Future Generation Computer Systems 143 (2023) 179–190

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

Metaverse intrusion detection of wormhole attacks based on a novel statistical mechanism

Shu-Yu Kuo [a,b,c], Fan-Hsun Tseng [d], Yao-Hsin Chou [e,f,*]

[a] IBM Quantum Hub at National Taiwan University, Center for Quantum Science and Engineering, National Taiwan University, Taipei, 10617, Taiwan
[b] Department of Physics and Center for Theoretical Physics, National Taiwan University, Taipei, 10617, Taiwan
[c] Department of Computer Science and Engineering, National Chung Hsing University, Taichung, 40227, Taiwan
[d] Department of Computer Science and Information Engineering, National Cheng Kung University, Tainan, 701401, Taiwan
[e] Department of Computer Science and Information Engineering, National Chi Nan University, Puli, 54561, Taiwan
[f] Physics Division, National Center for Theoretical Sciences, Taipei, 10617, Taiwan

ARTICLE INFO

Article history:
Received 24 September 2022
Received in revised form 3 December 2022
Accepted 18 January 2023
Available online 24 January 2023

Keywords:
IoT security
Network security
Malware analysis
Wormhole attacks
Intrusion detection
Probability ratio test
Mobile cloud security
Security for Metaverse
Virtualization security
Cyber-Physical security

ABSTRACT

The Metaverse shows great potential to facilitate the development of new technologies. Because the security of the Metaverse has attracted considerable attention, the automatic detection of malware in different scenarios related to the Metaverse has become significant as well. A Metaverse-based wireless system comprises various physical and virtual sensing models, and the security between these sensors and nodes should be further considered. A wormhole link is created by two malicious radio transceivers connected by high capacity out-of-band wireless or wired links. Wormhole attacks can easily subvert many network protocols and maliciously collect a large amount of traffic. As many new mobile applications in the Internet of Things (IoT) have emerged recently, the threat posed by wormhole attacks has increased. Accordingly, research efforts have been made to develop countermeasures for wormhole attacks. However, most of them have considered a static network, assuming that a wormhole does not move and that wormhole nodes cannot adaptively turn their radio on or off to avoid being detected. These assumptions limit the use of existing detection methods. Hence, we first study the security impact and characteristics of wormhole attacks in mobile cloud and Metaverse environments and find the possibility of matching statistical methods such as the sequential probability ratio test (SPRT) to detect wormholes. Moreover, in our assumed topology, nodes in the network have mobility. From our investigation results, we attempt to derive a novel defense mechanism design against wormhole attacks.

1. Introduction

Metaverse provides immersive experiences between the virtual and physical worlds with the support of interactive experience technologies, cloud computing, artificial intelligence, etc. The expansion of the next-generation wireless communication technology and industry related to the Metaverse shows great potential [1]. The Internet of Things (IoT) is a convenient way to achieve communication between humans, mobile devices, GPS nodes, smart sensors, and people who share information with each other through many functional tools, particularly mobile devices such as smartphones. Accordingly, the impact of IoT applications and wireless systems on Metaverse [1,2] has received considerable attention. However, this popular invention may cause security issues and negative network attacks. Among the several types of attacks, we focus on the wormhole attack problem [3–5] because it is one of the most severe attacks in a wireless system.

To complete a wormhole attack, a specific wormhole link, connected by high-capacity out-of-band wireless or wired links, should be connected between two wormhole nodes and inserted into the network topology. This link will be powerful when two wormhole nodes are far enough, such as repeating the packet it receives from one to another immediately. This process does not appear dangerous; however, it may cause destructive damage. For instance, if one normal node wants to check the number of neighbors in its communication range, it will result in some neighbors being actually unreachable because wormholes report illegal nodes, so the routing protocol in use would be consequently unreliable. Therefore, an intrusion detection mechanism for wormhole attacks [6] is required for Metaverse-based wireless environments.

To solve the wormhole attack problem, several methods have been proposed in the literature. We classify these mechanisms

* Corresponding author.
E-mail address: yhchou@mail.ncnu.edu.tw (Y.-H. Chou).

https://doi.org/10.1016/j.future.2023.01.017
0167-739X/

into the *secret-keeping* and *point-view* families. The former involves keeping something secret to identify the difference between wormhole nodes and normal nodes, while the latter finds wormhole nodes by making nodes estimate whether some type of contradiction phenomenon exists. The proposed scheme is similar to the *point-view* family. All studies on the *point-view* family follow five concepts: (1) *Methods using distance or timing analysis*, (2) *methods using special hardware*, (3) *methods using special guarding nodes*, (4) *methods using neighborhood discovery*, and (5) *methods using network topology*. We agree that the above concepts have their own advantages; however, each one of them lacks wormhole node ability or has some limitation in performance. We will discuss this in detail later. Moreover, when considering the wormhole attack problem in a mobile network and the Metaverse environment, there is still scarce literature that can provide effective performance. Because the network topology is more unpredictable, the difficulty is increased.

In this study, we propose a novel detection mechanism that uses the statistics sequential probability ratio test (SPRT) method to improve the wormhole attack problem in different environments. This scheme has three main contributions, which are described as follows.

- First, we investigate the Metaverse environment, which contains information on various physical and virtual devices, and propose an intrusion detection scheme to prevent wormhole attacks in this environment. Our scheme performs well in environments with numerous sensors and also benefits a mobile network environment.
- Second, this detection mechanism requires no additional hardware devices or complex calculations, making it an efficient method. Because we observe an intriguing phenomenon when a wormhole node attempts to report its neighbors to another wormhole node, the normal nodes of reachable neighbors will fluctuate in a distinct way.
- Third, we select SPRT as the fundamental approach because it is an effective way to dynamically analyze sequential data. Regardless of whether nodes enter or escape the wormhole attack area, similar fluctuations occur. This characteristic naturally fits the SPRT concept; therefore, we design a scheme to overcome the wormhole attack problem. This also makes it suitable for the Metaverse system, which often includes a variety of data over the course of time.

The rest of this paper is organized as follows: In Section 2, we explain wormhole attacks and the related literature in detail. Section 3 presents the proposed scheme similar to SPRT to solve wormhole attacks. Section 4 presents our parameter configuration and simulation results. Finally, Section 5 concludes the paper and shares some future works.

2. Background and related work

Metaverse comprises physical and meta space. Irrespective of the type of space, it contains various physical and virtual sensors to collect data, monitor situations, provide service, etc., enabling us to have a more immersive experience. Therefore, the wormhole attack in different sensing scenarios needs to be tackled. The wormhole attack is defined in Section 2.1. The existing detection methods will be briefly described in Section 2.2.

2.1. Wormhole attacks

A wormhole attack was independently introduced in [7–9] to a wireless network and is defined as placing two radio transceivers connected by high-capacity out-of-band wireless or wired links. Signals or packets near one transceiver are tunneled through the wormhole link to the other transceiver. After this transmission, it appears that the signals or packets are replayed maliciously at the strategic position. According to their replay strategy, wormhole attacks can be divided into two categories: store-and-forward and instantaneous-resend attacks. In the store-and-forward scheme, one wormhole node copies the entire packet and then transmits it to the other wormhole node. Compared to the simple store-and-forward scheme, the instantaneous-resend strategy is more sophisticated. In the instantaneous-resend strategy, as its name suggests, the wormhole can be launched at the bit level, which means that the replay due to the packet transmission through the wormhole link is performed bit-by-bit even before the entire packet is received. This packet transmission behavior is similar to cut-through routing [10]. Moreover, the instantaneous-resend strategy can also be launched at the physical layer [11], making it similar to the case where the actual physical layer signal is replayed and similar to a physical layer relay [12]. Furthermore, in recent years, the Metaverse has emerged as a potential model for the next generation of the internet, and security issues [13] such as wormhole attacks have received considerable attention.

The wormhole attack creates an illusion for the nodes near one wormhole node that they can directly communicate with the nodes near the other wormhole node. Consequently, the number of immediate neighbors of the nodes near the wormhole node will increase significantly. The above statement indicates that a wormhole attack is easy to launch for three reasons: (1) It is independent of the network protocols that are being executed in the network to be attacked, (2) it is immune to cryptographic techniques because the usual functionalities provided by cryptographic techniques such as key establishment, authentication, and access control, are irrelevant to the existence of wormhole links, and (3) it does not require the adversary to break into the wireless nodes because the existence of wormhole links does not depend on security credentials or confidential messages.

In the presence of wormholes in the network, when the distance between two wormhole nodes is greater than the transmission range, the wormhole appears to provide a shorter, faster, and probably more reliable path to the nodes near the wormhole nodes. Routing mechanisms in wireless networks rely on the shortest-path routing strategy. In this case, the nodes will discover such wormhole-polluted paths and eventually utilize them to deliver data. As shown in [14], nodes are assumed to be uniformly deployed in a sensing region with d nodes per unit area on average, and the wormhole nodes are placed a distance k apart. In this case, approximately at least $\pi dk^2/8$ pairs of nodes will find shorter paths through the wormhole link. This implies that a wormhole attack, particularly one with a long tunneling distance, will be able to attract considerable traffic through the wormhole link because the longer the tunneling distance, the larger the d, and the more the number of node pairs that are affected by the wormhole. The above discussion reveals that the wormhole attack has a powerful position in attacking the network efficiently and effectively because it allows the adversary to exploit this position in various ways.

Overall, the wormhole attack helps the adversary achieve at least the following tasks. The wormhole link, which creates a fake but faster path between two groups of nodes that originally cannot communicate with each other directly, will disturb the routing protocols in the network. From another perspective, the network traffic is controlled by the adversary via wormhole attacks. This means that by launching wormhole attacks, the adversary can collect a large volume of network data more easily. To do so, the adversary must expend considerable effort without the wormhole attack. However, the existence of wormhole links controlling the network traffic can effectively reduce the efforts required by the adversary. Consequently, with the collected network data, the adversary is able to conduct cryptanalysis on

them. The wormhole attack is also helpful for other insider and outsider attacks, making them more effective. The impact of a wormhole attack is mainly measured by the number of pairs whose shortest paths are affected by the wormhole attack. In this sense, a wormhole attack has a larger impact/potentially more damage when two wormhole transceivers are placed relatively far away.

2.2. Wormhole detection

In the literature, several techniques have been proposed to detect wormhole attacks. One approach for preventing wormhole attacks, called *secret modulation*, involves the use of a secret method for modulating bits over transmission. However, this secret modulation approach cannot resist insider attacks; this means that once a node is compromised, this approach will fail unless the radio is kept inside tamper-resistant hardware. Another approach, called *RF watermarking*, authenticates the transmission without decoding the data by modulating the RF waveform in a way known only to authorized nodes [15]. Similar to the secret modulation idea, RF watermarking relies on keeping something secret. Here, the knowledge of which RF waveform parameters are being modulated is the knowledge to be kept secret. Because of the RF watermarking approach, the watermark may still be intact, even though the packet was made to travel beyond the valid wireless transmission range. This is a flaw for RF watermarking in detecting wormholes. One more approach to wormhole detection is intrusion detection. Although intrusion detection could be used in some cases to detect a wormhole, isolating the attacker in a software-only approach is generally difficult because the packets sent by the wormhole are identical to those sent by legitimate nodes. This means that with the use of intrusion-detection techniques, only the presence of some wormholes in the network may be known; unfortunately, their positions cannot be identified. This characteristic renders this approach useless in practice. Recently developed methods have adopted a viewpoint fundamentally different from those of the above methods. However, all recently developed methods still have their respective limitations, *e.g.*, assuming additional hardware or explicit communication models or lacking the ability to identify wormhole links. They are briefly categorized in a way similar to [14] and described below.

2.2.1. Methods using distance or timing analysis

Several schemes have been proposed in attempts to detect wormhole attacks by measuring packet traverse distance or time. The rationale behind this is that tunneling the packets must consume additional time, and the packets being tunneled through the wormhole link must arrive somewhere originally it cannot reach. Unfortunately, the limitation of the methods in this category is that one usually needs to either obtain the node location information using GPS or have very accurate synchronized clocks to bind packet propagation time. Because these two requirements are apparently energy-consuming, it is unclear whether the techniques can be practically implemented in low-cost hardware such as sensors. The methods in this category are briefly described in more detail below.

The technique of packet leashes was first presented in [7]. A leash is any information that is added to a packet designed to restrict the packet's maximum allowed transmission distance. Two types of packet leashes have been proposed in [7]: geographic leash and temporal leash. The basic idea behind the geographic leash is simple: location information is attached to the packet it sends. Each node that receives the packet first checks the authenticity of the location information in the received packet and then forwards or processes the packet if the distance between the locations of the source node and receiving node are within a certain threshold. The basic idea behind the temporal leash is also simple: it is observed that the wormhole needs time to tunnel the signals. Thus, in the temporal leash, the sending-time information is attached to the packet. Each node receiving the packet checks whether the sending time of the source node is within a certain threshold. These two leashes can detect the wormhole only under the assumptions that the geographic position is available to each node and the clock of each node is tightly synchronized, which are not practical.

Two centralized methods, namely, the neighbor number test (NNT) and all distance test (ADT), are presented in [16]. As the names suggest, in NNT, each node periodically sends the neighbor list to the base station. The base station can detect the wormhole according to the number of node appearances. Note that without the wormhole involved in the network, the number of times the node acts as the neighbor of the other nodes follows a hypothetical distribution, which can be calculated according to the knowledge of node density. Therefore, nodes appearing too many times are an indicator that the wormhole exists in the networks. ADT uses an idea similar to that in NNT. In particular, the information of the length of the shortest path to other nodes is sent instead. The drawback of these two methods is that they are centralized, implying high and unbalanced communication costs and a single point of failure.

Wang and Wong proposed a method called EDWA in [17]. The idea of EDWA is quite similar to a geographic leash. The difference between them is that the geographic leash works in a hop-by-hop fashion, while EDWA works in an end-to-end fashion. In particular, it is assumed that the source node adds its geographic position to the packet. With this information, the destination node (not the intermediate nodes) can estimate the hop distance between the source and destination nodes (The mapping formula between the Euclidean distance and hop distance can be found in [18].) In addition, the packet can also record the number of hops it traverses. After comparing the estimated hop distance and the hop distance recorded in the packet, if an unavoidable inconsistency is found, EDWA declares that there is a wormhole in the network.

Truelink is a defense against wormhole attacks proposed in [11]. It focuses on detecting wormholes in IEEE 802.11 networks. The detection effectiveness relies on rapid information exchange, eliminating the intervention of the wormhole. It incurs no additional packet overhead because it is designed to utilize the unused slot in the IEEE 802.11 MAC frame.

2.2.2. Methods using special hardware

By using purely physical layer mechanisms, wormhole attacks such as those involving authentication in packet modulation and demodulation [7] can be prevented. However, such techniques require special hardware, restricting their applicability and practicality. For example, directional antennas can also be used to prevent wormhole attacks [19]. Obviously, not all devices can be equipped with directional antennas. The directional antenna method in this category is described in more detail below.

Hu and Evans presented a wormhole prevention technique in [19]. Further, a simple observation has been made in [19], which constitutes the basic idea of the corresponding detection method. Assume that a directional antenna is attached to each node. The radio coverage of the antenna, which should be a full circle in an omnidirectional antenna, is restricted or partitioned into several zones. Therefore, we know that nodes A and B are real neighbors of each other only if $A(B)$ can hear the signals emitted from $B(A)$, but the directions of zones they use to hear the other one are opposite. Using this simple criterion, Hu and Evans demonstrated the detection effectiveness of their proposed method. However, the need for a directional antenna is a limitation of this method.

2.2.3. Methods using special guarding nodes

A few protocols in this category [20–22] have been proposed. In these protocols, special-purpose guard nodes are used. These special-purpose guard nodes have known locations, higher transmit power, and different antenna characteristics to attest to the source of each transmission. The rationale is to guarantee that the source of each transmission is within a predefined range of the guard node so that the trace of each transmission is trackable. Obviously, the use of such special-purpose guard nodes limits the applicability of this approach. The methods in this category are briefly described in more detail below.

A graph theoretic technique was proposed in [23]. In particular, utilizing geometric random graphs induced by the communication range constraint of the nodes, the authors in [23] presented the necessary and sufficient conditions for detecting and defending against wormholes. The authors in [23] also presented a defense mechanism based on the proposed technique of local broadcast keys. In [23], many special powerful guard nodes were deployed in the network in advance. A guard node has a much longer communication range, can acquire its geographic position, and has no power consumption limit. The basic idea of this method is that each guard node broadcasts a key to the ordinary nodes under its radio coverage. Subsequently, each node can receive a certain number of keys from different guard nodes. Each pair of neighboring nodes can faithfully establish the neighborhood relationship via those keys from guard nodes. The faithful neighborhood relationship helps eliminate the wormhole. Note that this method is essentially similar to LEAP [24], which is a well-known key predistribution method in sensor networks that can inherently also resist against wormhole attacks. Apparently, this method requires the deployment of special guard nodes, thereby limiting its use.

LiteWorp [20] uses a combination of one-time authenticated neighbor discovery and the use of guard nodes that attest the source of each transmission. However, the neighbor-discovery process can be vulnerable to wormhole attacks if the attack is launched prior to such a discovery. A follow-up paper from the same authors attempted to remove this inefficiency [21] but assumed the availability of location information. As mentioned before, this itself could be suspicious.

Graaf et al. [25] proposed a fully monitored criterion, which is useful for wormhole detection. Graaf et al. proposed an implementation to detect the wormhole based on the fully monitored criterion. The idea is that many special guard nodes should be deployed in advance. This ensures that for a relay path, each pair of consecutive nodes has to be monitored by the guard nodes.

2.2.4. Methods using neighborhood discovery

Statistical approaches can be used to detect the increase in the number of neighbors and the decrease in the lengths of the shortest paths between all pairs of nodes because the presence of wormholes [16] obviously changes some network connectivity characteristics. An implementation of statistical measurements of multipath routing in wormhole detection was proposed in [26]. The above two schemes both assume that the network is free of wormholes. In the literature, this time period is usually called *secure bootstrapping time*. The need for a secure bootstrapping time largely limits the use of this type of method because it is vulnerable if the attack is launched prior to such discovery. The methods in this category are described in more detail below.

A different approach examines the changes in the connectivity graph by the wormhole attacks and looks for forbidden substructures in the connectivity graphs that should not be present in a legal connectivity graph [27]. The idea comes from the fact that in the unit disk graph (UDG) model, the intersection of two unit circles whose centers reside at the edges of each other contains at most two points. Assume that there is a mapping between point and node and a mapping between circle and radio coverage. The above fact can easily be utilized for designing a detection method in which each pair of nodes always checks whether they have more than two common neighbors. Maheshwari et al. [27] extended the above disk-packing result to the case of nonunit circles. Thus, many detection criteria could be utilized, making the detection fairly effective. Nevertheless, this method is effective only under the UDG model, which has been proven to not reflect the signal behavior in reality. Note also that finding a UDG embedding in 2D is an NP-hard problem [28]. Thus, detecting a wormhole attack using connectivity information alone is equally difficult.

Lee and Suzuki [29] presented a detection method called SWAT. The basic idea behind SWAT is that each node continuously monitors the network condition and then emits an alarm once the predefined anomaly is found. Three anomalous phenomena are defined. For example, once the node finds that the number of neighbors of its neighboring node is far less or more than its number of neighbors, an alarm should be announced. For another example, in the aggregation tree model, once the node finds that its parent node has too many children nodes, it also announces an alarm. The biggest problem in this method is that it can only be applied to a static sensor network with an aggregation tree, which is not always the case.

2.2.5. Methods using network topology

The last family of works examines the network topology. Essentially, the wormhole attack drastically changes the network connectivity by gluing links between the nodes near wormhole nodes. Although it appears that the methods in this category are definitely centralized owing to the need for knowledge of network topology, properly designed methods can exploit the knowledge of local connectivity merely to perform the detection. Nevertheless, the common drawback of the methods in this category, whether centralized or distributed, is their high communication overhead. The methods in this category are described in more detail below.

A method called MDS-VOW was proposed in [30]. This is a centralized method in which each node sends its geographic position to the base station. With the location information of each node, the base station can estimate the distance between all pairs of nodes. A base station can construct a map that corresponds to the logical placement of all nodes using a technique called multidimensional scaling. The idea behind MDS-VOW is that this map remains intact and flat if there is no wormhole in the network. Nonetheless, this map will be distorted because the wormhole "connects" or "glues" two points in the map together. MDS-VOW can identify the position of this type of distortion, which is the position of the wormhole.

Xu et al. [31] also presented a method . In essence, this method is a distributed version of MDS-VOW. Each node constructs its own local map by exploiting the neighborhood information and then detects the anomaly from the local map.

Dong et al. developed a method called wormcircle in [32]. The idea comes from a physical phenomenon in wave propagation. In principle, this also relies on the assumption that the wormhole needs additional time to tunnel the signals. In particular, a node acts as a starter node in the wormcircle. A starter node floods a signal to the entire network. According to the flooding behavior, the nodes in the network can be labeled according to their hop distance to the starter node. In the case where there is no wormhole, the labels can be imagined to form many contour lines (or circles). However, in the case where there should be more than two circles, two wormhole nodes are far away from each other. Thus, wormcircle detects whether there are more than two

circles and identifies the positions of their centers, which can be an indicator of the wormhole. Dong et al. suggested the use of global topological properties to detect the presence of wormholes in [33]. This idea has some merit for certain 2-manifolds but does not translate to actual networks because real-world network graphs are not surfaces.

Ban et al. [14] exploited the local connectivity information to devise their detection method. In their method, each node repeatedly checks whether the nodes in the donut region, which is defined to contain its k-hop neighbors, $\alpha \leq k \leq \beta$, form only one connected component. If that is the case, they claim that no wormhole exists in the network. Otherwise, at least one wormhole exists. The rationale behind this detection is that once the wormhole nodes are not considered (the nodes in the inner circle of the donut region), the subgraph induced by the k-hop neighboring nodes will be separated into at least two parts because in reality they are distant from each other. In the course of detection, α and β should be modified for each try. The shortest-path tree needs to be calculated for each try, implying unavoidable computation and communication overhead. In addition, once the node density is insufficient, even if there is no wormhole, the subgraph induced by the nodes in the donut region could still present the possibility of being broken into several parts. The above reasons render the method impractical for detecting wormholes.

Following the discussion of the relevant literature, most of them have great perspectives to detect wormhole attacks and demonstrate possible results. Nonetheless, we list all potential points that could be further improved as follows: (1) The need for secret authentication. (2) The need for GPS data. (3) The need for time synchronization. (4) The need for additional hardware. (5) The need for special-purpose nodes with powerful ability. (6) The need for extra bootstrapping time. (7) The need for large amount transmission packets to communicate. Moreover, as the Metaverse requires a mobile network to run on, if the reported methods assuming a mobile network environment are implemented [34], their efficiency will dramatically deteriorate because of the high challenges in mobile networks [35–39]. Therefore, our proposed scheme makes contributions in two places: (1) Wormhole attack detection is achieved to avoid emerging known defects. (2) The mobile network employment related to the Metaverse system can be beneficial.

3. The proposed intrusion detection system of wormhole attacks

This section presents the proposed wormhole-detection mechanism, wormhole detection with sequential probability ratio test (WD-SPRT), which can be applied to various scenarios. The essence of the proposed mechanism is simplicity, reasonableness, and efficiency. In the WD-SPRT, one of the moving normal nodes is randomly assumed as the investigator and allowed to continuously count the neighbors. If a large fluctuation emerges, it defines this fluctuation into two cases: (1) reducing half of the neighbors compared to the last count and (2) increasing the neighbor count to double when compared with the last count; these two cases are considered to indicate that wormhole attacks may exist. Neighbor information is important to this model. Because wormhole nodes need to play the same role as normal nodes do, both of them should have equal transmission ability. Then, after the investigator just moves into the wormhole node's transmission range and is attacked by the wormhole node, it is obvious after the investigator checks its neighbor count with the value at the last time that it is not yet in the wormhole attack; the related mathematical computation will be discussed later. The flowchart of WD-SPRT is shown in Fig. 1, and its procedure is illustrated in Algorithm 1.

The kernel concept is practical but needs further enhancement. The main concern behind using the neighbor information is the ratio of the mentioned fluctuation; this ratio is the key point affecting our WD-SPRT scheme in achieving more accurate detection of wormhole attacks. A high ratio will make it too slow to make judgments, while a low ratio could lead to low accuracy. It requires a suitable value to properly determine if it is currently being attacked by a wormhole. We investigated this issue and discovered the possibility of a matching sequential probability ratio test (SPRT) inspired by [40]. With a process similar to SPRT, at the beginning, this model presents a dynamic border value and a suitable accumulation fluctuation ratio. After checking the neighbor count for sufficient time, a more reliable judgment is made. The reason this model selects SPRT is that it can assist in making smart decisions through continuous and real-time statistical methods, which can be efficiently implemented in the Metaverse environment. This study aims to determine whether there is an attack by a wormhole or not; therefore, SPRT is more suitable than other methods. SPRT has two thresholds and one test value, which change dynamically with different data coming in, and these two thresholds are used to help judge whether there is a wormhole or not. The rest of this section is organized as follows: Section 3.1 provides an illustration of the random waypoint model in the mobile network environment. Section 3.2 deals with the abovementioned fluctuation and its mathematical calculation. Finally, Section 3.3 shows the details of our proposed method, WD-SPRT.

Algorithm 1: Wormhole Detection with Sequential Probability Ratio Test (WD-SPRT)

```
Collect the information of Area and the number of nodes;
Calculate the average nodes in the one-hop transmission range;
while (Decision is not made) do
    Gather information on the previous and current neighbors ;
    //Apply to SPRT model
    Compute Bernoulli random variable S_i ;
    Determine the value of wc, f_null, and f_altr ;
    if wc ≤ f_null then
        Determine that this scenario is not attacked by the wormhole;
    end
    else if wc ≥ f_altr then
        Determine that this scenario is attacked by the wormhole;
    end
end
```

3.1. Random waypoint model

A mobility model commonly used in the literature (*e.g.* [41–45]) in modeling the mobility of *ad hoc* and sensor networks is the random WayPoint (RWP) model [46]. The random waypoint model is a random model for the movement of mobile nodes. We implement this random waypoint model because of its simplicity and wide availability. Each node's location, velocity, and acceleration change over time. The mobile nodes are able to move randomly and freely. In particular, the destination, speed, and direction are all selected randomly and independently of other nodes. For example, in this model, each node starts with a pause for a fixed number of seconds and then selects a random destination in the given area. Subsequently, it randomly chooses a speed between 0 and maximum and then moves to the destination;

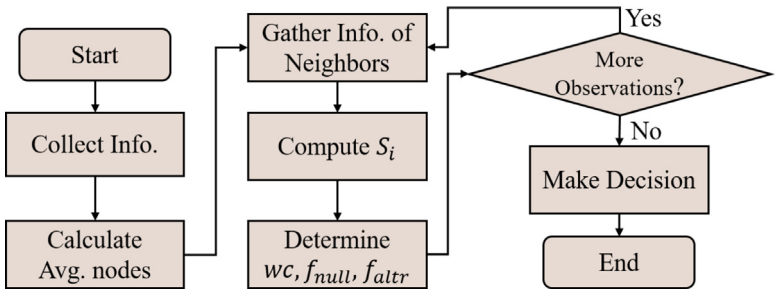


Fig. 1. Flowchart of WD-SPRT.

then, it repeats the above movements until the end. Obviously, an evenly distributed node topology will be created by this flexible model because every possible position has the same possibility of employing a node.

3.2. Fluctuation occurrence probability

As we described in the first part of Section 3, we defined a fluctuation that represents a type of alert message to remind our investigator – any one of all nodes – of the unusual neighbor count caused by a wormhole attack. The method attempts to find the double or half neighbors compared with the last connected node check, and the mathematical explanation is also simple. Suppose the investigator keeps moving in a random waypoint model with the same transmission range R as all nodes. The entire implementation area size is *Area* and the total number of nodes is N. The average nodes in the one-hop transmission range can be defined as follows:

$$R^2\pi\frac{N}{Area} \tag{1}$$

In other words, the random waypoint model ensures arbitrary node movement, and we use the ratio between the total number of nodes N and the area size *Area* to estimate the probability of node employment for each unit area. Then, we use the same ratio to multiply the transmission range area size to calculate the average neighbor count after the investigator, i.e., any one of all nodes, checks their connected nodes in one hop. Moreover, the total number of nodes N does not need to be determined specifically because Eq. (1) ensures that all nodes in the same scenario should have the same result, and the fluctuation mentioned is posed only by the multiple relationship. This result also supports our assumed fluctuation. If normal nodes – even wormhole nodes – imitate ordinary rules with the same delivery ability, they are both inside the implement area without overlapping. The neighbor-measuring result doubles when the investigator just enters the wormhole attack range last movement, while the ordinary node goes outside the attacked area, causing the connected node checking result to reduce to half. The scene diagram is illustrated in Fig. 2.

Another interesting point is the probability that the ordinary node meets the fluctuation. In the previous paragraph, we described the fact that an unusual phenomenon would emerge just between the movements when investigators enter and go outside the wormhole node transmission range R. Therefore, R and the area size *Area* play important roles in quantifying the fluctuation frequency, shown as follows:

$$\frac{2R\pi}{Area} \tag{2}$$

Instead of adopting the entire transmission range circle, we only take the circular border line into consideration. In particular, the frequency of an investigator meeting a fluctuation is decided by how often it goes to "'see" whether the border wormhole nodes are reached, not the integral attack area.

3.3. Wormhole detection with sequential probability ratio test (WD-SPRT)

According to [40], the traditional SPRT needs improvement to fit our problem. Before setting two conflicting hypotheses H_0 and H_1 in the first step, we recognize $neighbor_{this}$ as the total neighbor number that an investigator i estimates after the current movement, and $neighbor_{last}$ is the result before the last movement. Then, we can apply the Bernoulli random variable S_i definition as follows:

$$S_i = \begin{cases} 1, & \text{if } neighbor_{this} \geq 2neighbor_{last} \\ & \text{or } neighbor_{last} \geq 2neighbor_{this}, \\ 0, & \text{if } neighbor_{this} < neighbor_{last} \\ & \text{or } neighbor_{last} < 2neighbor_{this}. \end{cases} \tag{3}$$

Assuming that the investigator frequently surveys its connective neighbors, it will obtain a 1 or 0 value according to the conditions each time. The movement of invading the wormhole node transmission area might increase the checking result; a double value compared with the early consequence is the threshold to obtain Bernoulli random variable 1 and vice versa when oppositely shifting out of the wormhole attack zone to obtain 0. After discovery of a large number of neighbors, the ratio accumulation of S_i divided by the total number of investigations is meaningful. If the ratio value is high enough, we can conclude that the H_1 hypothesis – a wormhole attack – is real with more confidence. Of course, the H_0 hypothesis, which states that there is no wormhole attack, would be acceptable with a low ratio.

From the literature, we know that the likelihood ratio is a key point in processing SPRT, and precise parameters are required to achieve better performance. The formulation is shown below.

$$L_n = \ln\frac{Pr(S_1, S_2, \ldots, S_n|H_0)}{Pr(S_1, S_2, \ldots, S_n|H_1)} \tag{4}$$

We assumed that a random waypoint model was employed, so every node neighbor investigation was independent for the same reason that all node appearances were independent. Then, the accumulation of S_i should be independent and identically distributed. Additionally, the likelihood ratio L_n can be written as follows.

$$L_n = \ln\frac{\prod_{i=1}^{n} Pr(S_i|H_0)}{\prod_{i=1}^{n} Pr(S_i|H_1)} = \sum_{i=1}^{n}\ln\frac{Pr(S_i|H_0)}{Pr(S_i|H_1)} \tag{5}$$

SPRT is a simple statistical method to estimate two conflicting hypotheses that are true, but it has a restriction that any unknown parameter is not allowed. To fit this condition, we must attempt to include conditional probability in Eq. (5) a reasonable estimation, and Eq. (2) is accordingly convenient. We define λ as the probability that the Bernoulli random variable S_i obtains the value 1, and this probability is related to the ratio in Eq. (2):

$$Pr(S_i = 1) = 1 - Pr(S_i = 0) = \lambda \equiv x\frac{2R\pi}{Area}, x \in R \tag{6}$$

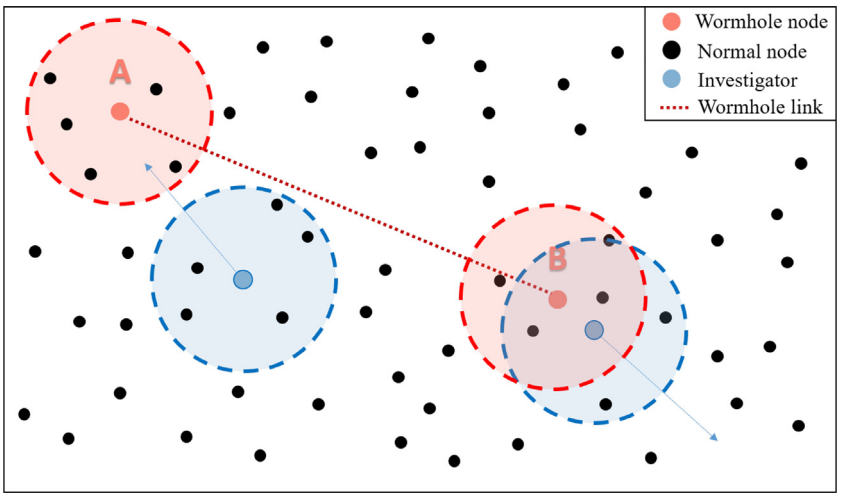


Fig. 2. Scenarios of investigators discovering wormhole attacks.

However, we still need λ_0 and λ_1 to represent the additional probability, λ_0, similar to the possible chance of finding a neighbor count fluctuation when the H_0 hypothesis is actually true, with no wormhole nodes in the environment. λ_1 denotes the opportunity for an investigator to capture the same type of fluctuation if the H_1 hypothesis is real. Absolutely, $\lambda_0 < \lambda_1$ is a premise to take λ into consideration with preset tolerance. Therefore, we define these three parameters as follows:

$$\lambda_0 \leq \lambda \leq \lambda_1. \tag{7}$$

where

$$Pr(S_i = 1|H_0) = \lambda_0, Pr(S_i = 1|H_1) = \lambda_1. \tag{8}$$

Thus far, we have cautiously given each unknown parameter value; now, we can write the likelihood ratio Eq. (5) in an uncomplicated way as follows:

$$L_n = w \ln \frac{Pr(S_i = 1|H_0)}{Pr(S_i = 1|H_1)} + (n - w) \ln \frac{Pr(S_i = 0|H_0)}{Pr(S_i = 0|H_1)} \tag{9}$$

where n is the total number from our WD-SPRT sampling, and w is the count that the Bernoulli random variable gives a value of 1. The advanced significance of λ_0 and λ_1 is to let likelihood ratio L_n have a different force to close one of two borders; in these two borders, we follow the same process as the traditional SPRT does. First, the value is determined for type I and type II error possibilities α and β, respectively. Then, the borders and entire WD-SPRT is given as follows:

$$\begin{cases} L_n \leq \ln \frac{1-\beta}{\alpha} & \text{:accept } H_0 \text{ and terminate the test} \\ L_n \geq \ln \frac{\beta}{1-\alpha} & \text{:accept } H_1 \text{ and terminate the test} \\ \ln \frac{1-\beta}{\alpha} < L_n < \ln \frac{\beta}{1-\alpha} & \text{:continue the test process with another observation.} \end{cases} \tag{10}$$

We rewrite it as follows:

$$\begin{cases} wc \leq f_{null}(n) & \text{:accept } H_0 \text{ and terminate the test} \\ wc \geq f_{altr}(n) & \text{:accept } H_1 \text{ and terminate the test} \\ f_{null}(n) < wc < f_{altr}(n) & \text{:continue the test process with another observation.} \end{cases} \tag{11}$$

where,

$$f_{null}(n) = \ln \frac{1-\beta}{\alpha} - n \ln \frac{1-\lambda_0}{1-\lambda_1}, f_{altr}(n) = \ln \frac{\beta}{1-\alpha} - n \ln \frac{1-\lambda_0}{1-\lambda_1},$$

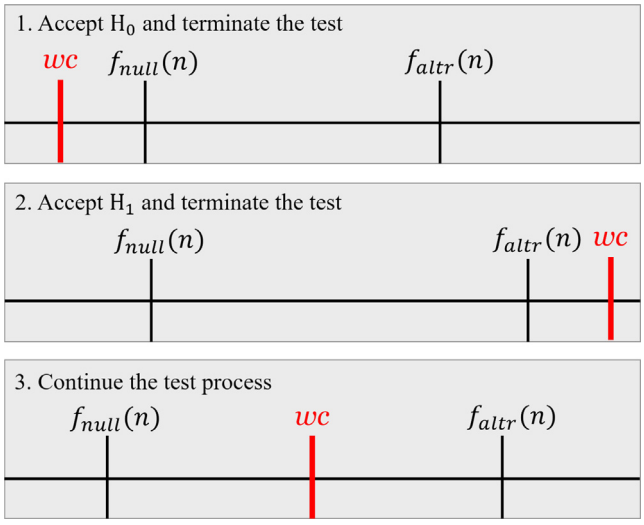


Fig. 3. Main concepts of WD-SPRT.

$$c = (\ln \frac{\lambda_0}{\lambda_1} - \ln \frac{1-\lambda_0}{1-\lambda_1}).$$

The last point to note is variable c. To avoid incorrect WD-SPRT consequences caused by negative value division, this equation must multiply w before making comparisons. After all the preparation work is finished, our proposed wormhole-detection scheme, WD-SPRT, is now ready to contribute to mobile networks related to the Metaverse. Fig. 3 shows the main concepts of WD-SPRT, which can collect neighbor information and dynamically update the thresholds and current value until it can determine the situations.

4. Simulations

In this section, we run our proposed WD-SPRT scheme to determine whether it is useful. As we mentioned in Section 3, WD-SPRT is a scheme based on SPRT, and the main restriction of SPRT is to configure all parameters without any unknown value. How to preset all possible variables is the challenge to earn better performance. The other key point affecting the simulation result is our random waypoint model. We believe that our employed nodes should be uniformly distributed in the whole topology if

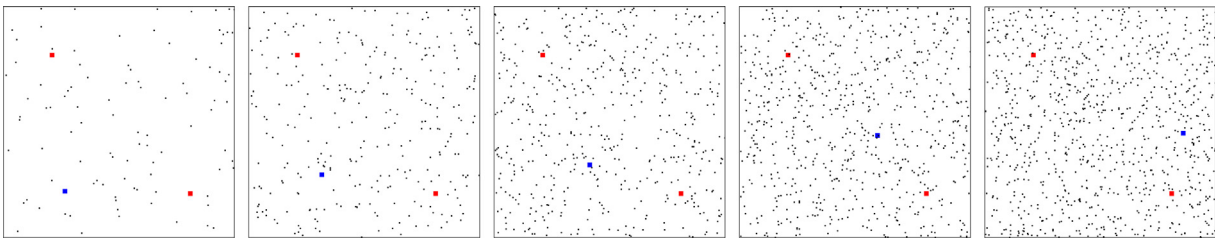


Fig. 4. 100, 300, 500, 700, and 900 mobile nodes employed in a 500 × 500 square area.

we carefully configure our scenario. Therefore, in the rest of this section, we will discuss it in two directions. The first simulation environment will be explained for each step in Section 4.1, and soon the simulation result will be illustrated in Section 4.2.

4.1. Simulation environment

Our simulation is run by the program we write in C++, and at the beginning, we assumed a square area of 500 × 500 m with 100 to 900 mobile sensor nodes, giving these nodes random directions and moving distance as a random waypoint model defining. Moreover, we make each node have two levels of mobility to test our simulation applicability in different scenarios: an ability of 0 10 m/s speed with arbitrary direction for urban mobile nodes and suburbs nodes with 0 20 m/s. More specifically, we separate all possible directions into $x - axis$ and $y - axis$ similar to $2 - D$ coordination with both positive and negative directions, and a randomly created speed value might be set to this coordination system intending to avoid nodes going outside the area. We have refraction based on light characteristics to keep nodes always in the square scenario. The last factor is to make these node movements change frequently; we let all nodes update their velocity information every second in intuition. The illustration in Fig. 4 is shown to ensure that nodes are uniformly distributed as a random waypoint model. We diagonally put an illegal link connected by two wormhole nodes to each scenario as the red squares without moving ability, while the blue square is the investigator randomly chosen to sample each simulation round.

The transmission range has two values: a 50-meter delivery radius similar to Bluetooth and a 100-meter match to Wi-Fi. The additional probabilities λ_0 and λ_1 are given in four ways according to Table 1 after referring to Eq. (2). We notice that a special case of low node average neighbors causes SPRT mistakes many times if the total neighbor count this time changed from one to two in the last investigation. The same result might be normal even if we check node count fluctuation in low node density, so we consider a threshold that both $neighbor_{last}$ and $neighbor_{this}$ should be larger than 2 to avoid confusing samples and causing our scheme to be misunderstood.

Finally, we calculate the performance. We run each combination of parameters 100 times, in which 50 times no wormhole nodes are employed, while the last 50 times a wormhole attack exists. In the end, we count the classical type I error and type II error possibility when α and β always have the same value of 0.01 and then record the average sampling times of each correct judgment to compare whether null or alternative hypotheses need more investigation.

4.2. Simulation results

This simulation investigates the impact on transmission ranges, speeds, and various numbers of nodes in the scenario and examines the significant diverse parameter setting combinations, as Table 1 shows. The simulation findings are divided into two

Table 1
Combinations of λ_0, λ_1 and transmission range.

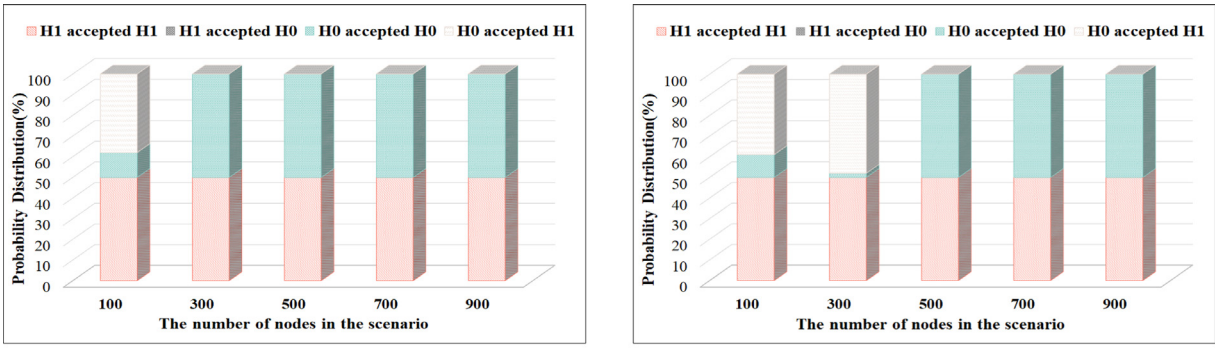
Transmission range	50 m	50 m	100 m	100 m
λ_0	1.25 ‰	2.5 ‰	2.5 ‰	5 ‰
λ_1	2.5 ‰	5 ‰	5 ‰	7.5 ‰

categories: one in which the wormhole is immobile or static, as shown in Figs. 5–8, and another in which it is mobile or dynamic, as shown in Figs. 9–12. In these simulation figures, the orange color parts indicate there is a wormhole attack and the proposed detection scheme determines positive (H_1 accepted H_1, true positive, Sensitivity). The green color parts indicate there is no wormhole attack, and the proposed WD-SPRT judges negative (H_0 accepted H_0, true negative, Specificity). It is important to note two evaluations: sensitivity and specificity. The higher both the sensitivity and specificity, the better.

Figs. 5–6 demonstrate that as the number of nodes increases, the detection success rate rises, indicating that both sensitivity and specificity are excellent. The sensitivity is high when the number of nodes is limited, but specificity needs to be further enhanced because not sufficient information is captured in this circumstance. In the 50 m transmission cases, it can discover that the H_0 hypothesis, which states that there are no wormhole nodes, might be accepted as an improper judgment due to the low node velocity. Too low velocity directly increases the difficulty of meeting the fluctuation of wormhole attacks, leading to errors after sufficient sampling cycles with no fluctuation discovery. Moreover, a lower node density also makes it easy to break our restriction of the Bernoulli random variable, and misunderstanding a wormhole attack appears to accept the alternative hypothesis H_1. The above findings will be more obvious when giving nodes more energetic actions of max 20 m/s velocity, especially in the loose node scenario. It can be noticed that more connectivity will lead to more accurate efficiency after giving higher node communication ability.

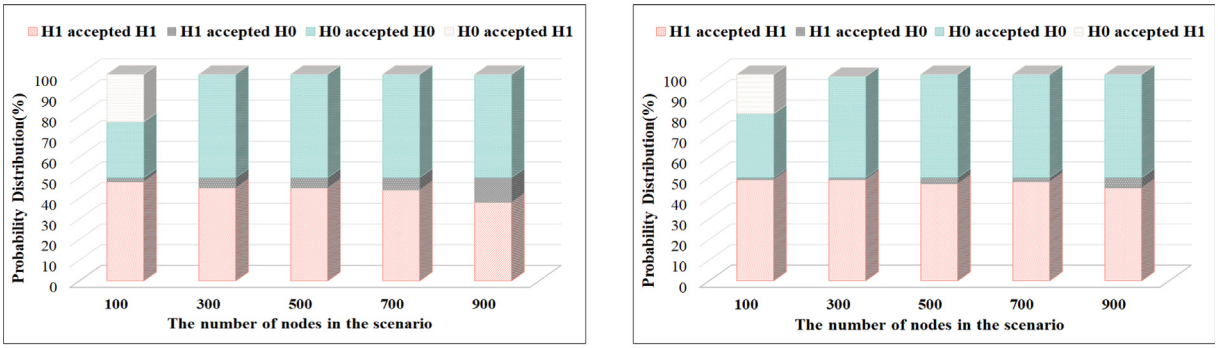
On the other hand, we believe WD-SPRT can earn better performance when the transmission range is proportional to the node velocity because the condition that a small transmission range matches high speed has our scheme rashly accepted H_1 with a high frequency of fluctuation, while an oppositely large transmission range matching low speed may extremely increase the difficulty of fluctuation sampling. Figs. 7–8 indicate several simulation outcomes in which nodes with 100 m transmission range circumstances. In these circumstances, two different λ_0 and λ_1 settings both show promising results and reach a more stable performance.

Based on the above discovers, when the wormhole is immobile or static, the transmission range is 50 m situations, and the best parameter settings are $\lambda_0 = 2.5‰$, $\lambda_1 = 5‰$, and the velocity of nodes from 0 to 20 m/s. When the transmission range is 100 m situations, and both parameter settings that $\lambda_0 = 2.5‰$, $\lambda_1 = 5‰$, and $\lambda_0 = 5‰$, $\lambda_1 = 7.5‰$ achieve potential results. This model can produce promising results when it collects sufficient



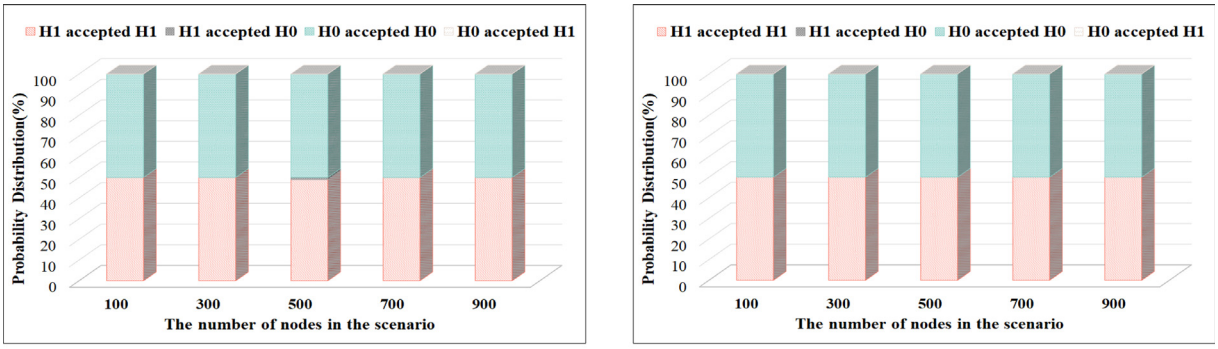
(a) (b)

Fig. 5. Detection results with $\lambda_0 = 1.25‰$, $\lambda_1 = 2.5‰$, 50 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s.



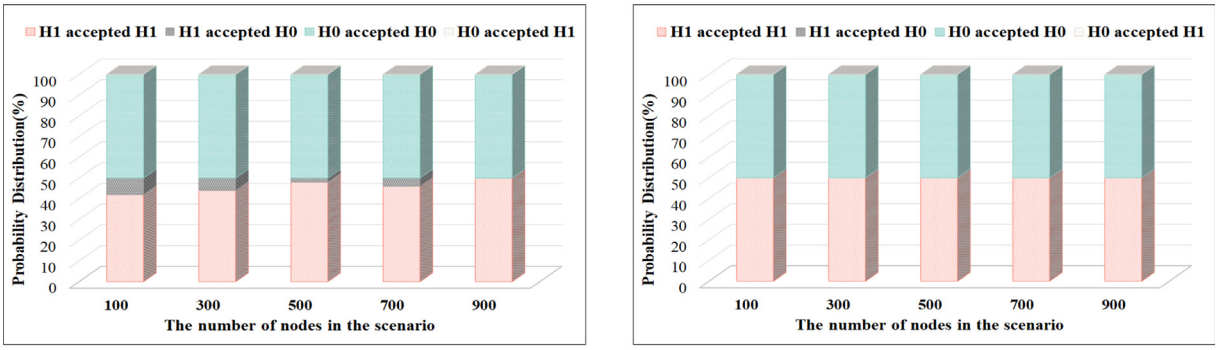
(a) (b)

Fig. 6. Detection results with $\lambda_0 = 2.5‰$, $\lambda_1 = 5‰$, 50 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s.



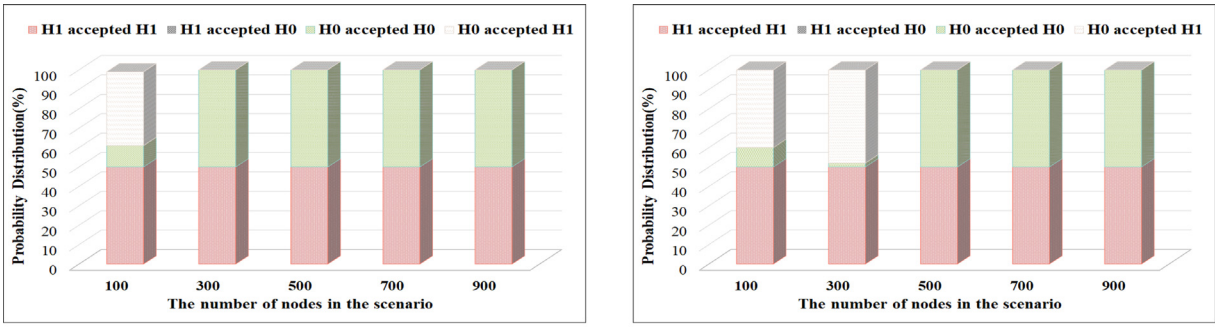
(a) (b)

Fig. 7. Detection results with $\lambda_0 = 2.5‰$, $\lambda_1 = 5‰$, 100 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s.



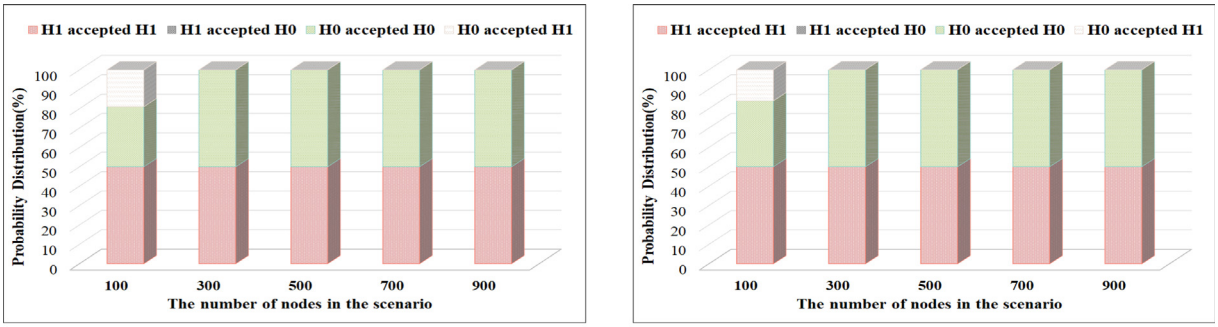
(a) (b)

Fig. 8. Detection results with $\lambda_0 = 5‰$, $\lambda_1 = 7.5‰$, 100 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s.



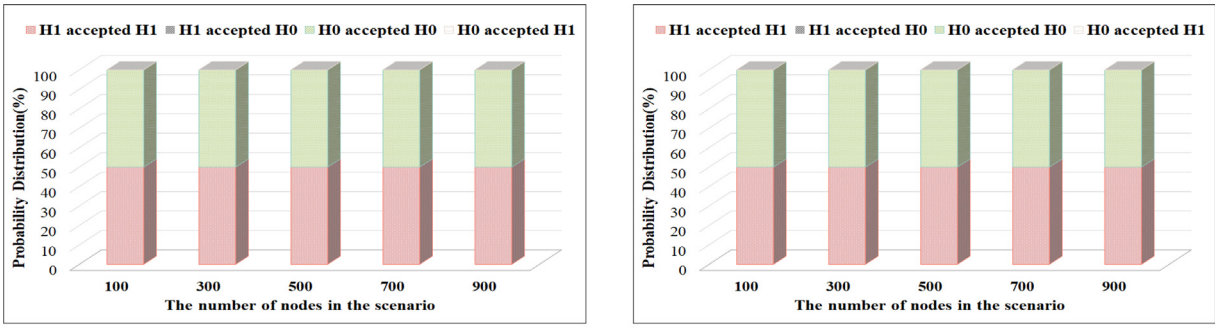
(a) (b)

Fig. 9. Detection results with $\lambda_0 = 1.25‰$, $\lambda_1 = 2.5‰$, 50 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s. The wormhole possesses mobility.



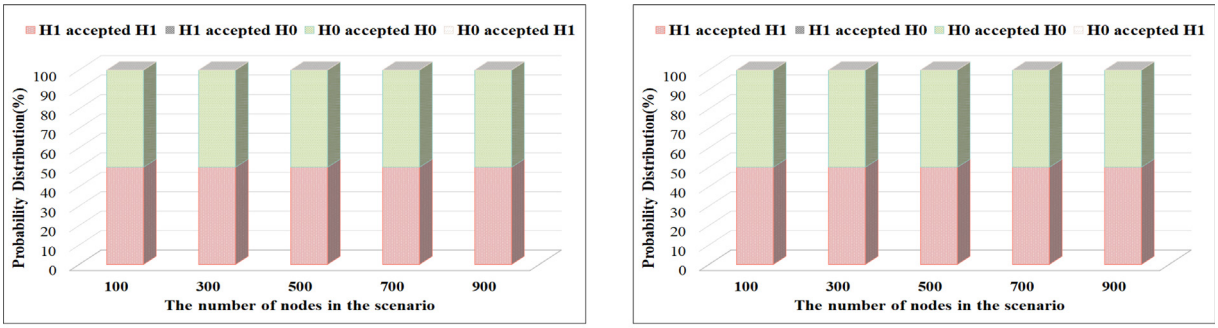
(a) (b)

Fig. 10. Detection results with $\lambda_0 = 2.5‰$, $\lambda_1 = 5‰$, 50 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s. The wormhole possesses mobility.



(a) (b)

Fig. 11. Detection results with $\lambda_0 = 2.5‰$, $\lambda_1 = 5‰$, 100 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s. The wormhole possesses mobility.



(a) (b)

Fig. 12. Detection results with $\lambda_0 = 5‰$, $\lambda_1 = 7.5‰$, 100 m transmission range, and the velocity of nodes randomly from 0 to (a) 10 and (b) 20 m/s. The wormhole possesses mobility.

neighbor information, which typically occurs in environments with a high transmission range and high node density. This simulation is pretty suitable for large-scale Metaverse systems that contain a high capacity of physical and virtual nodes, sensors, monitors, and devices.

Figs. 9–12 show the experimental results that all nodes, including the wormhole sensors, have mobile capability. Compare with Figs. 5–8, both scenarios show some similar phenomena. In Figs. 9 and 10, the WD-SPRT model earn better performances when the sensor of nodes in the scene is large. But when the node density is low, the situation of False Positive which is H_0 accepted H_1 happens. In Figs. 11–12, as the communication range of nodes enhances to 10 m/s, the successful rate increases to 100%, indicating that all situation are correctly determined and both sensitivity and specificity are attained flawlessly. Compared to the static wormhole scene with the same parameters, the success rate in the mobile wormhole environment is even greater. In mobile wormhole scenes, investigators have a greater chance of encountering the wormhole's attack range; therefore, this model collects sufficient data to make proper decisions. In summary, the proposed scheme WD-SPRT is useful not only in the scenario with sufficient node density, high node moving ability, and large transmission range but also in the situation with no special cases.

5. Conclusions and future works

Metaverse has great potential to facilitate new technologies, but the challenges of security issues also arise. Because the Metaverse ecosystem contains many sensing mechanisms and technologies, the security of these sensors needs to be further discussed. In this paper, different mechanisms to address the wormhole attack problem in the IoT application related to Metaverse are investigated. We propose a new design for securing IoT nodes against wormhole attacks. This research topic has not yet been discussed well, although wormhole attacks have occurred in the mobile network environment of the IoT, because most metrics proposed in the literature did not consider this scenario. This is an opportunity to attempt an automatic detection method of a sampling scheme based on SPRT. This paper proposes the WD-SPRT mechanism to determine wormhole nodes from the fluctuation of neighbors caused by wormhole attacks, and this concept works in cases with different situations. The proposed new design of a defense mechanism does not require additional hardware devices and can be achieved using a few software resources. This feature is suitable for use in the Metaverse. Experiment findings indicate that our intrusion detection system is applicable to a variety of scenarios, including different transmission ranges, node counts, parameter settings, and static or mobile wormhole node characteristics, and is capable of achieving excellent performance. However, challenges still exist and need to be overcome. The first challenge is that the performance in loose node employment can be further improved because lower average neighbors make it difficult to decide whether the wormhole attack exists. We believe it is related to the design of Bernoulli random variable conditions and parameters λ_0 and λ_1 in our WD-SPRT scheme. Furthermore, as our primary idea is the neighbor count in each sampling, the random waypoint model is another area we could consider in the future. In addition, we are interested in maintaining effectiveness in different scenarios and taking the energy cost into account in various Metaverse systems.

CRediT authorship contribution statement

Shu-Yu Kuo: Methodology, Software, Investigation, Writing – original draft, Visualization. **Fan-Hsun Tseng:** Validation, Resources, Writing – review & editing. **Yao-Hsin Chou:** Conceptualization, Formal analysis, Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported in part by the National Science and Technology Council, Taiwan, under Grants 109-2222-E-005-002-MY3, 111-2221-E-260-014-MY2, 111-2627-M-002-001, and 111-2119-M-033-001; and in part by the National Center for Theoretical Sciences of Taiwan.

References

[1] L.U. Khan, Z. Han, D. Niyato, E. Hossain, C.S. Hong, Metaverse for wireless systems: Vision, enablers, architecture, and future directions, 2022, arXiv:2207.00413.
[2] K. Li, Y. Cui, W. Li, T. Lv, Xin. Yuan, S. Li, W. Ni, M. Simsek, F. Dressler, When Internet of Things meets metaverse: Convergence of physical and cyber worlds, 2022, arXiv:2208.13501 [Cs.NI].
[3] M. Hanif, H. Ashraf, Z. Jalil, N.Z. Jhanjhi, M. Humayun, S. Saeed, A.M. Almuhaideb, AI-based wormhole attack detection techniques in wireless sensor networks, Electronics 11 (15) (2022) 2324–2352.
[4] M.A. Ferrag, L.A. Maglaras, H. Janicke, J. Jiang, L. Shu, Authentication protocols for Internet of Things: A comprehensive survey, Secur. Commun. Netw. (2017).
[5] Parvathy. K., Wormhole attacks in Wireless Sensor Networks (WSN) & Internet of Things (IoT): A review, Int. J. Recent Technol. Eng. (IJRTE) 10 (1) (2021) 199–203.
[6] M. Goyal, M. Dutta, Intrusion detection of wormhole attack in IoT: A review, in: 2018 International Conference on Circuits and Systems in Digital Enterprise Technology, ICCSDET, 2018, pp. 1–5.
[7] Y. Hu, A. Perrig, D.B. Johnson, Packet leashes: A defense against wormhole attacks in wireless ad hoc networks, in: Proceedings of the 22th IEEE International Conference on Computer Communications, INFOCOM, 2003.
[8] P. Papadimitratos, Z. Haas, Secure routing for mobile ad hoc networks, in: Proceedings of SCS Communication Networks and Distributed Systems Modeling and Simulation, 2002.
[9] K. Sanzgiri, B. Dahill, B. Levine, C. Shields, E. Belding-Royer, A secure routing protocol for ad hoc networks, in: Proceedings of IEEE International Conference on Network Protocols, ICNP, 2002.
[10] L.M. Ni, P.K. McKinley, A survey of wormhole routing techniques in direct networks, Computer 26 (2) (1993) 62–76.
[11] J. Eriksson, S.V. Krishnamurthy, M. Faloutsos, Truelink a practical countermeasure to the wormhole attack, in: Proceedings of IEEE International Conference on Network Protocols, ICNP, 2006.
[12] A. Scaglione, Y.W. Hong, Opportunistic large arrays: Cooperative transmission in wireless multihop ad hoc networks to reach far distances, IEEE Trans. Signal Process. 51 (8) (2003).
[13] Y. Wang, Z. Suy, N. Zhang, R. Xing, D. Liu, T.H. Luan, X. Shen, A survey on metaverse: Fundamentals, security, and privacy, IEEE Commun. Surv. Tutorials (2022) (Early Access).
[14] X. Ban, R. Sarkar, J. Gao, Local connectivity tests to identify wormholes in wireless networks, in: Proceedings of the 12th ACM International Symposium on Mobile Ad Hoc Networking and Computing, MobiHoc, 2011.
[15] Defense Advanced Research Projects Agency. Frequently Asked Questions V4 for BAA 01-01, FCS Communications Technology, Washington, DC.
[16] L. Buttyan, L. Dora, I. Vajda, Statistical wormhole detection in sensor networks, in: Proceedings of Second European Workshop on Security and Privacy in Ad-Hoc and Sensor Networks, ESAS, 2005.
[17] X. Wang, J. Wong, An end-to-end detection ofwormhole attack in wireless ad-hoc networks, in: Proceedings of the 31st Annual International Computer Software and Applications Conference, COMPSAC, 2007.
[18] H. Wu, C. Wang, N.-F. Tzeng, Novel self-configurable positioning technique for multi-hop wireless networks, IEEE/ACM Trans. Netw. 13 (3) (2005) 609–621.
[19] L. Hu, D. Evans, Using directional antennas to prevent wormhole attacks, in: Proceedings of Network and Distributed System Security Symposium, NDSS, 2004.

[20] I. Khalil, S. Bagchi, N.B. Shroff, LITEWORP. A lightweight countermeasure for the wormhole attack in multihop wireless network, in: Proceedings of the International Conference on Dependable Systems and Networks, DSN, 2005.
[21] I. Khalil, S. Bagchi, N.B. Shroff, MOBIWORP mitigation of the wormhole attack in mobile multihop wireless networks, Ad Hoc Netw. 6 (3) (2008).
[22] R. Poovendran, L. Lazos, A graph theoretic framework for preventing wormhole attacks in wireless ad hoc networks, ACM J. Wirel. Netw. (WINET) 13 (2005).
[23] L. Lazos, R. Poovendran, C. Meadows, P. Syverson, L.W. Chang, Preventing wormhole attacks on wireless ad hoc networks: A graph theoretic approach, in: Proceedings of IEEE Wireless and Communications and Networking Conference, WCNC, 2005.
[24] S. Zhu, S. Setia, S. Jajodia, LEAP: Efficient security mechanisms for large-scale distributed sensor networks, in: Proceedings of the Annual ACM Computer and Communications Security, CCS, 2003.
[25] R. Graaf, I. Hegazy, J. Horton, R. Safavi-Naini, Distributed detection of wormhole attacks in wireless sensor networks, in: Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering (ADHOCNET 2009), vol. 28, (Part 1) 2010, pp. 208–223.
[26] L. Qian, N. Song, X. Li, Detection of wormhole attacks in multi-path routed wireless ad hoc networks: A statistical analysis approach, J. Netw. Comput. Appl. 30 (1) (2007) 308–V330.
[27] R. Maheshwari, J. Gao, S.R. Das, Detecting wormhole attacks in wireless networks using connectivity information, in: Proceedings of the 26th IEEE International Conference on Computer Communications, INFOCOM, 2007.
[28] H. Breu, D.G. Kirkpatrick, Unit disk graph recognition is NP-hard, Comput. Geom. Theory Appl. 9 (1-2) (1998) 3–24.
[29] C. Lee, J. Suzuki, Swat a decentralized self-healing mechanism for wormhole attacks in wireless sensor networks, in: Yang Xiao, Hui Chen, Frank H. Li (Eds.), Handbook on Sensor Networks.
[30] W. Wang, B. Bhargava, Visualization of wormholes in sensor networks, in: Proceedings of the 3rd ACM Workshop on Wireless Security, WiSe, 2003.
[31] Y. Xu, G. Chen, J. Ford, F. Makedon, Detecting wormhole attacks in wireless sensor networks, 2007, IFIP International Federation for Information Processing.
[32] D. Dong, M. Li, Y. Liu, X. Liao, Wormcircle: Connectivity-based wormhole detection in wireless ad hoc and sensor networks, in: Proceedings of the Fifteenth International Conference on Parallel and Distributed Systems, ICPADS, 2009.
[33] D. Dong, M. Li, Y. Liu, X. Li, X. Liao, Topological detection on wormholes in wireless ad hoc and sensor networks, in: Proceedings of IEEE International Conference on Network Protocols, ICNP, 2009.
[34] M. Karthigha, L. Latha, K. Sripriyan, A comprehensive survey of routing attacks in wireless mobile ad hoc networks, in: International Conference on Inventive Computation Technologies, ICICT, 2020, pp. 396–402.
[35] T. Camp, J. Boleng, V. Davies, Mobility models for ad hoc network simulations, Wirel. Commun. Mob. Comput. (WCMC): Special Issue Mob. Ad Hoc Netw.: Res. Trends Appl. 2 (5) (2002) 483–502.
[36] S. Capkun, L. Buttyan, J.P. Hubaux, SECTOR: Secure tracking of node encounters in multi-hop wireless networks, in: Proceedings of the 1st ACM Workshop on Security of Ad Hoc and Sensor Networks, SASN, 2003.
[37] S. Capkun, J.-P. Hubaux, L. Buttyan, Mobility helps security in ad hoc networks, in: The 17th Annual International Conference on Mobile Computing and Networking, MobiCom, 2003.
[38] D. Estrin, Participatory sensing: applications and architecture, in: Proceedings of the 8th International Conference on Mobile Systems, Applications and Services, MobiSys, 2010.
[39] A. Jardosh, E.M. Belding-Royer, K.C. Almeroth, S. Suri, Towards realistic mobility models for mobile ad hoc networks, in: The 17th Annual International Conference on Mobile Computing and Networking, MobiCom, 2003.
[40] Jun-Won Ho, Matthew Wright, Sajal K. Das, Fast detection of mobile replica node attecks in wireless sensor networks using sequential hypothesis testing, IEEE Trans. Mob. Comput. 10 (6) (2011) 767–782.
[41] P. De, Y. Liu, S.K. Das, Energy-efficient reprogramming of a swarm of mobile sensors, IEEE Trans. Mob. Comput. 9 (5) (2010).
[42] L. Hu, D. Evans, Localization for mobile sensor networks, in: ACM International Conference on Mobile Computing and Networking, MobiCom, 2004.
[43] J. Yi, J. Koo, H. Cha, A localization technique for mobile sensor networks using archived anchor information, in: IEEE Conference on Sensor, Mesh, and Ad Hoc Communications and Networks, SECON, 2008.
[44] L. Zhou, J. Ni, C.V. Ravishankar, Supporting secure communication and data collection in mobile sensor networks, in: IEEE International Conference on Computer Communications, INFOCOM, 2006.
[45] G. Lin, G. Noubir, R. Rajaraman, Mobility models for ad hoc networks, in: Proceedings of the 22th IEEE International Conference on Computer Communications, INFOCOM, 2004.
[46] D.B. Johnson, D.A. Maltz, Dynamic source routing in ad hoc wireless networks, Mob. Comput. (1996) 153–181.

Shu-Yu Kuo received the Ph.D. degree from the Department of Computer Science and Information Engineering, National Chi Nan University, Nantou, Taiwan, in 2018. She was a visiting postdoctoral research associate with the Department of Electrical Engineering, Princeton University, Princeton, NJ, USA, in 2018 and 2019. In 2019, she was a visiting scholar with the Department of Electrical and Computer Engineering, University of Washington, Seattle, WA, USA. From 2019 to 2020, she joined the faculty of the Department of Computer Science and Information Engineering, National Taipei University of Technology, Taipei, Taiwan. From 2020 to 2023, she was an assistant professor with the Department of Computer Science and Engineering, National Chung Hsing University, Taichung, Taiwan. Currently, she works at IBM Quantum Hub at National Taiwan University, Taipei, Taiwan. Her research interests include information security, network security, quantum cryptography, quantum communication and computation, computational intelligence, metaheuristic algorithms, and financial technology.

Fan-Hsun Tseng received his Ph.D. degree in computer science and information engineering from the National Central University, Taiwan, in 2016. From 2018 to 2021, he joined the faculty of the Department of Technology Application and Human Resource Development, National Taiwan Normal University, Taiwan. In 2021, he joined the faculty of the Department of Computer Science and Information Engineering, National Cheng Kung University, Taiwan, where he is currently an assistant professor. His research interests include mobile networks, cloud and edge computing, IoT and big data, AI/ML and evolutionary computing. Dr. Tseng received the Young Scholar Fellowship from the Ministry of Science and Technology in Taiwan in 2018, the Distinguished Young Scholar Award from the Computer Society of the R.O.C in 2020, and the Best Young Professional Member Award from IEEE Tainan Section in 2021. He was qualified and awarded for the rank of EAI Fellows class of 2021. He has served as associate editor-in-chief of Journal of Computers, associate editor of IEEE Access, Human-centric Computing and Information Sciences, Journal of Internet Technology, and IET Networks, and topic editor of Sensors and Electronics. He is a senior member of the IEEE.

Yao-Hsin Chou received his Ph.D. degree from the Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, in 2009. He is currently a Distinguished Professor with the Department of Computer Science and Information Engineering, National Chi Nan University, Nantou, Taiwan. He has authored over 80 papers in journals and conference proceedings. His current research interests include computational intelligence, network security, financial technology, circuit synthesis and testing, and quantum information science. He is a senior member of the IEEE.