (b) Centrifugal routing. Centrifugal routing finds the node whose hop count is larger than that of the next node until reaching the network boundary. (c) Same-hop routing. When the centrifugal routing route is outward one hop, this process determines whether to create a new routing. The condition of the judgment is based on the current node b_2 (for example), and its hop count to the Sink is $\hbar$ if $\lceil (2\pi\hbar r)/\mathcal{L}_k \rceil \geq \xi\Psi$, which illustrates the creation of a new centrifugal routing. At this time, node b_2 initiates same-hop routing to the left and right. The routing length is calculated as $\lceil (2\pi\hbar r)/(3\mathcal{L}_k) \rceil$, and then, centrifugal routing is performed at the end of routing. The principle for creating a new route is as follows: (I) The routing node that does not initiate the new routing path will determine whether to create a detection route during the ring routing process. (Ⅱ) The new route is created

on both sides; therefore, the number of detection routes doubles after every construction. (Ⅲ) Fewer detection routes are generated in regions near the sink and more routes are generated in regions away from the sink to fully utilize the remaining energy and meet the requirement that the distance between any two detection routes is shorter than the witness route length. Finally, the generation of detection routes in the LSCD protocol is completely distributed, and the nodes will not be attacked before routing. Thus, this protocol provides strong robustness against attacks. The process continues and eventually forms the clone detection route, as shown in Fig. 2.

B. LSCD Protocol

The LSCD protocol includes the witness building stage and the clone detection stage, as detailed in **Algorithm 1**.

V. PERFORMANCE ANALYSIS AND OPTIMIZATION

A. Energy consumption and efficiency optimization

The core idea of the LSCD protocol is to fully utilize the remaining energy in regions away from the sink to construct as many clone detection routes as possible to reduce witness routes and thus achieve "energy for storage". The following analyzes the energy consumption and the calculation of the LSCD storage requirements.

Theorem 1: Considering $R = \hbar r$, where the nodal transmission radius is *r*, the clone detection data load of the node at ring *i* is $b_i = 1 + (\hbar^2 - i^2)/(2i - 1)$.

Proof: Because the nodal data of ring $\geq$ i must be forwarded by nodes along ring *i*, the nodal data load at ring *i* is

$$B_i = \pi\hbar^2 r^2\rho - \pi(i-1)^2 r^2\rho = \pi r^2\rho(\hbar^2 - i^2 + 2i - 1)$$

The number of data packets for each node is

$$b^2 = (\pi r^2\rho(\hbar^2 - i^2 + 2i - 1))/(\pi(2i-1)r^2\rho)$$

This simplifies and proves the method. ■

When *i*=1, the maximum nodal data load is $b_{max} = \hbar^2$. Considering that the energy consumption for a unit packet transmission is e_u, the nodal remaining energy at ring *i* is

$$\gamma_i = (b_{max} - b_i)e_u = (\hbar^2 - 1 - (\hbar^2 - i^2)/(2i-1))e_u \quad (6)$$

Theorem 2: If the number of detection routes at ring *i* is c_i, then the number of clone detection packets forwarded by nodes at ring *i* is $\zeta_i = (\hbar^2\varepsilon c_i)/(2i-1)$.

Proof: There are $\pi\hbar^2 r^2\rho$ nodes in the network. Each node will process c_i detection routes at ring *i*, and the number of transmission hops for each detection route is εc_i, where ε is a constant greater than 1. Therefore, the number of hops that must be forwarded is $\pi\hbar^2 r^2\rho\varepsilon c_i$. Then, for each node, we find $(\pi\hbar^2 r^2\rho\varepsilon c_i)/(\pi(2i-1)r^2\rho) = (\hbar^2\varepsilon c_i)/(2i-1)$ ■

In the LSCD protocol, except for detection routes, there are routes in the circumferential direction. Thus, the following gives the energy consumption for the clone detection of this part.

Theorem 3: In the LSCD protocol, the length of any node's clone detection route in the circumferential direction is $2\pi\hbar r$, and the average number of packets forwarded is $\zeta = 2\pi\hbar^3/(\hbar^2 - 1)$. **Proof**: The number of detection routes in the outermost layer is $\mu = (2\pi\hbar r)/\Psi$. The length of any node's detection route in the circumferential direction is Ψ; then, the total length is $\mu\Psi = 2\pi\hbar r$, and the number of times the node sends detection information is $2\pi\hbar$. There are $\pi\hbar^2 r^2\rho$ nodes in

Algorithm 1 Low-Storage Clone Detecting protocol

Initialize: The hops of each sensor node to the sink are built through the flooding protocol [17];
Exchange the relational information with neighbors;

Stage A: building witness

$X_a \leftarrow \text{Encrypt}(ID_a, l_a)$
k = PseudoRand(ID_a, l_a, h)
Random walk ε_1 hops to node *b*, i ← b.hop, b'=b;
while i≠ k **do**
  if i < k **then**
    b′ ← NextNodeOnMaxHop(b′), i ← i + 1;
  else
    b′ ← NextNodeOnLeastHop(b′), i ← i − 1;
  end if;
end while;
node b' Random walk ε_2 hops to node b″, where each node's hop count is the same as the route path; i ← 1
while i < ⌈Ψ/*r*⌉ **do**
  Let b″record X_a;
  b″ ← NextNodeOnSameHop(b″), i ← i + 1;
end while;

Stage B: clone detection

Random walk ε_3 hops to node a′; a′.tag = true
node a′ routing reverse sink to a‴ with broadcast X_a;
while a′.hop ≠ 2 **do**
  a′ ← NextNodeOnLeastHop(a′); Broadcast X_a;
end while;
∂ ←The hops need for routing to build the next clone route
a′.tag = false
routing ∂ hops to node *c* with same-hop routing;
c.tag = true, *c* route reverse to sink;
for each clone detection route reverse to the sink;
c′ ← NextNodeOnMaxHop(c′); Broadcast X_a;
  if c′.tag = true **then**
    compute ∂ using formula 9;
    if $\partial \neq 0$ **then**
      along both left- and right-hand directions, same-hop routing ∂hop to nodes c″, c‴;
      c′.tag = false;
      nodes c″, c‴ route reverse to the sink;
    end if;
  end if;
end for;
for each node *S* that hears X_a **do**
  if (ID_a, l_a) of $S \neq (ID_a, l_a)$ in X_a **then**
    trigger the revocation procedure;
  end if
end for

the network, and as such, the total number of times is $\pi\hbar^2r^2\rho 2\pi\hbar$, except that the first ring has no detection routes. The average number of times nodes forward detection information for other regions is $\pi\hbar^2r^2\rho 2\pi\hbar/(\pi\hbar^2r^2\rho - \pi r^2\rho)$

■

Theorem 4: Considering that the witness construction cycle is λ'' data collection cycles, the number of forwarded packets for witness construction is

$$\vartheta = \lambda''\left(\varepsilon_2 + \frac{(\hbar-1)}{2} + \varepsilon_3 + \frac{\Psi}{\mathrm{r}}\right)\hbar^2/(\hbar^2-1) \quad (7)$$

Proof: The nodal walk route hop count is ε_2, and the average hop to the witness ring is $(\hbar-1)/2$. The walk hop count along the ring is ε_3, R route hops along the ring is Ψ/r. Therefore, the hop count for the witness construction phase for each node is $\varepsilon_2 + (\hbar-1)/2 + \varepsilon_3 + \Psi/r$. Because no witness is constructed in the first ring, the average number of forwarded packets for each node is $\lambda''(\varepsilon_2 + (\hbar-1)/2 + \varepsilon_3 + \Psi/\mathrm{r})\hbar^2/(\hbar^2-1)$.

Inference 1: For nodes at ring *i*, the number of detection routes that can be created without decreasing the network lifetime is

$$c_i = \frac{\sigma(\hbar^2 - 1 - \frac{(\hbar^2-i^2)}{(2i-1)} - \frac{\varsigma\lambda'}{\sigma} - \frac{\vartheta\lambda''}{\sigma})(2i-1)}{(\hbar^2\varepsilon)\lambda'}$$

Proof: Considering that the number of detection routes is c_i, the detection packet length is $1/\sigma$ of the data packet length, and the clone detection frequency is λ' data collection cycles. According to Theorem 2, the detection energy consumption in the radial direction is $\zeta_i e_u\lambda'/\sigma$. The remaining energy at ring *i* is γ_i, the energy consumption in the circumferential direction is $\varsigma e_u\lambda'/\sigma$, and the witness construction energy consumption is $\vartheta e_u\lambda''/\sigma$. Then, the remaining energy for detection is $\gamma_i - \varsigma e_u\lambda'/\sigma - \vartheta e_u\lambda''/\sigma$, and combining with the above, this aspect is proved.

■

Theorem 5: In the LSCD protocol, considering that the witness route length is Ψ, for a network with radius $\mathrm{R} = \hbar\mathrm{r}$, the number of required detection routes is $\mathrm{K}_i = 2^j | 2^j > \left\lceil\frac{2\pi ir}{\Psi}\right\rceil$

Proof: To ensure that a detection route encounters a witness, if the distance between any two detection routes is shorter than Ψ, the number of detection routes must be at least $c'_i = \lceil 2\pi ir/\Psi\rceil$. In the LSCD protocol, there are 2^j detection routes at ring *I*; then, *j* should be the minimum value that ensures 2^j is greater than c'_i, and this 2^j is what we desire.

■

Theorem 6: In the LSCD protocol, when the witness route length Ψ meets the following criterion, the detection process only uses the remaining energy.

$$c_i \geq 2^j | \forall i \in \{2 \ldots . \hbar\}, 2^j > \left\lceil\frac{2\pi ir}{\Psi}\right\rceil \quad (8)$$

Proof: According to Inference 1, c_i detection routes can be created at ring *i*. According to Theorem 5, when the witness route length is Ψ, K_i routes must be created at ring *i*. Thus, if Ψ can ensure $\mathrm{K}_i \geq c_i$ for any ring (see Formula 8), all detection routes can be created using remaining energy without affecting network lifetime.

■

Inference 2: Considering that node a in the detection route stores 2^j current routes, when a routes to ring *i*, the condition for new detection route construction and same-hop routing for these new routes is as follows:

$$\begin{cases} \partial = 0, & \text{if } 2^j \geq (2\pi ir)/\Psi \\ \partial = (2\pi\mathrm{ir})/(2^{j+1}), & \text{else} \end{cases} \quad (9)$$

Proof: Obviously, if $2^j \geq (2\pi ir)/\Psi$, then the distance of any two detection routes is smaller than Ψ. Therefore, no additional detection routes are needed, and $\partial = 0$; otherwise, additional detection routes are needed. According to the LSCD protocol, there are 2^j routes. With the distance of any two routes as $(2\pi\mathrm{ir})/2^j$, the newly created routes will be placed in the middle of the original routes, and the number of routes is doubled. Therefore, the length of the same ring route is $\partial = (2\pi\mathrm{ir})/(2^{j+1})$.

■

Theorem 7: In the LSCD protocol, considering Theorem 6, the lifetime ratio of the LSCD protocol to that of the RED (or LSM) protocol is

$$\varphi = \frac{h^2 + gpd\sqrt{(d+1)}h\lambda''}{(h^2 + 1 + \lambda'')} \quad (10)$$

Proof: Assume that $\lambda' = \lambda'' = 1$. According to [2], [8], it has been proven that the number of clone detection packets under the RED (or LSM) protocol is $gpd\sqrt{n}$ because the nodal degree is *d*; then, $\pi r^2\rho = d+1$, and the total number of nodes in the network is $\mathrm{n} = \pi(hr)^2\rho$. Because $\pi(hr)^2\rho/\pi r^2\rho = h^2$, $\mathrm{n} = h^2(d+1)$. Thus, $gpd\sqrt{n} = gpd\sqrt{(d+1)}h$. There are λ'' clone detections in each data collection round, and thus, there are $gpd\sqrt{(d+1)}h\lambda''$ clone detection packets because the amount of data in the first ring is maximized as h^2. Therefore, the maximum load of the RED and LSM protocols is $h^2 + gpd\sqrt{(d+1)}h\lambda''$.

In the LSCD protocol, the maximum load at the first ring is $h^2 + 1 + \lambda''$, among which 1 is the witness route construction load and λ'' is the clone detection load. Thus, the theorem is proved.

■

B. Storage Overhead

Theorem 8: The average nodal storage requirement is

$$\Phi = (\lceil\psi/r\rceil\hbar^2)/(\hbar^2-1) \quad (11)$$

Proof: In the LSCD protocol, the stored route length for each nodal witness is ψ, and it is stored $\lceil\psi/r\rceil$ times. There are $\mathrm{n} = (\pi(hr)^2\rho)$ nodes in total, and thus, the total storage is $\mathrm{n}\lceil\psi/r\rceil$ because the first ring generates the witness. These witness storage requires are undertaken by $\mathrm{n} - \pi r^2\rho$ nodes. Therefore, the storage needed by each node is $(\pi\hbar^2r^2\rho\lceil\psi/r\rceil)/(\pi\hbar^2r^2\rho - \pi r^2\rho)$.

■

C. Clone Detection probability

Theorem 9: Given that the selected witnesses of node *a* are trustful, if there exists a clone of node *a'*, the cloned node can always be detected.

Proof: As observed from the LSCD protocol, the witness of node *a* must be stored in an arc with length Ψ, and the distance between any two detection routes must be smaller than Ψ. Thus,

during clone detection, the detection route that contains node a's (ID, location) must encounter the witness of node a, and this reveals to the witness that nodes a and a' have the same ID but are at different locations. Thus, the cloned node can always be detected.

■

VI. Performance Analysis and Experimental Results

OMNET++ is used for experimental verification [26]. If not specified, the parameters are as follows: R=600 m, r=50 m, the node number is 1000, $\lambda' = \lambda'' = 1$, $\Psi = 7r$, and $\sigma = 5$. The energy consumption parameter settings can be observed in [12], [21] and can be found in Table II.

TABLE II
Network Parameter

Parameter	Value
Threshold distance (d_0) (m)	87
Sensing range r_s (m)	15
E_{elec}(nJ/bit)	50
e_{fs} (pJ/bit/m^2)	10
e_{amp} (pJ/bit/m^4)	0.0013
Initial energy (J)	0.5

In the experiment, the frequency for generating the data of the node is λ, and the generated data are routed to the Sink through the shortest-routing approach. The clone detection frequency of each node is λ', and the witness construction frequency is λ''. Data generation, clone detection, and witness construction are performed periodically according to the frequency period. The energy consumption of the nodes and the storage conditions are observed, and a certain proportion of nodes are cloned to check whether the LSCD protocol can check cloned nodes.

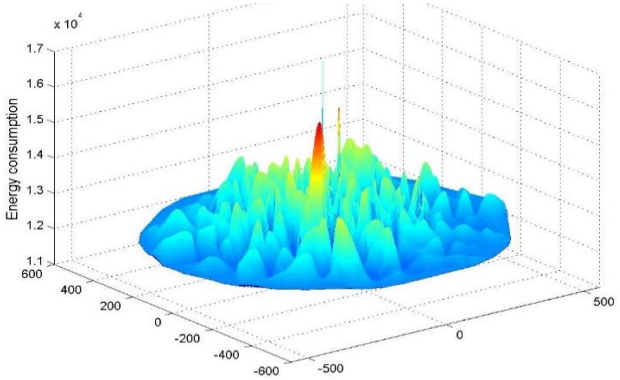

Fig. 3: The energy consumption of the LSCD protocol.

Fig. 3 shows the 3D energy consumption map of the LSCD protocol. In the clone detection protocol of [2], [9], the data collection results in not only a greater energy consumption of the area near the Sink but also the probability of the clone detection routing through the network center being higher than in other regions. This leads to the greater energy consumption of the area near the Sink, which reduces the network lifetime. However, the LSCD protocol is not the same as the previous protocol, which avoids the hotspot area, increasing the energy consumption in areas far from the Sink. In addition, as observed, although the energy consumption in the non-hotspot region is increased in the LSCD protocol, it remains smaller than in the hotspot region; therefore, the network lifetime is not affected by clone detection.

A. Experimental results for energy consumption and lifetime

Fig. 4 shows the energy consumption comparison between the LSCD and LSM protocols. The figure shows that the energy consumption for data collection is substantial. In the LSCD protocol, the detection and witness construction energy consumptions in hotspot regions are minor, thus having minimal effect on the total energy consumption. In the LSM protocol, the detection energy consumption in hotspots is quite large compared to that under the LSCD protocol; therefore, the network lifetime is reduced. Moreover, under the LSCD protocol, the remaining energy in non-hotspot regions can fully meet the conditions in Theorem 6.

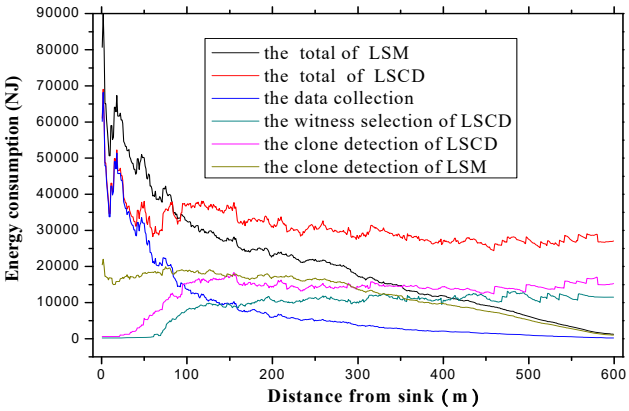

Fig. 4 Energy consumption under different distances from the sink.

Fig. 5 shows that, in one data collection round, the LSCD protocol improved the network lifetime by 30% to 90% compared to the LSM protocol with increased clone detection frequency. Fig. 6 shows the LSCD protocol's improved lifetime of 27% to 45% compared to LSM with increased network scale ($\hbar$). Fig. 7 shows that the energy consumption under the LSCD protocol is not related to the nodal degree, and the network lifetime is essentially unchanged. In contrast, under the LSM protocol, as the nodal degree increases, the communication cost rapidly increases; therefore, the lifetime quickly decreases for larger nodal degrees. The cost is only 1/3 of that of the LSCD protocol; thus, the LSCD protocol provides good lifetime performance.

B. Storage Overhead

Fig. 8 and Fig. 9 show the storage requirements under different nodal degrees and network scales. As observed, the LSCD protocol requires minimal storage, only 1/5 to 1/2 of that of LSM. The results show the following: the storage space requirements can increase with increasing number of nodes (see Fig. 8) or network size (see Fig. 9). In addition, the

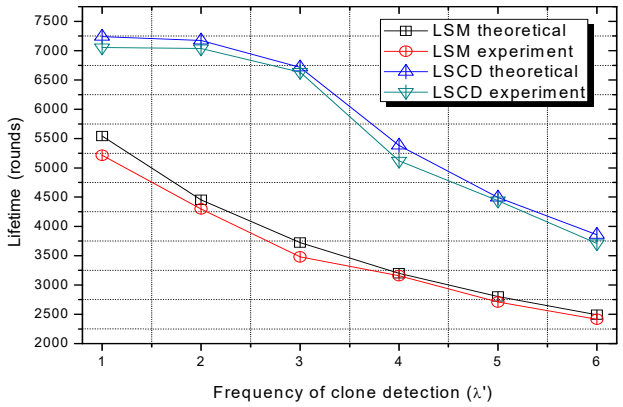

Fig. 5 Lifetime under different detection frequencies.

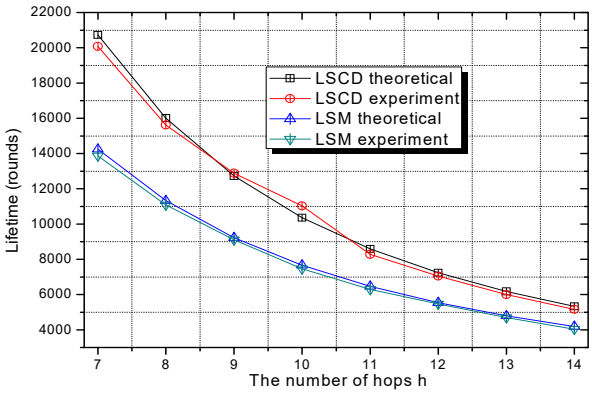

Fig. 6 Lifetime under different network scales.

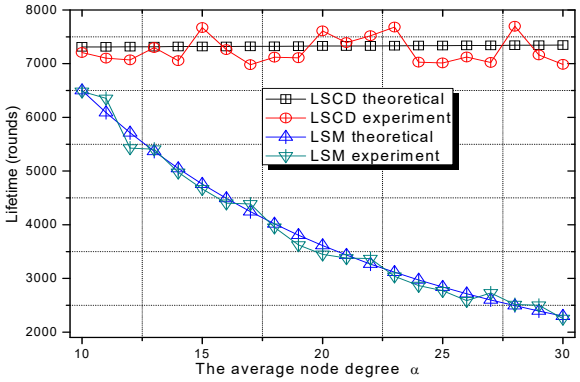

Fig. 7 Lifetime under different nodal degrees.

required storage space of the LSCD protocol is constant. This is because, in the LSCD protocol, the storage length of the witness of each node is fixed as Ψ; therefore, it is only a small constant and does not change with changing nodal degree or network scale. Thus, it is particularly suitable for large-scale networks. Because the LSCD protocol consumes substantially more energy in non-hotspot areas, the storage space is reduced. This result can be obtained by increasing the clone detection route length in the non-hotspot area, namely, so-called "energy for storage". In general, a smaller Ψ results in a reduced storage space requirement, and larger required routing paths when the protocol performs clone detection requires substantial energy in non-hotspot areas. Therefore, if there are no time limitations on clone detection, the general recommendation is to fully utilize non-hotspot energy and minimize Ψ.

C. *Detection probability*

Fig. 10 and Fig. 11 show the clone detection probability under different nodal degrees *d* and *h*, respectively. We can conclude the following. First, the clone detection probability under the LSCD protocol is higher, approximately 90%, whereas it is only 60% under LSM. Second, the clone detection probability under the LSCD protocol is not related to the network scale or nodal degree and ensures a high probability of success. In contrast, under LSM, this probability will decrease as *d* increases because clone detection under LSM is processed using at least two routes that intersect at the same node. Therefore, with increasing *d*, more nodes can be chosen as the next hop. Hence, the probability of two routes intersecting at the same node is decreased.

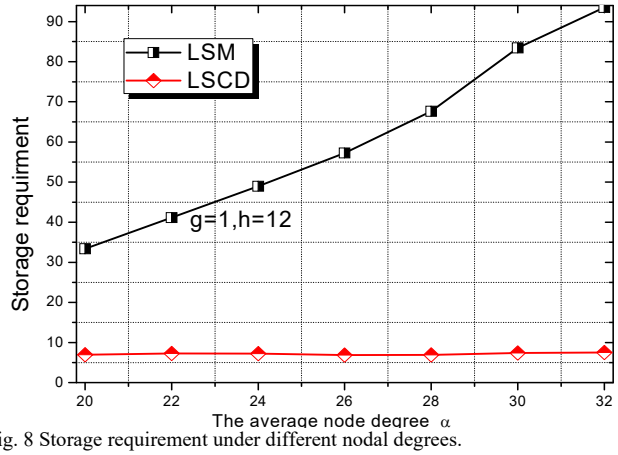

Fig. 8 Storage requirement under different nodal degrees.

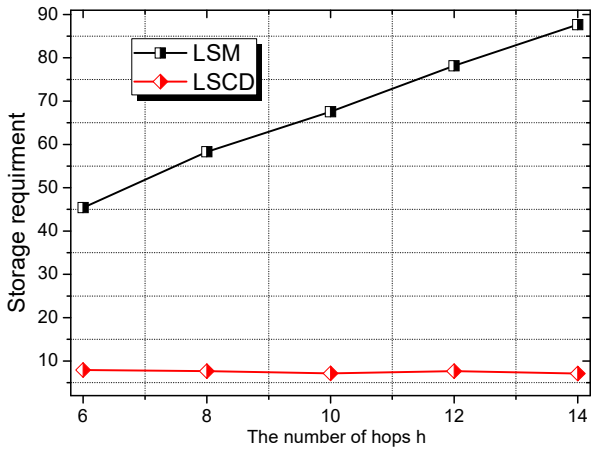

Fig. 9 Storage requirements under different network scales.

Table III displays the performance comparison between the LSCD protocol and several typical clone detection protocols, where the symbol ϖ denotes the number of witness nodes that store the local claim of a cell and *q* denotes the cloned number for one node in RED. The LSCD protocol is found to be superior to the other protocols in terms of all measures,

including lifetime, storage, detection probability, and robustness against adversaries' attacks. Table III shows that, although the SDC and LSCD protocols have high clone detection probabilities, SDC's capacity to resist compromise attacks is far less than that of the LSCD protocol for the following reasons: In the SDC protocol, the witness is stored in one or several nodes. When performing clone detection, the nodes compare with the witness by sending detection information to this area and broadcasting information. Therefore, in the SDC protocol, the adversary, once they compromise a node, can obtain the map function of the witness; thus, they know the area in which the witness is stored. Then, the adversary adopts some methods to prevent detection or further damages the nodes in this area to make the SDC protocol expire. Thus, its robustness against compromise is medium. In the LSCD protocol, the locations of each witness are not the same. Even if the adversary compromises a node, it still cannot obtain the storage location of the witness, thus making it unable to be attacked. Therefore, its robustness against compromise is high.

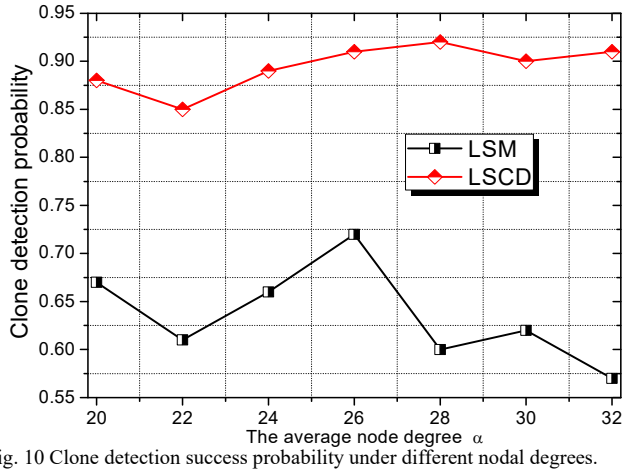

Fig. 10 Clone detection success probability under different nodal degrees.

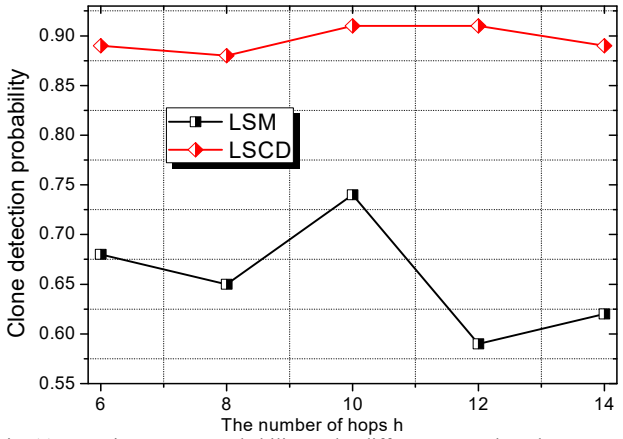

Fig. 11 Detection success probability under different network scales.

We turn to the analysis of the time overhead of the LSCD protocol. The time overhead is defined as the hops required to perform clone detection or to construct the witness path. The time overhead is $\varepsilon_2 + (\hbar - 1)/2 + \varepsilon_3 + \Psi/r$ for constructing the witness path in the LSCD protocol, and the time overhead is $\hbar$ in the other protocols (i.e., LSM, RED, and SDC). The $\varepsilon_2 + \varepsilon_3 + \Psi/r$ and $\hbar$ are the same order of magnitude; thus, the time overhead for constructing those clone protocols is equal. The maximum time overhead is $\pi\hbar r + \hbar$ for clone detection in the LSCD protocol, and the time overhead is also $\hbar$ in the other protocols (i.e., LSM, RED, and SDC). Because the LSCD protocol is well designed to use the energy of the area far from the Sink to perform clone detection, the clone detection routing path in the LSCD protocol is longer than that in other protocols. Thus, the overhead time of the LSCD protocol is higher than those of the other protocols.

TABLE III
PERFORMANCE COMPARISON OF SEVERAL CLONE DETECTION PROTOCOLS

	Lifetime	Memory occupancy	Detecting probability	Robustness against compromise
LSM	low	$O(\mathrm{g.p.d}\sqrt{n})$	$(1-0.235)^{gpd}$	strong
RED	medium	$O(\mathrm{g.p.d})$	$(1-(1-p)^d)^q$	weak
SDC	medium	$O(\varpi)$	≈ 1	medium
LSCD	high	$\lceil \Psi/r \rceil$	≈ 1	strong

Although the LSCD protocol exhibits a good performance compared to other protocols in terms of the energy utilization rate, network lifetime, the storage performance, the protocol suffers from certain disadvantages. One such disadvantage is that, although establishing the witness path in the LSCD protocol is simple, the clone detection routing is more complex; therefore, the LSCD protocol has higher requirements on the network. The other disadvantage is that this protocol is more sensitive to the failure of the node in terms of the success rate of clone detection, similar to the RED protocol [2]. Because the LSCD protocol clone detection routing can only ensure that there is a detection route to meet the witness, if the route has failed, the success rate of clone detection will be affected.

VII. CONCLUSION AND FUTURE WORK

This paper is the first to propose a clone detection protocol for WSNs whose storage requirement is only a small constant: the LSCD protocol. Utilizing energy for storage, the LSCD protocol stores witness nodes along a ring in a constant-length route and fully utilizes remaining energy in non-hotspots to construct sufficient clone detection routes that are bound to encounter witness routes. Thus, the protocol successfully achieves a small constant storage requirement and a higher detection probability. The LSCD protocol has fully distributed characteristics and strong robustness against attacks. Based on the theoretical analysis and experimental results, the LSCD protocol is proven to improve various performance indicators, namely, the network lifetime is increased by 20%, the detection probability is increased by 50%，and the storage requirements are only 1/5 those of the LSM protocol. The achievements of this research will contribute to the design of energy-efficient and secure WSNs, which represent key components of CPS.

In the preceding discussion, we have assumed that the node IDs cannot be replicated; however, a powerful adversary can

also replicate node IDs, which leads to the need for improved clone detection. In our future work, we would like to explore additional mechanisms to ensure that our protocols continue to function even in the face of powerful adversaries who can replicate node IDs. Moreover, trust is a powerful mechanism for detecting and distinguishing misbehaving nodes. We could also use trust techniques in conjunction with traditional clone detection protocols to enhance the security of WSNs, thus preventing an adversary from damaging the network.

REFERENCES

[1] M. Dong, X. Liu, Z. Qian, A. Liu, and T. Wang, "QoE-ensured price competition model for emerging mobile networks," *IEEE Wireless Commun.*, vol. 22, no. 4, pp. 50-57, Aug. 2015.

[2] M. Conti, R. Di Pietro, L. V. Mancini, and A. Mei, "Distributed detection of clone attacks in wireless sensor networks," *IEEE Trans. Dependable and Secure Comput.*, vol. 8, no. 5, pp. 685-698, Sep. 2011.

[3] Y. Liu, A. Liu, and S. He, "A novel joint logging and migrating traceback scheme for achieving low storage requirement and long lifetime in WSNs," *AEU Int. J. Electron. Commun.*, vol. 69, no. 10, pp. 1464–1482, Oct. 2015.

[4] H. Choi, S. Zhu, and T. F. La Porta, "SET: detecting Node clones in sensor networks," in Proc. in Secure Comm. '07, 2007, pp. 341-350.

[5] Y. Zeng, J. Cao, S. Zhang, S. Guo, and L. Xie," Random-walk based approach to detect clone attacks in wireless sensor networks," *IEEE J. Select. Areas Commun.*, vol. 28, no. 5, pp. 677-691, Jun. 2010.

[6] K. Xing and X. Cheng, "From time domain to space domain: detecting replica attacks in mobile ad Hoc networks," in INFOCOM 2010 Proc., 2010, pp. 1-9.

[7] J. Ho, M. Wright, and S. K. Das, "Fast detection of mobile replica node attacks in wireless sensor networks using sequential hypothesis testing," *IEEE Trans. on Mobile Comput.*, vol. 10, no. 6, pp. 767-782, Jun. 2011.

[8] J. Luo, L. Zhou, and H. Wen, "Lightweight and effective detection scheme for node clone attack in wireless sensor networks," *IET Wireless Sensor Systems*, vol. 1, no. 3, pp. 137-143, Sept. 2011.

[9] B. Parno, A. Perrig, and V. D. Gligor, "Distributed detection of node replication attacks in sensor networks," *Proceedings*, vol. 5, pp. 49-63, May 2005.

[10] B. Zhu, S. Setia, S. Jojodia, S. Roy, and L. Wang, "Localized multicast: efficient and distributed replica detection in large-scale sensor networks," *IEEE Transactions Mobile Comput.*, vol. 9, no. 7, pp. 913-926, Jul. 2010.

[11] R. J. D'Souza and G. Varaprasad, "Digital signature-based secure node disjoint multipath routing protocol for wireless sensor networks," *IEEE Sensors J.*, vol.12, no. 10, pp. 2941-2949, Oct. 2012.

[12] A. Liu, X. Jin, G. Cui, and Z. Chen, "Deployment guidelines for achieving maximum lifetime and avoiding energy holes in sensor network," *Inf. Sci.*, vol. 230, pp. 197-226, May 2013.

[13] Y. Liu, A. Liu, and Z. Chen, "Analysis and improvement of send-and-wait automatic repeat-reQuest protocols for Wireless Sensor Networks," *Wireless Personal Commun.*, vol. 81, no. 3, pp. 923-959, Apr. 2015.

[14] L. Jiang, A. Liu, Y. Hu, and Z. Chen, "Lifetime maximization through dynamic Ring-based routing scheme for correlated data collecting in WSNs," *Comput. Electr. Eng.*, vol. 41, pp. 191–215, Jan. 2015.

[15] S. Chouhan, R. Bose, and M. Balakrishnan, "A framework for energy-consumption-based design space exploration for wireless sensor nodes," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 7, pp. 1017-1024, Jul. 2009.

[16] R. Brooks, P. Y. Govindaraju, M. Pirretti, N. Vijaykrishnan, and M. T. Kandemir, "On the detection of clones in sensor networks using random key predistribution," *IEEE Trans. Syst., Man, Cybern. C*, vol. 37, no. 6, pp. 1246-1258, Nov. 2007.

[17] W. T. Zhu, "Node replication attacks in wireless sensor networks: bypassing the neighbor-based detection scheme," in NCIS Int. Conf., 2011, pp. 156-160.

[18] W. Naruephiphat, Y. S. Ji, and C. Charnsripinyo, "An area-based approach for node replica detection in wireless sensor networks," in TrustCom Int. Conf., 2012, pp 745-750.

[19] Z. Zheng, A. Liu, L. Cai, Z. Chen, and X. Shen, "Energy and memory efficient clone detection in wireless sensor networks," *IEEE Transactions Mobile Comput.* 2015.

[20] J. Ho, M. Wright, and S. K. Das, "Distributed detection of mobile malicious node attacks in wireless sensor networks," *Ad Hoc Netw.*, vol. 10, no. 3, pp. 512–523, May. 2012.

[21] Y. Hu and A. Liu, "An efficient heuristic subtraction deployment strategy to guarantee quality of event detection for WSNs," *Comput. J.*, vol. 58, no. 8, pp. 1747-1762, Nov. 2015.

[22] M. Dong, K. Ota, L. T. Yang, S. Chang, H. Zhu, and Z. Zhou, "Mobile agent-based energy-aware and user-centric data collection in wireless sensor networks," *Computer Networks*, vol. 74, pp. 58-70, Dec. 2014.

[23] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," *Proceedings*, vol.'02, pp. 41-47, Nov. 2002.

[24] M. Dong, K. Ota, A. Liu, and M. Guo, "Joint optimization of lifetime and transport delay under reliability constraint wireless sensor networks," *IEEE Trans. Parallel Distrib. Syst.*, vol. 27, no. 1, pp. 225-236, 2016.

[25] W. Huo, S. Mohammed, J. C. Moreno, and Y. Amirat, "Mobile target detection in wireless sensor networks with adjustable sensing frequency," *IEEE Systems Journal*, pp. 1–14, 2014.

[26] OMNet, Network Simulation Framework. [Online]. Available: http://www.omnetpp.org