International Journal of Networking and Computing – www.ijnc.org
ISSN 2185-2839 (print) ISSN 2185-2847 (online)
Volume 6, Number 2, pages 345–367, July 2016

A Countermeasure to Eavesdropping on Data Packets by Utilizing Control Packet Overhearing for Radio overlapping Reduced Multipath Routing in Ad Hoc Networks

Tetsuya Murakami, Eitaro Kohno, and Yoshiaki Kakuda
Graduate School of Information Sciences
Hiroshima City University
3-4-1 Ozuka-Higashi, Asaminami-ku, Hiroshima, 731-3194, Japan

Received: February 14, 2016
Revised: May 6, 2016
Accepted: July 6 2016
Communicated by Satoshi Fujita

Abstract

Ad hoc networks are autonomously distributed wireless networks which consist of wireless terminals (hereinafter, referred to as nodes). They do not rely on wireless network infrastructures such as base stations. Relaying nodes and their surrounding nodes are susceptible to data theft and eavesdropping because nodes communicate via radio waves. Previously, we had proposed the secure dispersed data transfer method for encryption, decryption, and transfer of the original data packets. To use the secure dispersed data transfer method securely, we had proposed using the node-disjoint multipath routing method. In this method, multiple versions of encrypted data packets are transferred along each disjoint multipath to counter data packet theft. We had also proposed the enhanced version of the aforementioned routing method to reduce radio area overlap by using rebroadcasting of control packets to counter eavesdropping attacks. In this paper, we propose a multipath routing method to reduce radio area overlap through the introduction of control packet overhearing. We introduce control packet overhearing mechanisms to eliminate excess control packet counts and latency in the pathfinding process. Our main contributions are as follows: (1) our proposed method can reduce radio area overlap without each node's geographical location information (e.g., using GPS information); (2) our proposed method also can eliminate excess control packets and latency without degradation of the security. Furthermore we conducted simulation experiments to evaluate our proposed method. We observed that our proposed method can construct the desired paths with a smaller amount of control packets and a shorter latency in the pathfinding process. We also conducted additional experiments to discuss the applicable scope of our proposed method. As a result, we confirmed that our proposed method was more effective as the average number of adjacent nodes increased.

Keywords: Ad hoc networks, Node-disjoint multipath, Eavesdropping, Control packet overhearing, Secure dispersed data transfer method

1 Introduction

Through wireless communication, nodes in ad hoc networks can communicate with other nodes. While this function is useful for reachability between nodes, ad hoc networks are susceptible to various attacks [1].

Many researchers attempt to categorize attacks in several ways. One such distinction is passive versus active attacks. Attacks are determined to be passive or active based on the actions of adversaries. In active attacks, malicious, falsified, or duplicated data packets are inserted into the network. Malicious behavior has to be recognized and eliminated or mitigated. In passive attacks, adversaries acquire data without creating any disruption in the network. Undetectable malicious behavior must be completely prevented. In general, creating countermeasures for passive attacks is harder than for active attacks.

Both node capture attacks and eavesdropping attacks are passive attacks that make nodes vulnerable to future passive attacks. In node capture attacks, adversaries acquire access privileges to nodes as a superuser, and they steal secret keys or transferred data packets from inside the node. They can also utilize node functions such as wireless communication for eavesdropping on surrounding nodes' conversations. Generally, to counter the theft of the keys and transferred data packets as well as eavesdropping, common key-based and public key-based systems have been proposed. However, these systems consume time with encryption/decryption. Furthermore, they face the problem of keys being stolen or revoked. To counter this, the secure dispersed data transfer method has been proposed [2] [3] [4].

This method utilizes both the threshold secret sharing scheme and a one-way function for the encryption, decryption, and verification of transferred secret data packets. In the encryption process, the secure dispersed data transfer method creates multiple encrypted data packets, (referred to as "shares"), from an original data packet. To create multiple shares, the secure dispersed data transfer method must determine the threshold number. When a node collects more than its threshold number of shares, it can decrypt the original data. The threshold number must be less than or equal to the number of shares. Therefore, once a node collects the threshold number of shares, it does not need to wait for every shares' arrival. To transfer each share, we have to prepare disjoint multiple paths. All nodes except source and destination nodes do not forward multiple disjoint shares simultaneously when we can prepare node-disjoint multiple paths. Therefore, node-disjoint multiple paths are desirable to prevent the theft of the original data being forwarded.

Previously, we have proposed the node-disjoint multipath routing method [5] [6] to collaborate with the secure dispersed data transfer method. In this node-disjoint method with the secure dispersed data transfer method, source nodes can send shares along disjoint multiple paths without bottlenecking. Thus, employing the node-disjoint method with the secure dispersed data transfer method, we can send the original data packets confidentially. In this method, when a node is selected as a member of a path, notification messages are sent to its surrounding nodes to instruct them not to select it as a member of another path. In this method, when a node receives notification messages, it registers the sender's address and eliminates it from its next hop node list. (Fig. 1) However, choosing to use the node-disjoint multipath routing method with the secure dispersed data transfer method only counteracts the bottlenecking of paths, it is insufficient in countering eavesdropping due to overlap broadcast ranges.

To cope with eavesdropping, we also have proposed node-disjoint multiple paths which take into consideration the radio area overlap [7]. Fig. 2 illustrates a part of the pathfinding process in our previous paper [7] and the control message flow. The details of control messages will be described later. Since this method utilizes a mechanism which re-broadcasts the "Notification message," the method can generate greater numbers of control packets as the number of hops between a source node and a destination node becomes greater. Additionally, large control packet counts cause a longer waiting time and consume network bandwidth in the pathfinding process. Therefore, we have to develop countermeasures to eavesdropping attacks that utilize a small number of control packets and achieve a shorter latency for wide applicability.

In this paper, we propose a new multipath routing method to reduce radio area overlap, reduce latency, and eliminate excess control packets through the introduction of control packet overhearing in the pathfinding process. In our proposed method, we introduce control packet overhearing mechanisms to eliminate excess control packets and latency in the pathfinding process. We also implemented our proposed method on a network simulator, Qualnet version 5.0 [8], and evaluated the results.

The rest of this paper is organized as follows: in Section 2, we explain related works. In Section

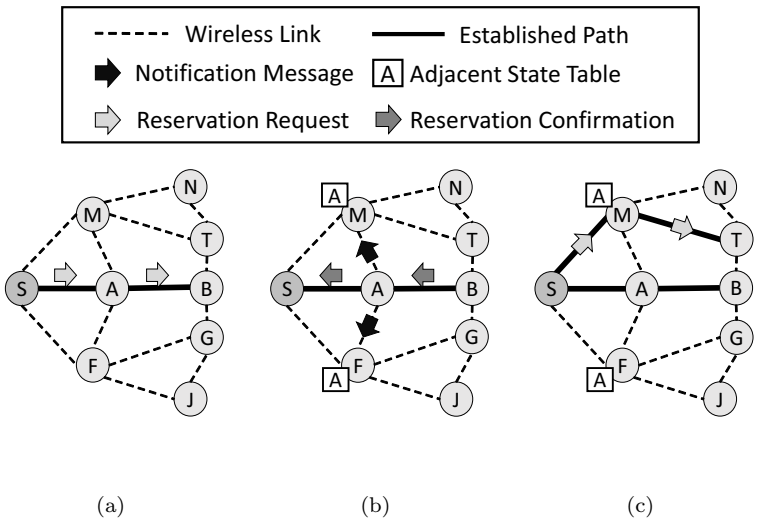


Figure 1: Pathfinding process in the existing method to avoid bottlenecking [5] [6].

3, we describe our proposed method. In Section 4, we illustrate and discuss our experimental results. Section 5 is devoted to the conclusions we arrived at through our analysis.

2 Related Works

2.1 The threat model of node capture attacks as defined in this paper

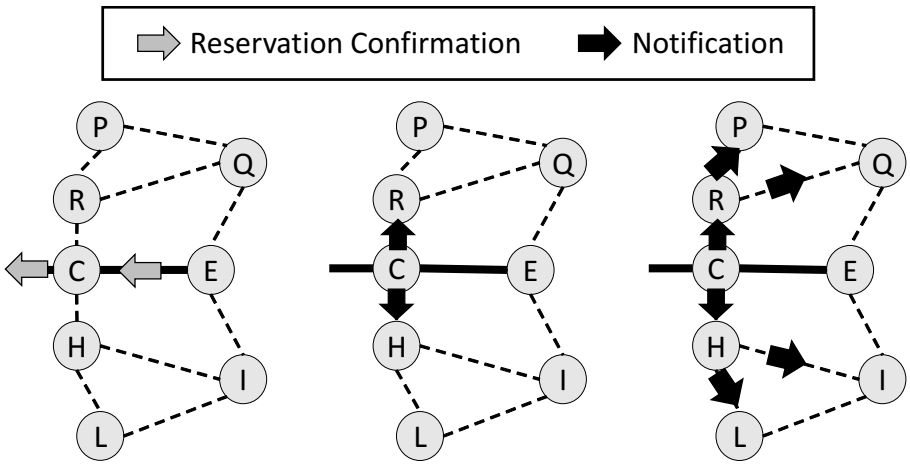
The nodes in ad hoc networks use radio links to communicate with each other. To allow such communication, nodes on ad hoc networks generally remain exposed to the networked environment. As a result, ad hoc networks are susceptible to many kinds of attacks. This section describes the kinds of attacks simulated in our experiments.

Zhang et al. [9] listed three kinds of attacks on wireless networks: compromising, eavesdropping, and node insertion. In this section, we describe two attacks: compromising and eavesdropping. In this paper, we will focus on compromising and eavesdropping attacks due to the interconnected nature of these attacks. In general, the aforementioned attacks can be prevented by various cryptographic methods using a common key-based system or public key-based system [10]. However, such systems are comparatively weak against node compromise because once one key is stolen, the entire system is penetrable. In a compromising attack, the adversaries steal forwarded data and keys inside a node. In eavesdropping attacks, adversaries access a node's OS through superuser privileges to eavesdrop on conversations taking place around the node. This is done by capturing the signal from the attacked node's wireless devices.

2.2 Outline of the secure dispersed data transfer method

For ad hoc networks, we have proposed using the secure dispersed data transfer method to handle confidential data transmissions without secret or public key systems. Fig. 3 shows a conceptual diagram of the secure dispersed data transfer method. As shown in Fig. 3(a), a source node creates shares from the original data by using the (k, n) threshold scheme [11] over a Galois field $GF(2^m)$ [12]. In the (k, n) threshold scheme, k and n are the threshold number and the number of shares, respectively. k and n are positive integers and n must be greater than or equal to k. When we encrypt the original information, say S, n versions of encrypted data packets will be created from S. For the encryption, we calculate shares using the following equation (1):

$$f(x) = S + a_1 x + a_2 x^2 + \cdots + a_{k-1} x^{k-1} \tag{1}$$



(a) Reservation Confirmation.

(b) Notification broadcast.

(c) Notification "re-broadcast".

Figure 2: Pathfinding process in paper [7].

where a_i $(i = 1, 2, \cdots, k-1)$ are random integers. We obtain each share $(u_i,\ v_i)$ $(i = 1, 2, \cdots, n)$ by substitution of the value u_i $(i = 1, 2, \cdots, n)$ for $f(x)$. A destination node attempts to decrypt the original data from the received shares. When we decrypt the original data, we need a set of at least k shares to decrypt the original data. We can decrypt the original data using Lagrange's interpolation method as in equations (2) and (3).

$$S = \lambda_1 v_1 + \lambda_2 v_2 + \cdots + \lambda_k v_k \tag{2}$$

where

$$\lambda_j = \prod_{i=1, i \neq j}^{k} \frac{u_i}{(u_i - u_j)}. \tag{3}$$

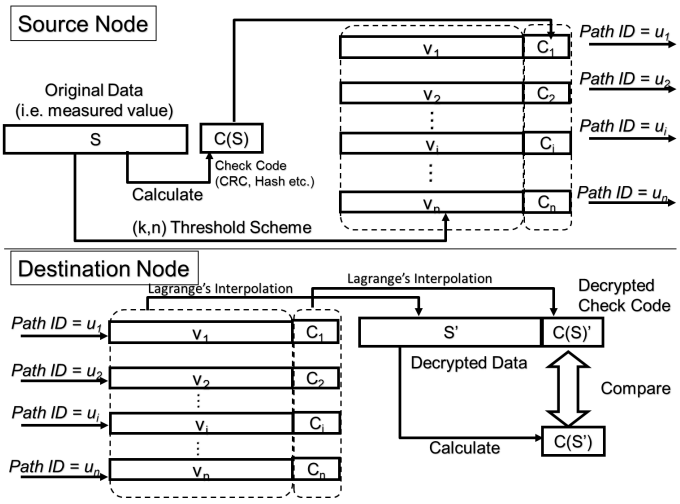
Fig. 3(b) shows the transmitted shares between a source node and a destination node with multiple paths A, B, and C. In this way, the original data can be transferred securely [3].

The secure dispersed data transfer method utilizes "light-weight" calculations such as the secret sharing scheme (the polynomial-based threshold scheme), and Lagrange's interpolation method (equations (2) and (3)). As a result, the secure dispersed data transfer method can be applied to devices with limited computational power.

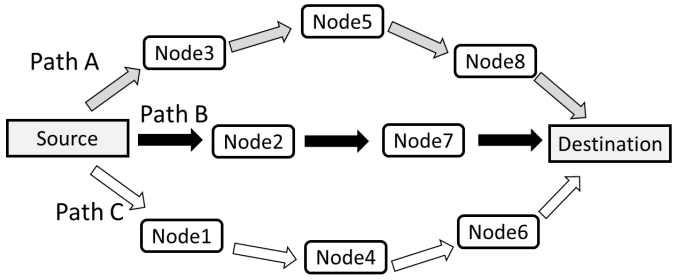
2.3 The node-disjoint multipath routing method for the prevention of theft keys/forwarded data through node capture attacks

As we mentioned in Section 1, since the secure dispersed data transfer method utilizes the secret sharing scheme, it can be used securely despite data loss. The secure dispersed data transfer method creates multiple encrypted versions of the original data packet (referred to as "shares") to protect the data. Therefore, the number of transferred packets is greater than the original number of packets. To compensate for this, we had to develop multipath routing methods with small control packet counts. We have previously developed two types of multipath routing methods that employ source initiated routing-ID routing (SRIDR) [13] [14] [15]. For convenience, the node-disjoint multipath routing method in paper [5] [6] is referred to as the "Existing method," in this paper.

SRIDR has been developed to construct local detour paths by using the DART [16] [17] [18]-based dynamic address routing ID assignment process. In Kohno et al.'s paper [4], we have shown that SRIDR can improve dependability when facing packet loss without degradation of security.



(a) Encryption and decryption [3].



(b) Transfer of encrypted data packets.

Figure 3: Conceptual diagram of the secure dispersed data transfer method.

2.4 Source initiated tree-based Routing ID Routing (SRIDR)

Like the existing method shown in Section 2.3, our proposed method has been developed based on SRIDR. Therefore, we illustrate the behavior and the characteristics of SRIDR below.

SRIDR is a multipath routing method, which utilizes binary tree-based routing ID, for static ad hoc networks. A routing ID consists of binary numbers with l bits and it is assigned uniquely to each node. Figure 4 shows an example of routing IDs assigned to a network. In Figure 4, each node is labeled with a routing ID. The solid and dotted lines illustrate direct links between two nodes. The solid link indicates a parent-child relationship for the routing ID assignment. When two nodes have a parent-child relationship, one bit of their routing IDs will be different. After the routing ID assignment process, subnets, which are illustrated in the dashed-dotted enclosure as shown in Figure 4, appear. A subnet of SRIDR is represented using 0, 1, or X (don't-care bit). The subnet, [00X], indicates the set of nodes [000] and [001]. Since SRIDR constructs the node's routing table, it can suppress the control packet counts. The routing table has the following entries.

- Subnet ID
- Next-hop node address
- Hop count to destination subnet

Figure 5 shows an example of the routing tables for nodes A and C. When node A transfers data packets to subnet [1XX], which includes nodes B, D, and G, node A checks its routing table and

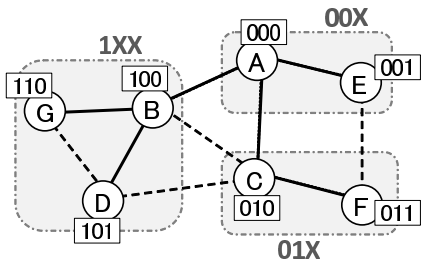


Figure 4: An example of assigned routing IDs to a network.

Node A

Subnet	Next-hop	Hop count
1XX	B	1
1XX	C	2
01X	C	1
01X	E	2
01X	B	2
001	E	1

Node C

Subnet	Next-hop	Hop count
1XX	D	1
1XX	B	1
1XX	A	2
00X	A	1
00X	B	2
011	F	1

Figure 5: An example of the SRIDR routing tables.

forwards data packets to nodes B or C. When node A transfers multiple data packets to node G using multiple paths, it can select two choices, one choice is paths A–B–G and A–C–D–G, or the other choice is paths A–B–G and A–C–B–G. While paths in the former choice are node-disjoint, paths in the latter choice are not (node B is shared by two paths).

Our proposed method constructs desirable multipaths as shown in Figure 6(b) by using SRIDR's routing table. In Section 3, we describe the process of our proposed method, in detail.

2.5 Resilience against eavesdropping attacks on simple node-disjoint multipath routing method systems

The aforementioned existing method [5] [6] was developed to prevent key/data packet theft on relaying nodes. Therefore, the existing method can limit the amount of theft. However, the existing method is still weak against eavesdropping attacks on transferred shares, which are the encrypted versions of the original data packet. When shares are transferred along established node-disjoint paths by the existing method [5] [6], radio signals can overlap on adjacent nodes among node-disjoint paths. Because the existing method does not consider the radio area overlap among transferred shares/paths, it is conducive to eavesdropping. To counter eavesdropping attacks, we must take into account the overlap of radio areas between paths, and we must introduce mechanisms to detect and avoid the possibility of radio area overlap.

2.6 Multipath routing methods for security

As we mentioned in Section 1, our proposed method is a countermeasure for eavesdropping attacks. Our proposed method also constructs multiple paths with a small radio overlap area for the secure dispersed data transfer method that is described in Section 2.2. In this section, we describe related works by other researchers on multipath routing methods and security.

Generally, many routing methods on ad hoc networks have been proposed and categorized [19][20][21][22]. According to papers [21][22], the multipath routing methods have been researched to counter unstable wireless links in ad hoc networks. Recently, the multipath routing methods have been proposed for countermeasures for security issues in ad hoc networks [19][20].

In paper [23], Lee et al. proposed the radio-disjoint multipath routing method. In this method, all nodes store the geographical location information of the source and the destinations. By using the geographical location information, the method can construct optimal paths to reduce radio-disjont multiple paths. In this method, however, the attackers can acquire the geographical location information of the source node and the destination node, and determine the best point to attack. Therefore, this method cannot be used for a countermeasure for eavesdropping attacks described in this paper.

In paper [24], Park et al. proposed the cross-layered multipath AODV (CM-AODV). In this method, they modified the route request (RREQ) packet to contain the "Route Quality" field to indicate signal-to-interference plus noise ratio (SINR). CM-AODV establishes its multiple paths based on the SINR value in RREQ. CM-AODV utilizes flooding to discover its paths. The network-wide flooding, however, can generate many control packets like AODV does. Many control packets increase latency and interfere with node broadcasts which makes the network less efficient at findings a path to the destination node. Additionally, bottlenecking and overlapping cannot be avoided in CM-AODV because the method did not attempt to address these problems. In contrast, since our proposed method has been developed based on SRIDR, it can find global or local detour paths without network-wide flooding while attempting to avoid overlapping.

Multipath optimized link state routing (MP-OLSR [25]) has been proposed by Yi et.al. based on OLSR (Optimized Link State Routing Protocol) [26]. In MP-OLSR, every node acquires the global topologies of a network, and they calculate multiple paths by using Dijkstra's shortest path algorithm [27] repeatedly. Uemori et.al. [5] proposed that the modified MP-OLSR method can construct node-disjoint multiple paths. In Section 4.3, we conducted simulation experiments to measure control packet counts between the MP-OLSR based method and our proposed method.

3 Proposed Method

3.1 Overview

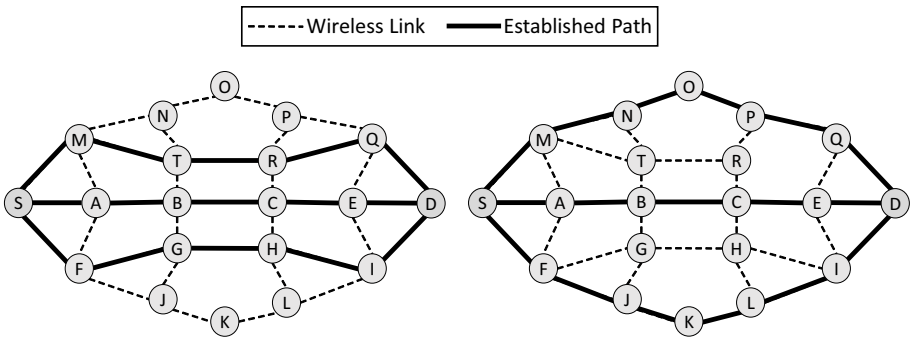
In this paper, we propose a new node-disjoint multipath routing method to counter eavesdropping attacks on transferred data packets in order to alleviate the problem of overlap. We designed our proposed method [28] based on the node-disjoint multipath routing method in papers [6] and [5]. As stated earlier, this method is referred to as the "Existing method." Additionally, our proposed method shares basic concepts with the method in paper [7]. In this paper, this method is referred to as the "notification rebroadcasting method." Hereinafter, we describe our proposed method. While many parts of the descriptions are similar to the notification rebroadcasting method, we also describe the differences between the notification rebroadcasting method and our proposed method in Section 3.3.

Since the existing method only addresses bottlenecking in multiple paths, radio areas of multiple paths which are constructed by the existing method can widely overlap. To counter the overlap, we have to consider the radio area interference of each path. Fig. 6 depicts example networks with node-disjoint multiple paths which are constructed by the existing method as well as our proposed method and the notification rebroadcasting method.

In our proposed method and the notification rebroadcasting method, notification messages are introduced to established paths with smaller overlap radio areas. (Fig. 7)

In all of the aforementioned methods (the existing method, the notification rebroadcasting method, and our proposed method), when a node is selected as a member of a certain path, the node receives a reservation confirmation (RC) message from the next hop node as shown in Fig. 8.

While RC messages are sent by unicast in a wireless network, all the surrounding nodes (ex. node T, R, G, H in Fig. 8) can receive the radio signals. In our proposed method, when a node sends an RC message, it does not have to send notification messages to its surrounding nodes because its



(a) Existing method [5] [6].

(b) Proposed method and notification rebroadcasting method [7].

Figure 6: Example of node-disjoint multiple paths.

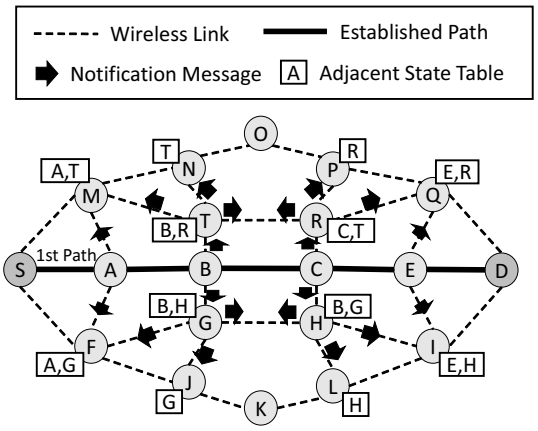


Figure 7: Notification message in notification rebroadcasting method.

surrounding nodes are actively overhearing the communications. When every node in the network is actively overhearing, fewer notification messages need to be sent and latency is decreased.

3.2 Algorithms

In the routing procedure, the adjacent nodes of the source node attempt to find their own single paths. Additionally, the intermediate nodes coordinate to prevent duplication of paths. The routing procedure is performed in sequence from the source node to destination node. When a node cannot find a node-disjoint path, the backtracking sub-procedure is invoked. With the backtracking sub-procedure, the node sends control messages along the reverse path. The node receives the control messages that re-discover the node-disjoint path. To execute those actions, we have introduced four control messages: (1) Reservation request message, (2) Reservation denial message, (3) Reservation confirmation message, and (4) Notification message. We also have introduced three node states: (a) Default, (b) Candidate, and (c) Determined.

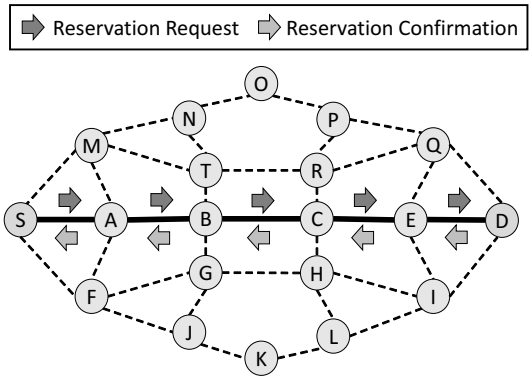


Figure 8: Constructing a node-disjoint path.

3.2.1 Control messages

(1) Reservation request (RR) message This message is sent from the source node or an intermediate node to its neighboring node to reserve a node as a member of a path. Each path between a node adjacent to the source node and destination node is found.

(2) Reservation denial (RD) message When the node cannot join the path, this message is sent back from the node which receives the RR to the originator of the RR message.

(3) Reservation confirmation (RC) message When the node can join the path, this message is sent back from the nodes that received RR to the originator of the RR message.

(4) Notification message This message is broadcasted from the node which sent the RR messages to its neighboring nodes and was selected to join a path. This node sends notification messages so that its adjacent nodes do not select it as the next-hop node for different path.

3.2.2 Node states

(a) Default This state shows that a node is not selected as a member of the path between the source and the destination. In its initial status, all nodes except the source node are in the default state.

(b) Candidate This state shows a node is reserved as a potential member of the path between the source and the destination. Since a node in the candidate state is not determined firmly as a member of the path, this node can change entries in its multipath routing table if necessary.

(c) Determined This state shows a node is determined as a member of the path between the source and the destination. In its initial status, the source node is in the determined state. Once a node in the determined state it is a member of the path, and the node cannot change entries in its multipath routing table.

3.2.3 The proposed routing table and the adjacent state table

The proposed routing table has the following entries.

- Destination node address
- Source node address

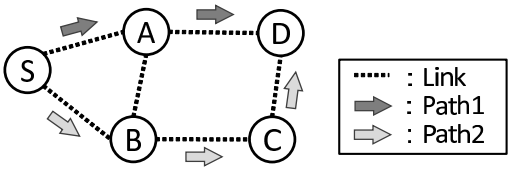


Figure 9: An example network

Node S

Src	Dst	Previous	Next
S	D	-	A
S	D	-	B

Node A

Src	Dst	Previous	Next
S	D	S	D

Node B

Src	Dst	Previous	Next
S	D	S	C

Node C

Src	Dst	Previous	Next
S	D	B	D

Node D

Src	Dst	Previous	Next
S	D	A	-
S	D	C	-

Figure 10: The proposed routing tables

- Previous-hop node address
- Next-hop node address

Figure 9 shows an example network where the source node is labeled as S, and the destination node is labeled as D. Figure 10 shows the proposed routing tables, which correspond to the network topology in Figure 9. As shown in Figure 10, only the source node and the destination node have multiple entries in their routing table. Meanwhile, the intermediate nodes between the source and the destination have only one routing entry.

Our proposed method constructs multipaths with the small radio overlap area by using both RC messages and notification messages as shown in Figure 14. At that time, we introduced the adjacent state table within a node in order to acquire the effect as shown in Figure 7. When a node's ID has been registered in the adjacent state table, the owner of the adjacent state table will not select the registered node for a member of another path.

3.2.4 The routing procedure

In the routing procedure, the behavior is different depending on the type of nodes. Henceforth, we describe the procedures for (1) the source nodes, (2) the intermediate nodes, and (3) the destination nodes.

1) The source nodes Figure 11 shows the flow chart of the procedure on the source node.

The source node unicasts a RC message to its adjacent nodes, and waits for the RC or RD message for a predefined time. When the source node receives the RC message, it acquires the results of the path discovery. If the path is discovered successfully, the source node registers the entry, which includes the source and the destination nodes on the multipath table. These procedures are performed periodically until the requested number's paths are found. The data packets are transferred along paths using the multipath table. If the number of found paths is smaller than the requested number of paths, the source node does not send the (original) data packets.

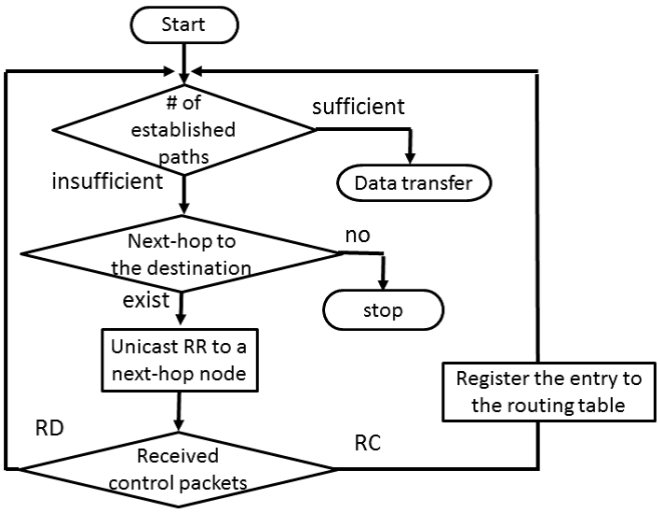


Figure 11: Procedure of a source node.

2) The intermediate nodes When the intermediate node receives/overhears a control message, the behavior of the intermediate node is different depending the type of the control messages

Figure 12 shows the flow chart of the procedure of the intermediate nodes.

A) RR message When the intermediate node receives an RR message, the behavior of the intermediate node is different depending on its state.

When an intermediate node's state is in default, it selects the entry which includes the destination node and a next-hop node, and sends an RR message. At this time, the intermediate node registers the destination node and the next-hop node information. It also changes its own state to the candidate state. After that, it broadcasts the notification message to its adjacent nodes. When an intermediate has no entry which includes a next-hop node on the path to the destination node, it sends an RD message to its previous-hop node.

B) RC message When the intermediate node receives the RC message, it changes its own state to the determined state and unicasts the RC message to the previous hop node.

C) RD message When an intermediate node is in the candidate or determined state, it sends the RD message to its previous-hop node by unicast.

When the intermediate node receives an RD message as a response to an RR message, it registers the RD message sender information on the adjacent node state table. In the case that an intermediate node receives an RD message, it judges that the previous RR message is invalid in discovering a node-disjoint path. Therefore, it searches for another path to be the node-disjoint path. To execute this action, the intermediate node searches for another next-hop node on the SRIDR routing table. When it finds another next-hop node, it updates the multipath routing table entry and it sends an RR message by unicast as was the case in the first search for the node-disjoint path. When it can no longer find another next-hop node, it unicasts an RD message to the previous-hop node. After that, it erases the corresponding entry on the multipath routing table. Additionally, it changes its own state to the default state. After that, it broadcasts the notification message to its adjacent nodes.

D) RC message by overhearing When the intermediate node overhears the RC message, it registers the routing ID of the RC message sender node to its adjacent state table.

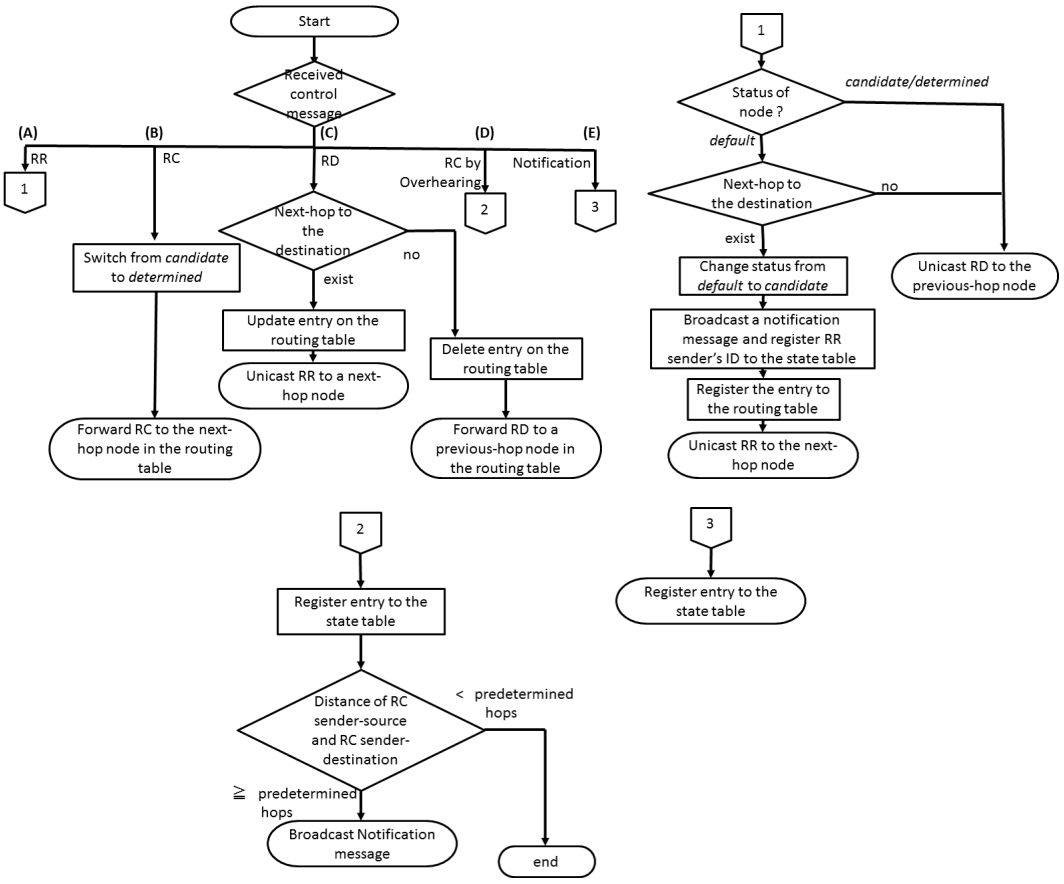


Figure 12: Procedure of intermediate nodes.

After that, it judges the distance between the RC sender node and the source node and the distance between the RC sender node and the destination node. When the distances are predetermined hops or more, it broadcasts a notification message to its adjacent nodes.

E) Notification message

When the intermediate node receives the notification message, it registers the routing ID of the notification message sender node to its adjacent state table.

3) The destination nodes When the destination node receives an RR message, it registers a new entry, which includes the routing ID that came with the RR message. When the destination node is in the default state, the destination node changes its own state to candidate, and it waits a predefined time. When the predefined time expires, the destination node changes its own state to the determined state. Subsequently, it acquires the next-hop node address, and unicasts a RC message to that address. When the destination node has multiple next-hop nodes, it unicasts RC messages in the received order of the RR messages.

Figure 13 shows the flow chart of the procedure on the destination nodes.

3.3 Overhearing of notification messages

As we mentioned in Sections 2 and 3.1, the notification rebroadcasting method attempts to construct reduced radio area overlap multiple paths to counter the eavesdropping attacks on transferred data

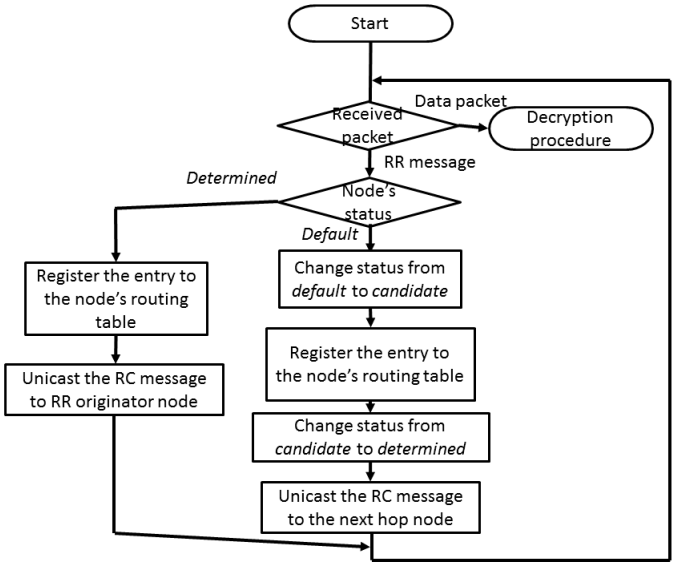


Figure 13: Procedure of destination node.

packets by using notification message rebroadcasting. In this section, we describe our proposed method by using Fig. 14. Fig. 14(a) depicts node X was selected as a path from node A. The large thick circle in Fig. 14(a) shows the radio area of node X. Short gray arrows show the RC messages. In our proposed method, several control messages are sent in the following sequence: (1) When node A selects node X as a member node of the path, node A send an RC message to node X; (2) Node X re-sends the RC message to its neighboring nodes; (3) Nodes C, E, F and G overhear node X's RC message and they determine that node X was selected as a member of the path; (4) Nodes C, E, F and G broadcast their notification messages to their surrounding nodes as shown in Fig. 14(b); (5) Nodes that received the notification messages from nodes C, E, F and G eliminate the entries of nodes C, E, F and G in their adjacent node list. The elimination of the entries, in the adjacent node list, effectively erases the nodes from the network and creates a wireless gap that reduces the susceptibility of eavesdropping.

4 Experiments and Disussions

In order to evaluate our proposed method, we have implemented it on a simulator, QualNet version 5.0 [8], and have conducted experiments. We describe evaluated items as follows:

Data packet delivery ratio

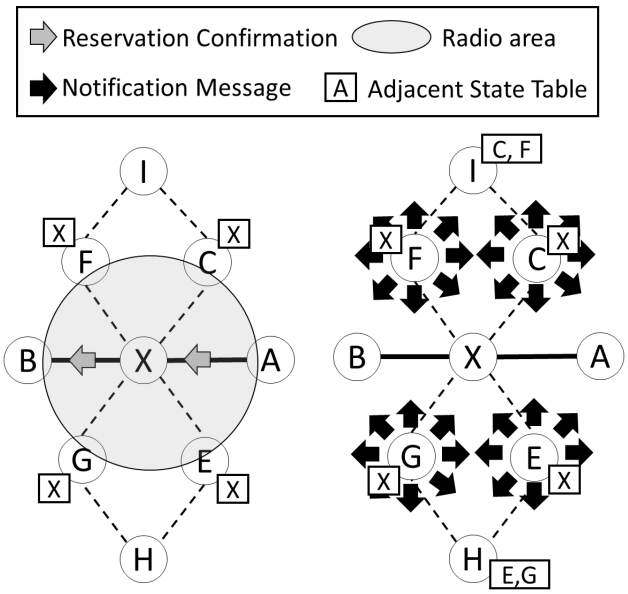
This value is the percentage of the number of successfully decrypted packets on the destination nodes divided by the number of the sent packets from the source nodes. The data packet delivery ratio is an average number per one pair.

Percentage of acquired original data packets

In our experiments, we measured the number of data packets of all nodes except the source and destination nodes. We also calculated the percentage of the amount of the original data packets that had been acquired divided by the number of nodes times the total number of the original data packets that were sent. This value is shown as an average for one pair of source and destination nodes.

Number of exploitable nodes

The number of exploitable nodes is the number of nodes that had decrypted and acquired one



(a) (b)

Figure 14: Overhearing and notification.

or more original data packets by using their eavesdropped shares. This value is an average value per one pair.

Total amount of control packets
This value is the total amount of all control packets needed, except control packets of SRIDR, to construct the desired node-disjoint multiple paths.

Latency of the pathfinding process
This value is the latency to construct the desired node-disjoint multiple paths.

Number of hops
This value is the average hop count between source and destination nodes.

End-to-end delay
This value is the delay from when a source node sent the first shares to its destination node and when the original data packet is subsequently decrypted by the destination node.

Paths construction rate
This value is the percentage of the number of the source and destination pairs which can construct three or more node-disjoint multiple paths divided by the number of all the source and destination pairs. For the "Existing 1 and over," which we will explain later, we substituted the percentage of the number of the source and destination pairs which can construct one or more node-disjoint multiple paths for the numerator.

4.1 Simulation configuration

4.1.1 Method

In the experiments, since we evaluated the performance of static ad hoc networks, we set the mobility of nodes to zero. We prepared six types of configurations with an average of 6, 8, 10, 12, 14 and 16

Table 1: PARAMETERS FOR EXPERIMENTS

Simulator	QualNet ver 5.0 [8]
Field size(m^2)	2215x2215
Node distribution	Random
Number of nodes	200, 250, 300
Average number of adjacent nodes	6, 8, 10, 12, 14, 16
Number of S-D pairs	10
Number of data packets	10
Intervals of data packets (s)	1
Radio area (m)	250
Transport layer protocol	UDP
Simulation time (s)	250
Galois field GF(2^m) [12]	GF(2^8)
Irreducible polynomial of GF(2^m)	$x^8 + x^7 + x^2 + \mathrm{x} + 1$
Number of simulation runs	50

adjacent nodes, respectively. The total number of network nodes was 150, 200, 250, 300, 350 and 400, respectively. The source and destination pairs were 10. The source sent 10 original (secret) data packets per one pair. The MAC layer protocol was IEEE802.11b and the maximum bandwidth of the protocol was 11 (Mbps). The number of simulation runs was 50. Table 1 shows the parameters in our simulation experiments.

For comparison, we employed the node-disjoint multipath method of papers [5] [6] as the existing method. In the existing method, a source node sends its shares when one or more ("Existing 1 and over") and three or more ("Existing 3 and over") node-disjoint multiple paths are constructed. At this time, the number of shares and the threshold number are three and two, respectively. These configurations in the secure dispersed data transfer method are the minimum threshold numbers required in node-disjoint multiple paths.

In the notification rebroadcasting method, a node forwards a received notification message when the node is apart from the source and destination nodes by four or more hops. Destination nodes attempt decryption utilizing their received shares. When decryption is successful, we calculate the destination node as having received one data packet. In the same manner, in our proposed method, a node overhears RC messages when the node is apart from the source and destination nodes by four or more hops. Destination nodes attempt decryption utilizing their received shares. When decryption is successful, we calculate the destination node as having received one data packet. Hereinafter, we denoted the results of the notification rebroadcasting method as "Notification-Notification," and the results of our proposed method as "Overhearing-Notification."

4.1.2 Simulating eavesdropping attacks

All nodes except the source and destination nodes acted as if they were attacked nodes. When the source nodes transmitted, every other node, except the destination node, records and stores the eavesdropped and accepted shares. This simulates eavesdropping. After every transmission, each node attempts to decrypt the original data packets using previously recorded shares in each node. To simulate the attack node, we used the same decrypting algorithm as the destination node. When a node can decrypt one original data packet, we counted it as such. In this situation, since all of the nodes except the source and destination nodes are attack nodes, it is an extremely hostile environment for networks. This simulation is how we measured the security of our proposed method. After each run, we measured the average number of data packets acquired by eavesdropping nodes. The lower the percentage of acquired data packets, the more secure the configuration.

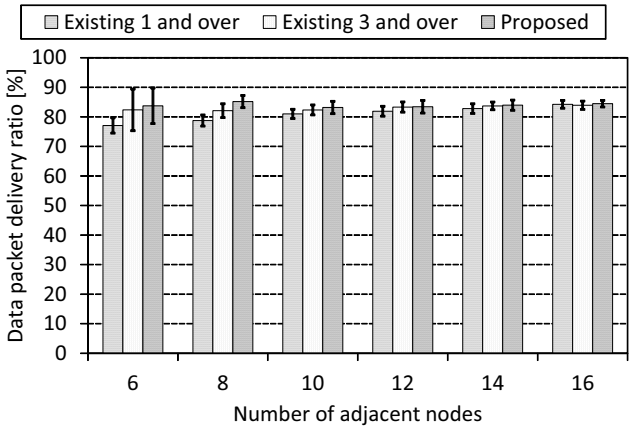


Figure 15: Data packet delivery ratio versus average number of adjacent nodes.

4.2 Experimental results

4.2.1 Data packet delivery ratio

Fig. 15 show the data packet delivery ratio. The horizontal axis of Fig. 15 is the average number of adjacent nodes. The error bars are 90 percent confidence intervals.

Data packet delivery ratio

In Fig. 15, the data packet delivery ratio for both the "Existing 3 and over" and our proposed method are almost identical in every average number of adjacent nodes. However, the data packet delivery ratio of the "Existing 1 and over" is about 6.4-6.6 % smaller than that of the others in cases where the number of adjacent nodes was between 6 and 8. In the "Existing 1 and over," the source nodes send data packets even when there are only one or two constructed paths. This indicates that since transmitted shares are colliding, the original data packets cannot be transmitted completely.

4.2.2 Resilience for eavesdropping

Figs. 16(a) and 16(b) show the percentage of acquired original data packets by eavesdropping nodes and the number of exploitable nodes, respectively. The horizontal axis of Fig. 16 is the average number of adjacent nodes. The error bars are 90 percent confidence intervals.

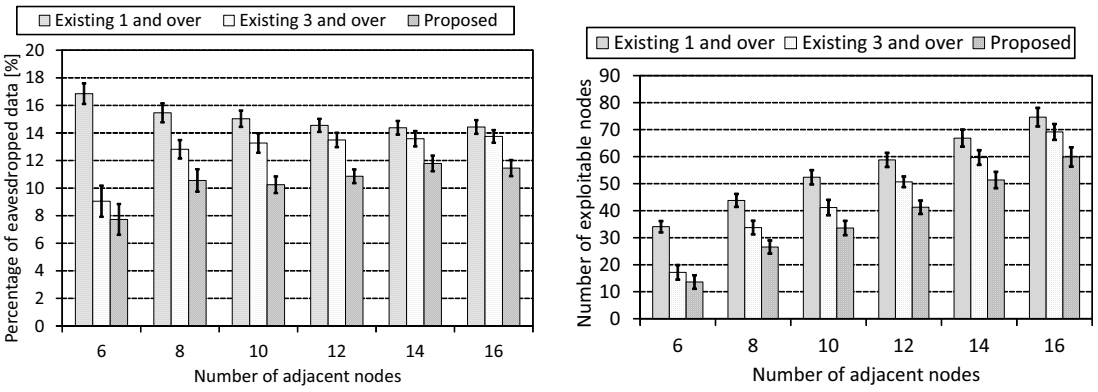
Percentage of acquired original data packets by eavesdropping nodes

In Fig. 16(a), the percentage of data packets acquired by eavesdropping attacks for both the " Existing 3 and over" and our proposed method increased as the node density increased. As the node density increased, the number of nodes surrounding the paths' radio area was increased. The number of exploitable nodes that collect more than the threshold number of shares increased. In contrast, the percentage of the original data packets acquired by the eavesdropping nodes of "Existing 1 and over" decreased as the average number of adjacent nodes increased. This is because as the node density increased, the number of the source and destination pairs that constructed three or more multiple paths increased.

The percentage of data packets acquired by eavesdropping in our proposed method is the smallest among the three methods. It is 2.59-9.12 points less than that in "Existing 1 and over" and is 1.32-3.02 points less than that in "Existing 3 and over."

Number of exploitable nodes

As shown in Fig. 16(b), the number of of exploitable nodes increased as the average number of adjacent nodes increased. In the "Existing 1 and over" configuration of the existing method,



(a) Percentage of acquired original data packets by eavesdropping nodes.

(b) Number of exploitable nodes

Figure 16: Resilience for eavesdropping.

since shares were sent only when one or more path(s) had been established, many nodes can collect and decrypt the original data packets. The number of exploitable nodes of our proposed method decreased 14.7-20.4 nodes than that of "Existing 1 and over," and 3.5-9.3 nodes than that of "Existing 3 and over."

4.2.3 The amount of control packets and latency

Figs. 17(a), 17(b), 17(c) and 17(d) show the total amount of control packets, the latency of the pathfinding process, the number of hops and the end-to-end delay, respectively. The horizontal axis of Fig. 17 is the average number of adjacent nodes. The error bars are 90 percent confidence intervals.

Amount of control packets

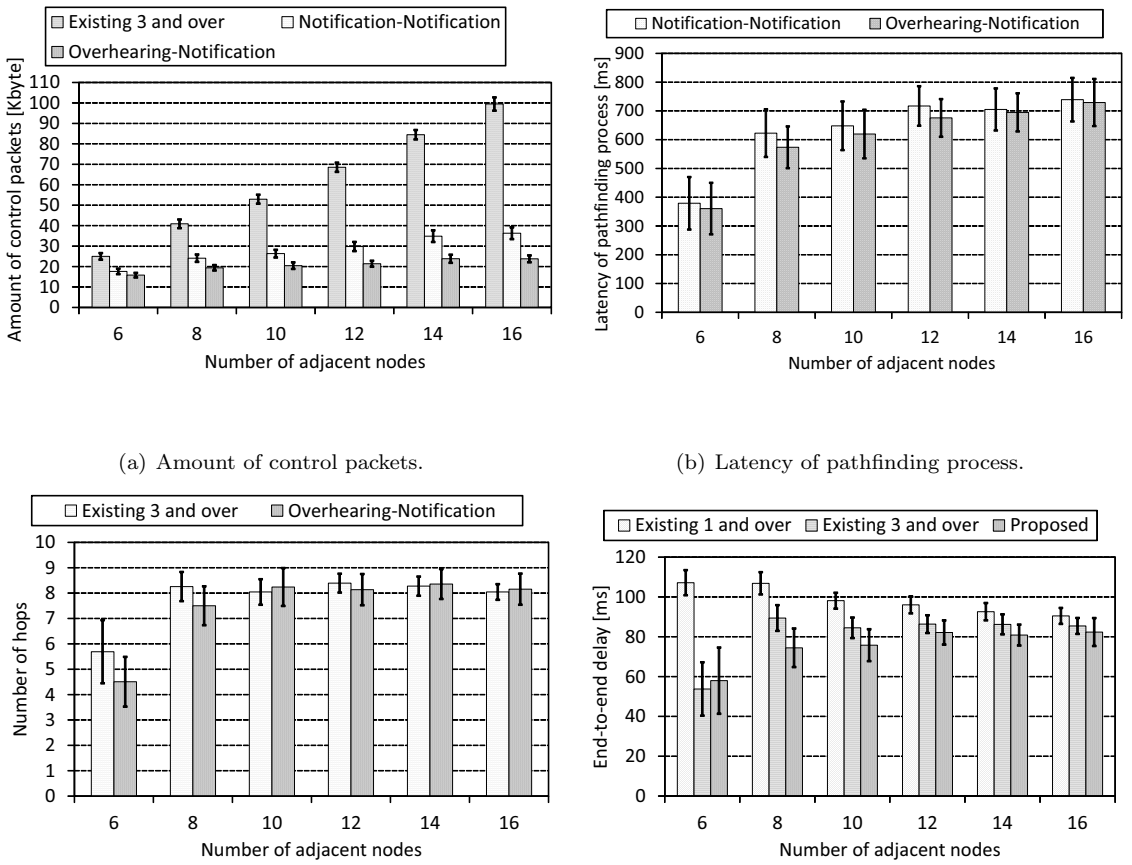
In Fig. 17(a), the total amount of control packets dramatically increased as the average number of adjacent nodes increased in "Existing 3 and over." Since the "Existing 3 and over" method constructs multiple paths concurrently, the number of pathfinding nodes increased. Therefore, the control packet count increased. The amount of control packets in the "Notification-Notification" method increased. This is because the number of relaying nodes had increased and the number of the notification messages had increased. On the other hand, the "Overhearing-Notification" method attempted to decrease the number of the notification messages by using overheard RC messages. As a result, the total amount of control messages is 36.7-76.0 % less than that of "Existing 3 and over," and 10.2-34.5 % less than that of the "Notification-Notification" method, respectively.

Latency of pathfinding process

In Fig. 17(b), latency in the pathfinding process of the our proposed method ("Overhearing-Notification") is 9.8-49.2 (ms) shorter than that of the notification rebroadcast method. As we described in Section 3.3, since our proposed method eliminates the excess notification messages of the notification rebroadcasting method, our proposed method can establish a path faster than that of the notification rebroadcast method. To improve performance, the overhearing mechanism of other nodes was introduced in the dynamic source routing method (DSR) [29] [30]. In our proposed method, the overhearing mechanism worked successfully.

Number of hops

As shown in Fig. 17(c), the number of hops of our proposed method and "Existing 3 and over"



(a) Amount of control packets.

(b) Latency of pathfinding process.

(c) Number of hops.

(d) End-to-end delay.

Figure 17: Influence of proposed method.

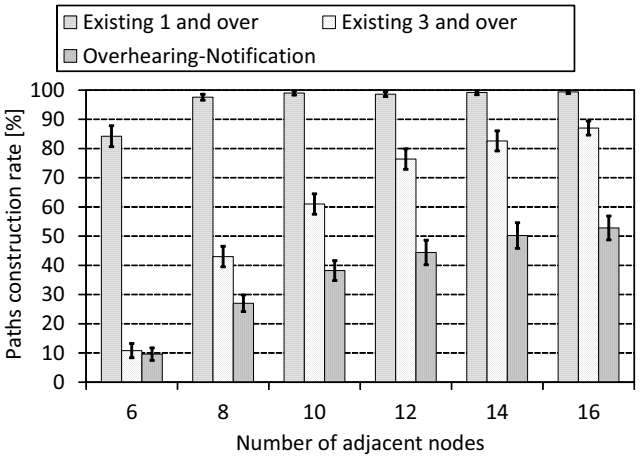
is almost the same. In the existing method, multiple paths are established simultaneously. When a node attempts to select a node as its next hop to find the shortest path, the node may already have been included in another path. At that time the pathfinding node selects other nodes. As a result, the pathfinding node cannot select its shortest path. In contrast, since our proposed method established multiple paths one by one, a node can establish its shortest path. Consequently, the number of hops of our proposed method and "Existing 3 and over" did not make a big difference.

End-to-end delay

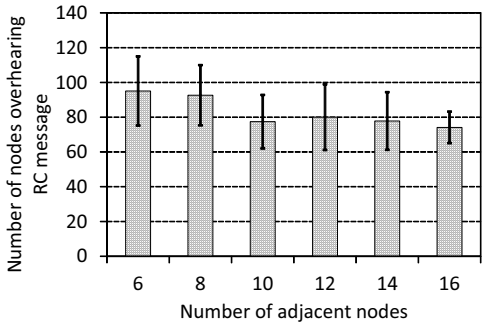
As shown in Fig. 17(d), the end-to-end delay of our proposed method is 8.1-49.2 (ms) faster than that of "Existing 1 and over" Moreover, the end-to-end delay of our proposed method in the average number of adjecent nodes at 8-16 is 3.0-14.9 (ms) faster than that of "Existing 3 and over." These results show our proposed method has little affect on the end-to-end delay.

4.2.4 Paths construction rate

Figs. 18(a) and 18(b) show the paths construction rate and the number of overhearing nodes, respectively. The horizontal axis of Fig. 18 is the average number of adjacent nodes. The error bars are 90 percent confidence intervals.



(a) Paths construction rate.



(b) Node overhearing RC message.

Figure 18: Paths construction rate and number of nodes overhearing RC messages.

Paths construction rate

As shown in Fig. 18(a), the paths construction rate increased as the average number of adjacent nodes increased. As the number of adjacent nodes increased, since the possible number of paths increased, the paths construction rate can be increased. In our proposed method, a source node should construct three or more paths with small radio overlap areas. This condition is more severe than the single path routing method. As shown on Fig. 18(a), the paths construction rate of our proposed method is 46.6-74.6 percent smaller than that of "Existing 1 and over" and is 1.2-34.2 percent smaller than that of "Existing 3 and over."

Our proposed method employs re-broadcasting the notification messages to prevent overlap between paths. To analyze the behavior of our proposed method, we have conducted additional simulation experiments with configurations as shown in Table 1. To analyze that, we have newly prepared extended RC messages with a field to indicate the number of paths. After the second pathfinding process has been completed, we will measure the number of nodes which have been eliminated in order to become relaying nodes for the second or the third paths. In our proposed method, a node that receives RC messages will be eliminated to become a member of the next path. Fig. 18(b) shows the number of nodes overhearing an RC message. As shown in Fig. 18(b), our proposed method eliminated 74-95 nodes in order to become relaying nodes for

the second or the third paths. Thus, in some network configurations, networks cannot prepare a sufficient number of nodes. This is the reason for the wide gap in the paths construction rate between our proposed method and the existing method.

4.3 Comparison to MP-OLSR

As we mentioned in Section 2.6, our proposed method has been developed based on SRIDR. In this section, we discuss the comparison to other methods.

Table 2: Simulation parameters to comparison to MP-OLSR.

Number of nodes	200, 300, 400, 500
Field size (m^2)	2215 × 2215
Node distribution	Random
Mobility (m/s)	0
Number of SD pairs	10
Application layer protocol	CBR
Data packet size (Byte)	512
Mac layer protocol	IEEE802.11
Maximum bandwidth(Mbps)	11
Radio area (m)	250
Simulation time (s)	150

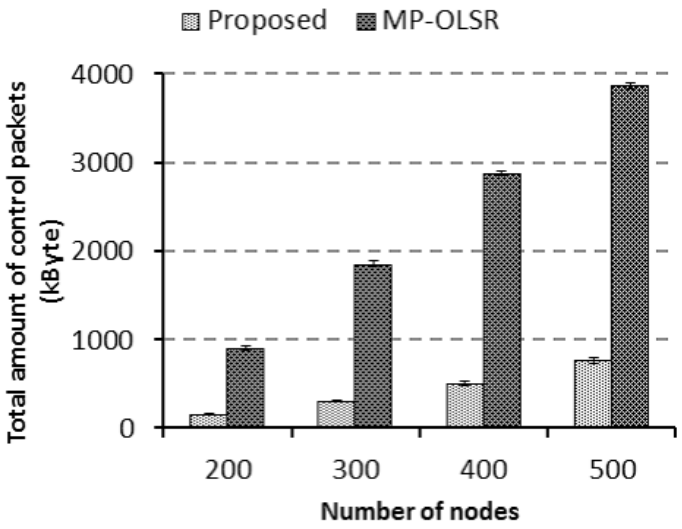


Figure 19: The amount of control packets between proposed method and MP-OLSR.

By using (dynamic) routing ID assignment, SRIDR can suppress the excess the flooding of control packets when it construct paths. Since our proposed method only uses a limited broadcast mechanism of notification messages, it can generate very few control packets. In order to confirm the amount of control packets, we have conducted experiments between MP-OLSR and our proposed

method. Table 2 shows the parameters of the simulation experiments. Fig. 19 shows the amount of control packets between our proposed method and MP-OLSR. The vertical axis and horizontal axis are the amount of control packets in (kByte) and the number of nodes, respectively. The error bars show a 95 % confidence interval. As shown in Fig. 19, the amount of control packets of our proposed method is 80.1-82.3 % fewer than that of MP-OLSR.

5 Conclusion

In this paper, we proposed a new node-disjoint multipath routing method to reduce overlap of radio areas in paths. In the notification rebroadcasting method, notification messages, which are a kind of control message, are rebroadcasted to reduce radio area overlap in multiple paths. In contrast, our proposed method introduces overhearing RC messages to reduce the control packet counts and the latency in the pathfinding process.

We also conducted simulation experiments to evaluate our proposed method. We observed that our proposed method can construct desired paths with a smaller amount of control packets and a shorter latency in the pathfinding process. Furthermore, we also conducted additional experiments to discuss the applicable scope of our proposed method. As a result, our proposed method is more effective as the average number of adjacent nodes increased.

In the future, we plan to develop a method to simultaneously find multiple paths. We have to create an extended method to take into account node density. Furthermore, we also have to consider more sophisticated eavesdropping attacks.

Acknowledgement

This research is supported by JSPS KAKENHI Grant Number (B) (No.24300028), and (C) (No.25330109). In addition, we would like to thank Mr. Ryuma TANI, Hiroshima City University for valuable discussions.

References

[1] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures," *Ad hoc networks*, vol. 1, no. 2-3, pp. 293–315, 2003.

[2] E. Kohno, T. Ohta, and Y. Kakuda, "Secure decentralized data transfer against node capture attacks for wireless sensor networks," in *Proc. 9th IEEE International Symposium on Autonomous Decentralized Systems (ISADS 2009)*, Mar. 2009, pp. 35–40.

[3] E. Kohno, T. Ohta, Y. Kakuda, and M. Aida, "Improvement of dependability against node capture attacks for wireless sensor networks," *IEICE Transactions on Information and Systems (IEICE/IEEE Joint Special Section on Autonomous Decentralized Systems Technologies and Their Application to Networked Systems)*, vol. E94-D, no. 1, pp. 19–26, Jan. 2011.

[4] E. Kohno, T. Okazaki, M. Takeuchi, T. Ohta, Y. Kakuda, and M. Aida, "Improvement of assurance including security for wireless sensor networks using dispersed data transmission," *Journal of Computer and System Sciences*, vol. 78, no. 6, pp. 1703–1715, Nov 2012.

[5] T. Uemori, E. Kohno, and Y. Kakuda, "A node-disjoint multipath scheme for secure dispersed data transfer in ad hoc networks," in *Proc. 2013 First International Symposium on Computing and Networking (CANDAR)*, Dec. 2013, pp. 441–447.

[6] ——, "A routing ID-based node-disjoint multipath scheme for ad hoc networks," in *Proc. 2012 9th International Conference on Ubiquitous Intelligence Computing and 9th International Conference on Autonomic Trusted Computing (UIC/ATC)*, Sep. 2012, pp. 621–626.

[7] T. Murakami, T. Kimura, T. Uemori, E. Kohno, and Y. Kakuda, "On notification message re-broadcasting for the node-disjoint multipath routing method in ad hoc networks to counter eavesdropping of data packets," in *Proc. 35th International Conference on Distributed Computing Systems Workshops (ICDCSW 2015)*, Jun. 2015, pp. 11–16.

[8] Scalable Network Technologies Inc., "Qualnet network simulator by scalable network technologies." [Online]. Available: http://www.scalable-networks.com/

[9] W. Zhang, S. K. Das, and Y. Liu, "Security in wireless sensor networks: A survey," in *Security in Sensor Networks*, Y. Xiao, Ed. Auerbach Publications, 2007, pp. 237–272.

[10] A.-S. K. Pathan, *Security of Self-Organizing Networks: MANET, WSN, WMN, VANET*, 1st ed. Boston, MA, USA: Auerbach Publications, 2010.

[11] A. Shamir, "How to share a secret," *Communications of the ACM*, vol. 22, no. 11, pp. 612–613, 1979.

[12] E. D. Karnin, J. W. Greene, and M. E. Hellman, "On secret sharing systems," *IEEE Transactions on Information Theory*, vol. IT-29, no. 1, pp. 35–41, Jan. 1983.

[13] T. Okazaki, E. Kohno, T. Ohta, and Y. Kakuda, "A multipath routing method with dynamic ID for reduction of routing load in ad hoc networks." in *ADHOCNETS'10*, ser. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, J. Zheng, D. Simplot-Ryl, and V. C. M. Leung, Eds., vol. 49, no. 2, 2010, pp. 114–129.

[14] T. Okazaki, E. Kohno, and Y. Kakuda, "Improvement of assurance for wireless sensor networks using packet detouring and dispersed data transmission," in *Proc. 2011 IEEE International Conference on Internet of Things and Cyber, Physical and Social Computing (iThings/CPSCom 2011)*, 2011, pp. 144–151.

[15] T. Okazaki, M. Takeuchi, E. Kohno, and Y. Kakuda, "Self-organized routing ID tree-based multipath construction for ad hoc networks," *Proceedings of the Fifteenth IEEE Computer Society symposium on object/component/service-oriented realtime distributed computing (ISORC2012), Third IEEE Workshop on Self-Organizing Real-Time Systems (SORT2012)*, pp. 172–179, 2012.

[16] J. Eriksson, M. Faloutsos, and S. V. Krishnamurthy, "DART: Dynamic address routing for scalable ad hoc and mesh networks," *IEEE/ACM Transactions on Networking*, vol. 15, no. 1, pp. 119–132, Apr. 2007.

[17] M. Caleffi, G. Ferraiuolo, and L. Paura, "Augmented tree-based routing protocol for scalable ad hoc networks," in *Proceedings of the Third IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS'07)*, Oct. 2007, pp. 1–6.

[18] M. Caleffi and L. Paura, "M-DART: multi-path dynamic address routing," *Wireless Communications and Mobile Computing*, vol. 11, no. 3, pp. 392–409, Mar. 2011.

[19] S. M. Zin, N. B. Anuar, M. L. M. Kiah, and A.-S. K. Pathan, "Routing protocol design for secure WSN: review and open research issues," *Journal of Network and Computer Applications*, vol. 41, pp. 517–530, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1084804514000460

[20] K. Sha, J. Gehlot, and R. Greve, "Multipath routing techniques in wireless sensor networks: A survey," *Wireless Personal Communications*, vol. 70, no. 2, pp. 807–829, 2012. [Online]. Available: http://dx.doi.org/10.1007/s11277-012-0723-2

[21] E. Alotaibi and B. Mukherjee, "A survey on routing algorithms for wireless ad-hoc and mesh networks," *Computer Networks*, vol. 56, no. 2, pp. 940–965, 2012. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S138912861100377X

[22] A. Boukerche, B. Turgut, N. Aydin, M. Z. Ahmad, L. Bölöni, and D. Turgut, "Routing protocols in ad hoc networks: A survey," *Computer Networks*, vol. 55, no. 13, pp. 3032–3080, 2011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1389128611001654

[23] J. Lee, H. Park, S. Oh, Y. Yim, and S. H. Kim, "A radio-disjoint geographic multipath routing in wireless sensor networks," in *2012 IEEE 26th International Conference on Advanced Information Networking and Applications*, March 2012, pp. 803–809.

[24] J. Park, S. Moh, and I. Chung, "A multipath aodv routing protocol in mobile ad hoc networks with sinr-based route selection," in *2008 IEEE International Symposium on Wireless Communication Systems*, Oct 2008, pp. 682–686.

[25] J. Yi, A. Adnane, S. David, and B. Parrein, "Multipath optimized link state routing for mobile ad hoc networks," *Ad Hoc Networks*, vol. 9, no. 1, pp. 28 – 47, 2011.

[26] T. Clausen and P. Jacquet, "Rfc 3626 - optimized link state routing protocol (olsr)," p. 75, 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3626.txt

[27] E. W. Dijkstra, "A note on two problems in connexion with graphs," *Numerische Mathematik*, vol. 1, no. 1, pp. 269–271, 1959. [Online]. Available: http://dx.doi.org/10.1007/BF01386390

[28] T. Murakami, E. Kohno, and Y. Kakuda, "Radio overlapping reduced multipath routing method by utilizing control packet overhearing to counter eavesdropping on data packets for ad hoc networks," in *Proc. Third International Symposium on Computing and Networking (CANDAR2015)*, Sapporo, Japan, Dec. 2015, pp. 167–173.

[29] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Kluwer Academic Publishers, 1996, vol. 353, pp. 153–181.

[30] D. Johnson, Y. Hu, and D. Maltz, "The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4," RFC 4728 (Experimental), Internet Engineering Task Force, Feb. 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4728.txt