International Journal of Networking and Computing – www.ijnc.org, ISSN 2185-2847
Volume 11, Number 2, pages 533–555, July 2021

A Control Method for Transmission Timing of Control Packets to Prevent the Concentration of Control Packets for Bluetooth MANETs

Akifumi NOMASAKI , Eitaro KOHNO , Reo MORISHIGE , and Yoshiaki KAKUDA†
Graduate School of Information Sciences, Hiroshima City University
3-4-1, Ozuka-Higashi, Asaminami-Ku, Hiroshima, 731-3194, Japan

Received: February 15, 2021
Revised: May 5, 2021
Accepted: May 28, 2021
Communicated by Satoshi Fujita

Abstract

Bluetooth MANETs are Bluetooth-based mobile ad hoc networks (MANETs) which consist of mobile terminals such as Bluetooth-enabled smartphones. In Bluetooth MANETs, a method to quickly establish a connection for Classic Bluetooth (Classic), which has a wide communication bandwidth, and by using Bluetooth Low Energy (BLE) for terminal discovery (Existing method) has been proposed. Bluetooth MANETs are effective as a temporary communication method in time of disaster because terminals can communicate without using communication infrastructure such as base stations. The existing method improves the speed of connection establishment and re-establishment of Classic between terminals, which is expected to occur more frequently compared to Wi-Fi and other technologies in the communication range of Bluetooth. On the other hand, there is a problem of establishing connections with a large number of terminals in high density areas, which is likely to occur when the network size increases. However, the existing method does not consider the problem of establishing connections with a large number of terminals in a high terminal density area. In this paper, we provided the following contributions. (1) By conducting real terminal-based experiments, we observed the problems that occur when Bluetooth MANETs are applied to a large network size, i.e., when the network size of the existing method is increased, and discussed the causes. In addition, we proposed a method to mitigate the problem and verified the effect of the method using actual terminals. In this paper, we first prepared 50 Android terminals to check the problems that occur when the network size increases, and carefully measured the behavior and problems of the existing method. As a result, in an environment where there are many terminals that applied the existing method, while a terminal executes the process of a Classic connection establishment, requests from other terminals to establish a Classic connection (hereinafter referred to as "control packets") are concentrated. Terminals must execute processes of connection establishment with multiple terminals simultaneously (hereinafter referred to as "concentration of control packets"). At that time, due to the specification of the profile used in Bluetooth MANETs, the success rate of established connections drops rapidly when multiple connection establishment processes are performed simultaneously. As a result, we found that it is difficult to establish connections in an environment where there are many terminals around, making it difficult to disseminate data packets. (2) To solve this problem, we propose a method to establish a connection more reliably by controlling the transmission of control packets (hereinafter referred to as "our proposed method"). First, we investigated the least number of terminals that can reproduce the situation where connection establishment becomes difficult in order to evaluate our proposed method. As a result, we confirmed that when the existing method is conducted with 6 terminals, the same problem occurs as when 50 terminals are used. Therefore, in this paper, we first implement our proposed method on 6 Android terminals and evaluate the performance. (3) In order to evaluate the applicability of our proposed method in an environment with many terminals, we increased the number of terminals from 6 and conducted experiments on actual terminals. As a result, we confirmed that our proposed method can be applied even when there are up to 14 terminals in the communication range.

Keywords: Bluetooth MANETs, Control packet, Data packet delivery

1 Introduction

Mobile ad hoc networks (hereinafter, referred to as MANETs) [5] [17] are an effective network technology to share data among mobile terminals such as smartphones by utilizing wireless multi-hop communication , such as in a time of disaster. Recently, various research is being conducted on MANETs [16] [6] [9]. In MANETs, terminals can communicate without any communication infrastructure such as base stations. Therefore, applications that can be used everyday are being developed [7] [8] [15]. Bluetooth MANETs are MANETs which consist of Bluetooth-enabled terminals such as smartphones. Bluetooth MANETs also have been studied for communication technology between terminals as shown in [11] [13] [14]. Bluetooth is a communication standard, which various mobile terminals are equipped with. It consumes less power than Wi-Fi because its signal strength of radio waves is lower than that of Wi-Fi. In addition, since the frequency hopping spread spectrum (FHSS) is used in Bluetooth, Bluetooth is more resistant to radio interference than Wi-Fi. Recently, in Bluetooth MANETs, a method has been proposed to communicate using Classic Bluetooth (hereinafter, referred to as "Classic") and discover neighboring terminals by using Bluetooth Low Energy (hereinafter, referred to as "BLE") [10] (hereinafter, referred to as "the existing method"). In the existing method, it has been confirmed that the time required to establish a connection between terminals can be reduced by experiments using 2 to 5 actual terminals such as Android and Raspberry Pi [10] [14]. However, the existing method does not consider the problem of establishing connections in an environment where there are many neighboring terminals within the communication range. This occurs when the network size expands. In a real environment such as an evacuation center during a disaster, it is expected that there will be many neighboring terminals, so it is necessary to consider the problems when establishing connections with multiple neighboring terminals simultaneously. In this paper, we examine the problem caused by increases in the network size when using Bluetooth MANETs, and propose a method to mitigate the problem. First, we confirm the problem by evaluating the performance of practical terminals with the existing method, especially smartphones running the Android OS. Next, we examine the cause of the problem, and propose and evaluate a method to mitigate the problem of unstable Bluetooth connections between terminals due to the increase in network size in the existing method (hereinafter, referred to as "our proposed method"). Furthermore, the public device address (hereinafter, referred to as "PDA") of terminals is used in the existing method to arbitrate the transmission timing of Classic connection establishment request (hereinafter, referred to as "control packet") between terminals. In our proposed method, PDA is used to arbitrate the transmission timing of control packet within a group of adjacent terminals. In this paper, we evaluate the effectiveness of our proposed method through experiments using actual terminals which applied our proposed method when the network size increased, and in high terminal density areas.

2 Bluetooth

Bluetooth [1] [4] is a 2.4(GHz) band wireless communication standard that can be used over short distances ranging from 10 to 100 (meters). Bluetooth has the following features:

1. Frequency Hopping Spread Spectrum (FHSS) is a spread-spectrum modulation method that switches the frequency of the radio wave carrying data one wave after another in a short period of time: Even if other terminals are using a specific frequency band and that band is

Table 1: Specifications of Classic and BLE

Item	Value	
	Classic	BLE
Frequency[GHz]	2.4	
Number of channels	79	40
Standard maximum speed [Mbps]	3	1
Effective maximum speed [Mbps]	2.1	0.65
Maximum transmission power [mW]	100	10
Minimum transmission power [mW]	0.25	0.01
Pairing	Required	Option
Broadcast when disconnected	impossible	possible
Maximum packet size [Byte]	1021	251

not available, communication is possible by automatically switching to a channel that uses another frequency. Thus, FHSS provides high interference resistance.

2. It has a shorter communication range than Wi-Fi, and operates with lower power consumption.

3. It is a connection-oriented communication standard, that requires a connection in advance for mutual communication between terminals. Therefore, it takes time to establish a connection before starting Bluetooth communication.

Table 1 shows the specifications of Classic and BLE in Bluetooth 4.2. Classic is the conventional communication standard of Bluetooth. From Table 1, Classic has a larger maximum packet size than BLE and a larger number of channels, so it is considered to have relatively higher anti-interference performance. In addition, when transmitting packets of 1021 (bytes) or more in Classic, packets are the automatically fragmented and forwarded.

BLE [4] is a standard added in Bluetooth 4.0. BLE is not compatible with Classic, but achieves lower power consumption by significantly reducing the standard values for communication speed, transmission power and data size compared to Classic. However, as shown in Table 1, in addition to the small maximum packet size of BLE, there is the issue that, unlike Classic, there is no function for data fragmentation transfer in the technology.

2.1 Classic Bluetooth (Classic)

Bluetooth uses the 48 (bit) address of the Bluetooth device, which is the same format as the MAC address used in IEEE802.11, to identify a terminal. The Bluetooth device address used in Classic is called the Public Device Address (PDA), and it is uniquely assigned to each Bluetooth device according to the IEEE 802-2014 standard. In Classic, a terminal detection method called active scan is adopted. The terminal to be detected sends packets to discover surrounding terminals (hereinafter, referred to as “terminal detection packets”). The terminal receiving the terminal detection packet returns a response packet to notify the terminal of presence. The terminal under detection can also detect the terminal that sent the packet when the response packet is received. In order for any 2 terminals to communicate with each other using Classic, pairing and connection establishment must be performed in advance. Classic has “Inquiry” indicating that the terminal is in the state of detecting surrounding terminals and “Inquiry Scan” indicating that the terminal is in the state of being able to detect itself. In addition,“Page” indicates that the terminal is in a state to start and

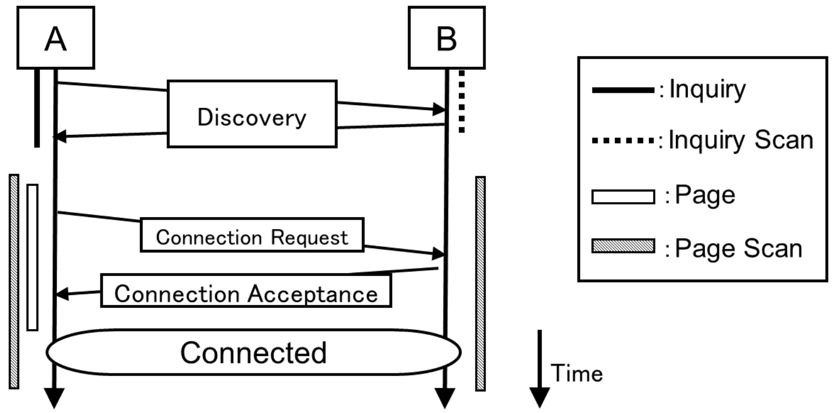

Figure 1: Connection establishment procedure of Classic

complete the connection establishment process and “Page Scan” indicates that the terminal is in a state to receive connection requests. In the following, the connection establishment procedure of Classic is explained using Figure 1.

1. Terminal A in the Inquiry state discovers terminal B in the Inquiry Scan state that exists within its communication range.

2. Terminal A pairs with the discovered terminal B.

3. After pairing is complete, terminal A is in Page Scan only to terminal B and terminal B is in Page Scan only to terminal A.

4. After terminal B accepts the request, a connection will be established between terminals A and B.

The detection of terminals and the transmission of connection request packets in Classic(hereinafter referred as “control packets”) can be performed arbitrarily by the user using a program or other means. If terminals have already been paired, they can send control packets without re-detecting each other again. In addition, a terminal is always in the Page Scan state only when pairing is complete. In Classic, when a connection is established, a master-slave relationship is established between the two terminals, and the master and slave share the communication bandwidth. The terminal that sends the connection request packet is the master, and the other terminal is the slave.

2.2 Bluetooth Low Energy (BLE)

There are two types of Bluetooth device addresses used in BLE, random device address and public device address (PDA). The random device address generates the address of the Bluetooth device with a random value when the application starts. PDA is the same as explained in section 2.1. In BLE, the pairing procedure is not mandatory, but optional. BLE uses a terminal detection method called Passive scan. A terminal sends advertising packets to the surrounding terminals to announce its presence. A terminal under detection can detect, in return, the advertising terminal. The maximum packet size of an advertisement packet is 251 (bytes). The BLE performs connectionless unidirectional communication (hereinafter referred to as “BLE-Broadcast”) using the broadcast function when the connection is not established, when the connection is established, it uses connection-type bidirectional communication. BLE-Broadcast is described in detail below.

2.2.1 BLE-Broadcast

BLE-Broadcast is a communication method that broadcasts a certain amount of information to surrounding terminals without requiring connection establishment. In BLE-Broadcast, there are two states: Advertising, in which the terminal is sending advertising packets, and Scanning, in which the terminal is ready to receive advertising packets. The terminal in the Advertising state broadcasts advertising packets to the surrounding area, and the terminal in Scanning state receives advertising packets. The above operation allows the Advertising terminal to send information to the surrounding Scanning terminals simultaneously.

2.3 RFCOMM

RFCOMM [2] is a part of the Classic protocol stack that emulates the serial port transfer function between two terminals. It is capable of replacing wired data transfer with wireless transfer. RFCOMM supports communication of endpoints such as computers, and devices involved in communication such as modems. It supports the emulation of multiple serial ports between two terminals and serial ports between multiple terminals. When a Bluetooth device establishes a connection between two terminals, an RFCOMM entity establishes a session at each terminal. A session is a communication channel with one other Bluetooth device on the RFCOMM, and each session is identified by an identifier called DLCI (Data Link Connection Identifier). When a terminal establishes a connection with several different terminals, there are several RFCOMM entities. However, when a Bluetooth device establishes a connection with multiple terminals at the same time, multiple internal RFCOMM entities also try to establish a session at the same time. In such a case, the RFCOMM entity may negatively interfere to the received connection command, in which case the connection establishment will likely fail. In these cases, it is necessary to retry the connection establishment after an arbitrary period of time or manually retrying the connection establishment.

3 Bluetooth MANETs

3.1 Application development

In this paper, we implement the methods described so far using only Bluetooth API. Figure 2 shows the protocol stack in Bluetooth MANETs. Figure 2 shows the physical layer of Classic+BLE Baseband, which operates in the 2.4 GHz frequency band and implements the FHSS scheme. As described in 2.3, “RFCOMM” emulates the transfer function of a serial port between two terminals, and can replace wired data transfers with wireless ones. The “Serial Port Emulation” is an entity that emulates a serial port using the Serial Port Profile [3] running on RFCOMM, and provides an API to applications. The “LMP” and “Link Layer” are the core protocols defined in the Bluetooth specifications [4] [1]. “GATT” and “GAP” are the profiles defined in the Bluetooth specification. The “Application” in this paper is described using the Android API, to which the Bluetooth API also belongs. One of the advantages of using Android devices is that the application does not depend on the carrier or manufacturer. Therefore, in this paper, we implement our proposed method described below by developing in “Application” using only the provided API.

3.2 Connection establishment procedure

There are several methods for building a Bluetooth MANET, but the method described in the literature [10] has been implemented in the current state. In this section, we describe the procedure for establishing a connection between two terminals. In the method shown in the document [10], BLE-Broadcast and Classic are used together to establish a Classic connection in order to build a Bluetooth MANET (hereinafter referred to as the existing method). In the existing method, BLE is used for terminal discovery and Classic is used for terminal-to-terminal communication . In the existing method, each terminal transmits its own PDA to the surrounding terminals by BLE-Broadcast. In addition, a timing arbitration mechanism is introduced to avoid control packets

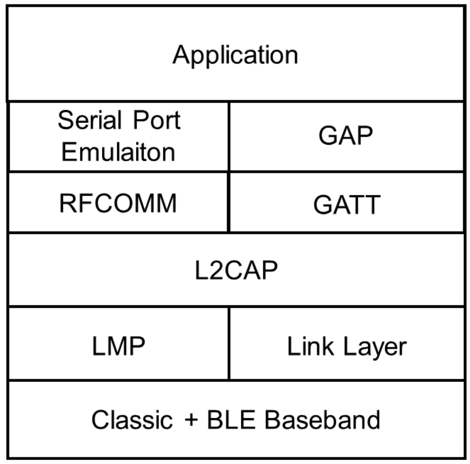

Figure 2: Protocol stack of Bluetooth MANET

conflict that occurs when two terminals discover each other at the same time. When two terminals transmit control packets at the same time, a deadlock occurs when the control packets conflict. When a deadlock occurs, the two terminals need to wait until a predetermined time elapses. As a result, the time required to establish a connection increases.

First, the connection establishment procedure in a situation when there is no contention between the control packets, i.e., when the two terminals are in different states, is explained using Figure 3. In the existing method, "Advertising" indicates a state in which an advertising packet containing the PDA and status information is being sent, and "Scanning" indicates a state in which an advertising packet can be received. In addition, the "State information" is a flag indicating whether the terminal is " scanning" or not.

1. Terminal A which is scanning discovers terminal B which is advertising.
2. Terminal A checks the state information of terminal B and sends a control packet to terminal B.
3. When terminal B accepts the control packet from terminal A, a connection is established between terminals A and B.

In step 1, both terminals are Advertising, but only terminal A, which is Scanning, receives terminal B's advertising packet and discovers terminal B. In step 2, terminal A confirms that terminal B is not scanning and performs the Classic connection establishment procedure based on terminal B's PDA.

Next, when control packets are in conflict, i.e., when two terminals receive advertising packets from each other while they are both advertising and scanning, our connection establishment procedure can arbitrate the conflict as shown in Figure 4.

1. Terminals A and B discover each other while scanning.
2. Terminals A and B check each other's state information and compare their PDAs.
3. Only terminal A, which has a larger PDA than terminal B, sends a control packet to terminal B.
4. When terminal B accepts the control packet from terminal A, a connection is established between terminals A and B.

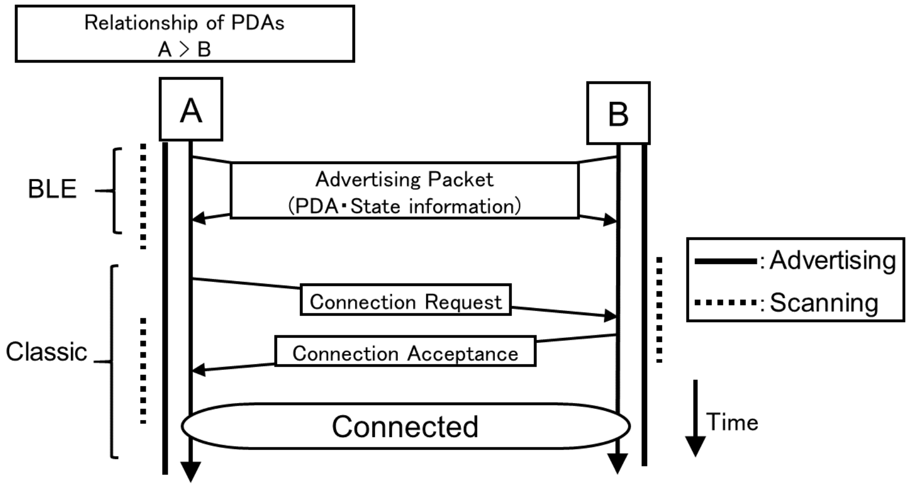

Figure 3: Connection establishment procedure of the existing method

In step 1, terminals A and B receive advertising packets from each other and discover each other. In step 2, terminals A and B confirm that they are scanning each other, and compare the received PDA with their own PDA. In step 3 and later, terminal A, which has a larger PDA, sends a control packet to terminal B and performs the connection establishment procedure. In the existing method, when a connection is established, the terminal that sends the control packet becomes the master and the terminal that receives the control packet becomes the slave. In other words, when a connection is established in Figure 3 and 4, terminal A becomes the master and terminal B becomes the slave.

3.3 Data transfer scheme

Since Bluetooth MANETs use Bluetooth, which has a relatively short communication range, the connection establishment status between terminals changes frequently due to the movement of terminals. In an environment with low terminal density, the number of terminals that can communicate directly by Bluetooth is limited, and data transfer by wireless multi-hop transmission may become impossible. On the other hand, in an environment with high terminal density, the number of terminals that can be directly communicated with is limited because the number of connections that can be established is limited by the specification of Bluetooth, just as in an environment with low terminal density. Additionally, the data transfer time becomes longer due to radio interference in high terminal density areas. To solve this problem, Bluetooth MANETs use a data transfer scheme that switches between wireless multi-hop forwarding method and Epidemic Routing [18] [12]. In this data transfer scheme , when each terminal receives data packets from a neighboring terminal and forwards the data packets to another neighboring terminal, the data packets are forwarded to the terminal with which a connection has been established by the wireless multi-hop forwarding method.

On the other hand, if a connection between terminals has not been established, the data transfer scheme is switched autonomously after the connection is established to transfer data based on Epidemic Routing using a list of data identification information called Summary Vector (hereinafter, referred to as SV). In this method, each terminal periodically sends its own SV to all terminals with which it has established connections. Figure 5 shows an example of the operation of the data transfer scheme that switches between wireless multi-hop forwarding method and Epidemic Routing. Terminal S starts to transmit a data packet. At this time, if the connection between terminals A and C is temporarily disconnected, the data packet is transferred from terminals A to B by wireless multi-hop forwarding. After that, the connection is re-established between terminals A and C, and the time T_{SV} of terminal C's SV transmission interval arrives. Terminal C transmits its own SV, and terminal A, which receives the SV of terminal C, compares it with its own SV and transmits

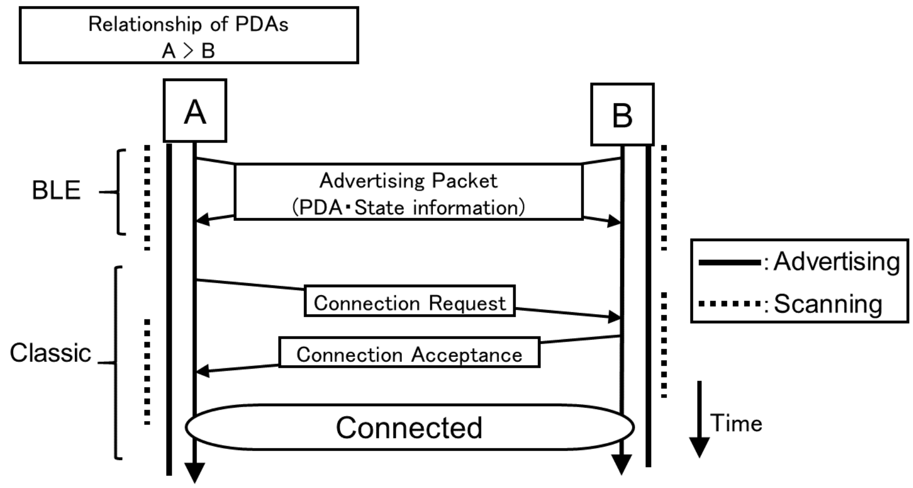

Figure 4: Connection establishment procedure of the existing method when A and B states are in conflict

only the data packet that terminal C does not have. The data packet is then transferred to terminal D, with which terminal C has established a connection, by wireless multi-hop forwarding method. In addition to the above data transfer scheme, paper [12] proposed a transfer function to speed up the data transfer rate in an environment where temporary disconnection is likely to occur. In this function, when the data packet is received, a certain short waiting time (hereinafter referred to as multi-hop waiting time) is set, and the data packet is also stored in a buffer for wireless multi-hop transmission during multi-hop waiting time. Then, when a new connection is established within the multi-hop waiting time, the data packet is transferred by wireless multi-hop forwarding method.

3.4 Problem of the existing method

The existing method achieves fast connection establishment by arbitrating the timing of sending control packets between terminals. To verify the performance of the existing method in a real-world dense network environment, we implemented an application of the existing method on 50 Android terminals. As a result, we confirmed that connection establishment was difficult because of frequent connection establishment failures. Figure 6 shows that the 50 terminals that ran the application with the existing method. In the above figure, 50 terminals running the application of the existing method are placed within a 1.8(m) x 1.2(m) area. In order to analyze and discuss the cause of this phenomenon, we reduced the number of terminals from 50 and investigated the minimum number of terminals that would cause the same phenomenon. As a result, we confirmed that the same phenomenon occurs when 6 or more terminals are placed within the communication range of each other. The reason for this phenomenon is that the existing method does not take into account the connection establishment when there are multiple terminals in the surrounding area. This situation is illustrated in Fig 7. In Figure 7, we use the case when 6 terminals exist within the communication range of each other. It is assumed that each terminal has been paired with all other terminals in advance, and that the PDA of terminal A is the smallest and the PDA of terminal F is the largest. In the existing method, each terminal receives an advertising packet and sends control packets to all the neighboring terminals it find. In other words, the phenomenon described in 2.3 occurs that multiple RFCOMM entities attempt to establish a session simultaneously.

In such a case, the RFCOMM entity may interfere with the receiving connection indication, which is considered to cause the failure of connection establishment. In this paper, we propose a new method for establishing connections among neighboring terminals. In this paper, we only develop the Application Layer as described in 3.1. In this paper, the above issues are solved by

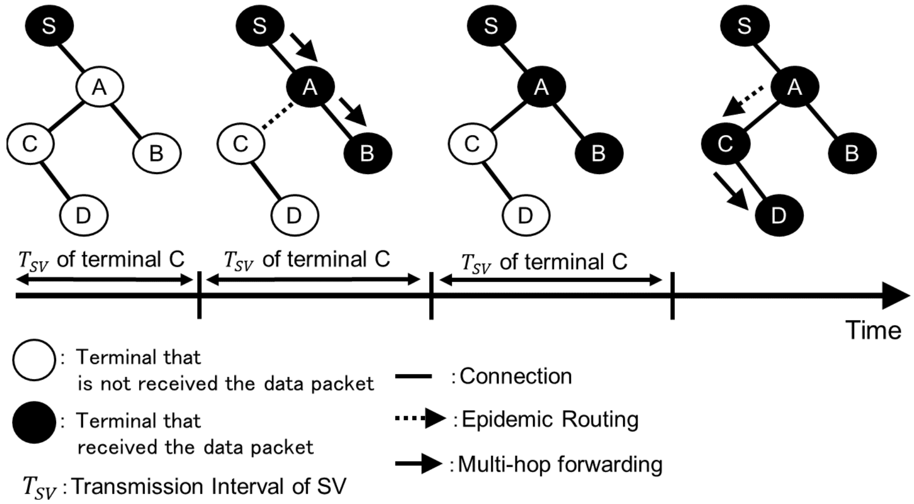

Figure 5: Example of data transfer in the Bluetooth MANET

controlling control packets by the application.

4 Preliminary experiments

In this section, we conducted experiments using a network consisting of 6 or more terminals to investigate the status of control packet processing when multiple terminals are trying to establish connections with each other in an environment when neighboring terminals exist. As described in 3.4, Bluetooth may fail to establish connections with multiple terminals simultaneously. In this paper, we list the situations in which we try to establish connections with multiple terminals simultaneously in a network consisting of 6 or more Android terminals. As a result, we found that the situations in which connection establishment processes are performed in the presence of neighboring terminals can be classified into the following 3 cases in Bluetooth MANETs. The cases are shown in Figure 8.

a). In one case, when a terminal sends control packets to multiple terminals and performs connection establishment processes with multiple terminals simultaneously. This situation occurs when a terminal discovers multiple neighboring terminals. Figure 8(a) shows that terminal A sends control packets to five other terminals.

b). In another case, when multiple terminals send control packets to a terminal. This situation occurs when multiple terminasl are discovered a neighboring terminal simultaneously. Figure 8(b) shows a situation in which terminal B receives control packets from five other terminals. In this case, terminal B establishes connections with up to five terminals simultaneously.

c). In the final case, when a terminal receives control packets from other terminals while performing the connection establishment process. This situation occurs when a terminal is discovered by another neighboring terminal while it is transmitting a control packet and performing the connection establishment processes. Figure 8(c) shows that terminal C sends control packets to 3 terminals and receives control packets from other 2 terminals.

In the following, we created the situations shown in Figure 8(a), Figure 8(b), and Figure 8(c) and measured the performance in each situation.

Figure 6: The 50 terminals within communication range of each terminal

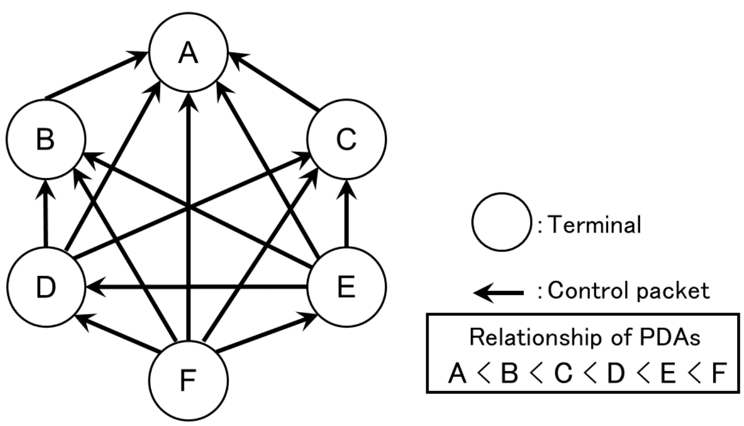

Figure 7: Example of concentration of control packets

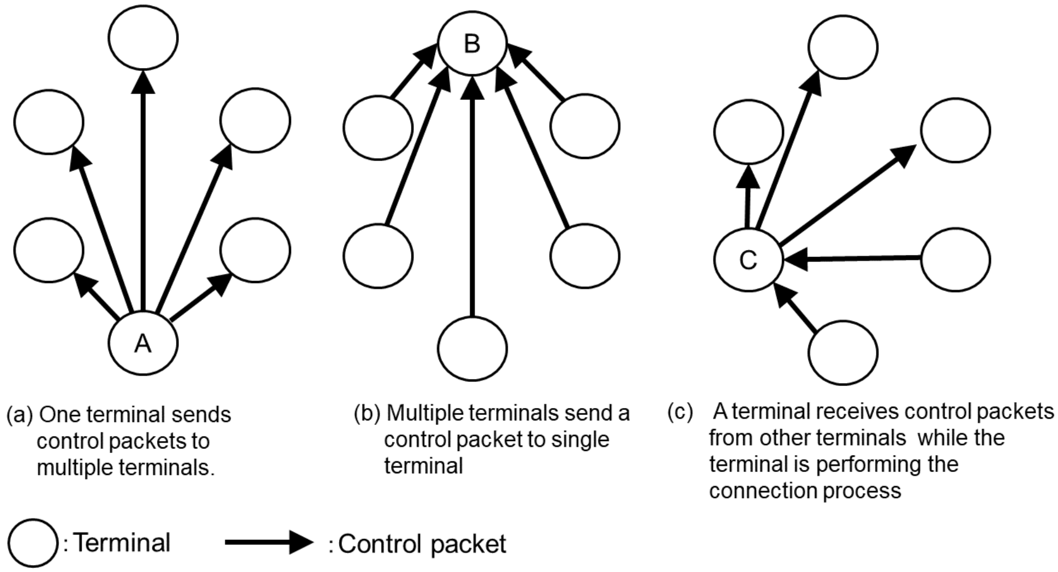

Figure 8: Cases of connection establishment process with multiple terminals in the Bluetooth MANET

Table 2: Preliminary experiment environment

Terminals used	AQUOS SH-M05
OS version	7.1.2
Bluetooth version	4.2
Predetermined random time[sec]	15 - 25
Scanning state time[sec]	3
Scanning state cycle[sec]	10
Number of trials	100

4.1 Experimental evaluation on the success rate of one terminal sending to multiple terminals

4.1.1 Overview

In this preliminary experiment, we measure the success rate of established connections in an environment when a terminal sends control packets to multiple terminals. Table 2 shows the experimental environment of this preliminary experiment. As shown in Figure 8(a), this preliminary experiment measures the success rate of established connections when the number of terminals receiving control packets sent by a terminal is 1 to 5. The success rate of established connections is the number of successfully established connections relative to the number of control packets. In this preliminary experiment, we measure the success rate of established connections when the number of terminals receiving control packets is 1, 2, 3, 4, and 5. The single terminal sends control packets 100 times to each of the 5 terminals. In this preliminary experiment, all the terminals are placed horizontally on a desk next to each other so that they are within the communication range of each other when Bluetooth is used.

4.1.2 Experimental results

Figure 9 shows the success rate of established connections for each terminal that receives control packet. The vertical axis shows the success rate of established connections, and the horizontal

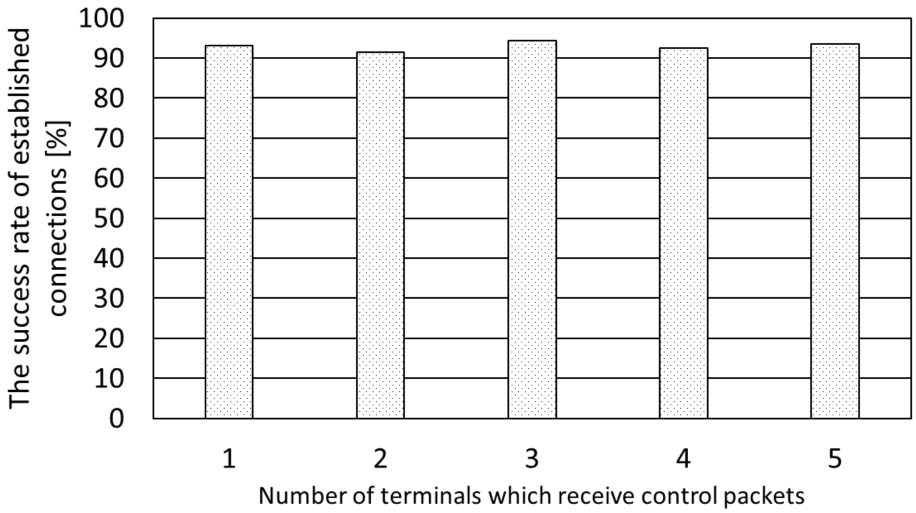

Figure 9: The success rate of established connections for each number of terminals that receive control packets

axis shows the number of terminals that receive control packets. The success rate of established connections with 2 to 5 terminals is the average number of control packets received. From Fig. 9, we confirmed that the success rate of established connections is almost constant at about 90% even when the number of terminals which receive control packets increases from 1 to 5. This is the same level as the success rate of established connections between two terminals. Therefore, it is considered that the success rate of established connections is unlikely to decrease in the situation when a terminal sends control packets to multiple terminals and performs multiple connection establishment processes.

4.2 Experimental evaluation on the success rate of multiple terminals sending to one terminal

4.2.1 Overview

In this preliminary experiment, we measure the success rate of established connections when control packets are sent from multiple terminals to single terminal simultaneously. Table 2 shows the experimental environment of this experiment. In this experiment, we measure the success rate of established connections when the number of terminals that send control packets to a terminal is 2 to 5, as shown in Figure 8(b). The definition of the success rate of established connections is the same as in 4.1.2. In this preliminary experiment, multiple terminals send control packets simultaneously and single terminal receives the control packets. In this preliminary experiment, we measure the success rate of established connections for 2 to 5 terminals that send 100 control packets to the single terminal. Each terminal sends control packets 100 times to a terminal. In this preliminary experiment, all the terminals are placed horizontally on a desk next to each other so that they are within the communication range of each other when Bluetooth is used.

4.2.2 Experimental results

Figure 10 shows the success rate of established connections for each number of terminals which send control packets. The vertical axis shows the success rate of established connections, and the horizontal axis shows the number of terminals sending control packets. The success rate of established connections is calculated for each terminal that sends control packets, and the average

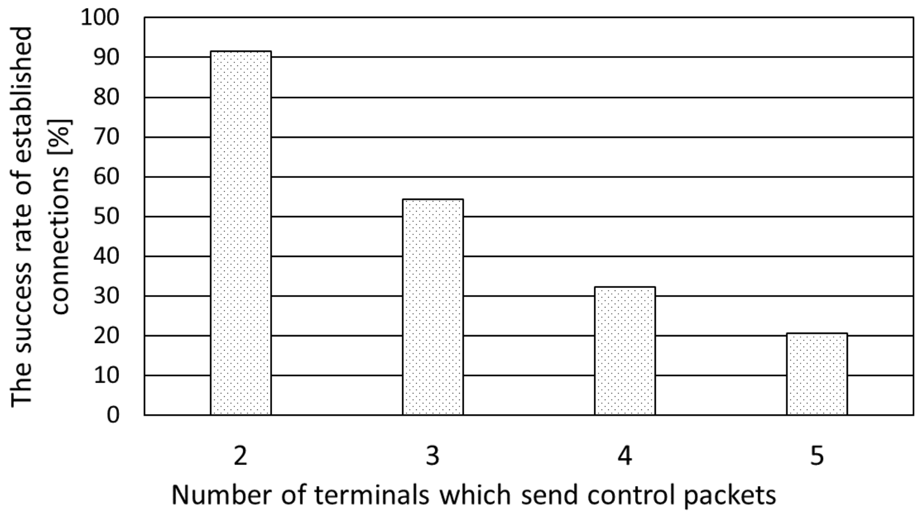

Figure 10: The success rate of established connections for each number of terminals that send control packets

value for each terminal is used as the success rate of established connections for each number of terminals that send control packets. From Figure 10, we confirmed that the success rate of established connections decreases as the number of terminals that send control packets increases. Therefore, it is considered that the success rate of established connections tends to decrease in the situation when multiple terminals send control packets to a single receiving terminal. The receiving terminal performs multiple connection establishment processes simultaneously. However, when the number of terminals transmitting control packets was 2, the success rate of connection establishment was about 90 %. This issue is thought to be due to the fact that the connections were disconnected at arbitrary times between 15 and 25 seconds. Therefore, the situation in which multiple connection establishment processes were performed simultaneously was not likely to occur.

4.3 Experimental evaluation on the success rate of a terminal simultaneously sending and receiving

4.3.1 Overview

In this preliminary experiment, we measure the success rate of established connections when a terminal receives control packets from other terminals while a terminal performs connection establishment processes. Table 2 shows the experimental environment of this preliminary experiment. The preliminary experiment is shown in Figure 11. As shown in Figure 11, terminal B sends control packets to terminals C to F and tries to perform connection establishment processes with these terminals. At the same time, terminal A sends control packets to terminal B and tries to perform connection establishment process with terminal A. In this preliminary experiment, we evaluate terminal B's rate of successfully established connections with terminals C to F, and terminal A's rate of successfully established connections with terminal B. Terminal B sends control packets 100 times to each terminal C through F, and terminal A sends control packets 100 times to terminal B. In this preliminary experiment, all the terminals are placed horizontally on a desk next to each other so that they are within the communication range of each other when Bluetooth is used.

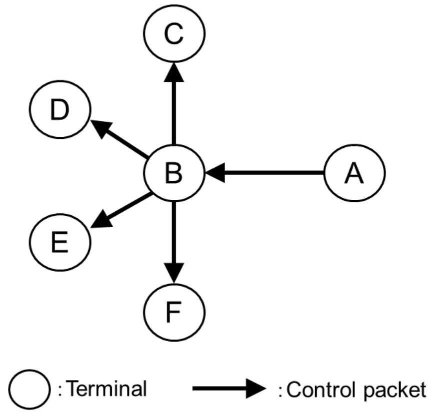

Figure 11: Layout of the terminals in this preliminary experiment

4.3.2 Experimental results

Figure 12 shows terminal B's rate of successfully established connections with terminals C to F. The vertical axis shows the success rate of established connections, and the horizontal axis shows the number of terminals which receive control packets. For example, when the number of terminals which receive control packets is 2, terminal B sends control packets to 2 of the 4 terminals C through F. In another case, when the number of control packets received is 4, terminal B sends control packets to terminals C to F. For the success rate of established connections for each number of terminals, we calculate the success rate of established connections for each terminal which receives control packets, and use the average value of each terminal's success rate of established connections. Figure 12 shows that success rate of established connections is about 85% to 95%, and the success rate of established connections does not decrease even when the number of terminals which receive control packet increases. Figure 13 shows terminal A's success rate of established connections with terminal B. The vertical axis indicates the success rate of established connections, and the horizontal axis shows the number of terminals which receive control packets. The definition of the number of terminals which receive control packets is the same as that in Figure 12. From Figure 12, we confirmed that terminal A's success rate of established connections decreases as the number of terminals which receive control packets increases. From the results of this preliminary experiment, it is considered that success rate of established connections decreases when control packets are sent to terminals which are performing connection establishment processes.

5 Proposed method [13]

5.1 Overview

In this section, to address the issues of the existing method, we propose a connection establishment method that controls the terminals that send control packets between peripheral terminals and the timing of those packets (hereinafter referred to as "our proposed method") [13]. Our proposed method handles the connection establishment procedure between terminals, which is particularly significant in Bluetooth MANETs. While the method described in [10] used a unique PDA pre-assigned to each terminal to arbitrate the contention of control packets between terminals trying to establish a connection, our proposed method uses the PDA of each terminal to control multiple neighbor terminals within communication range. In our proposed method, each terminal broadcasts its own PDA and the status of its own connection, and each terminal decides whether to send a control packet based on the PDA and the status of neighbor terminals. Then, only one terminal that satisfies certain conditions among itself and the neighboring terminals sends control packets. In our proposed method, as we have evaluated experimentally in Sections 4.1-4.3, only one terminal sends

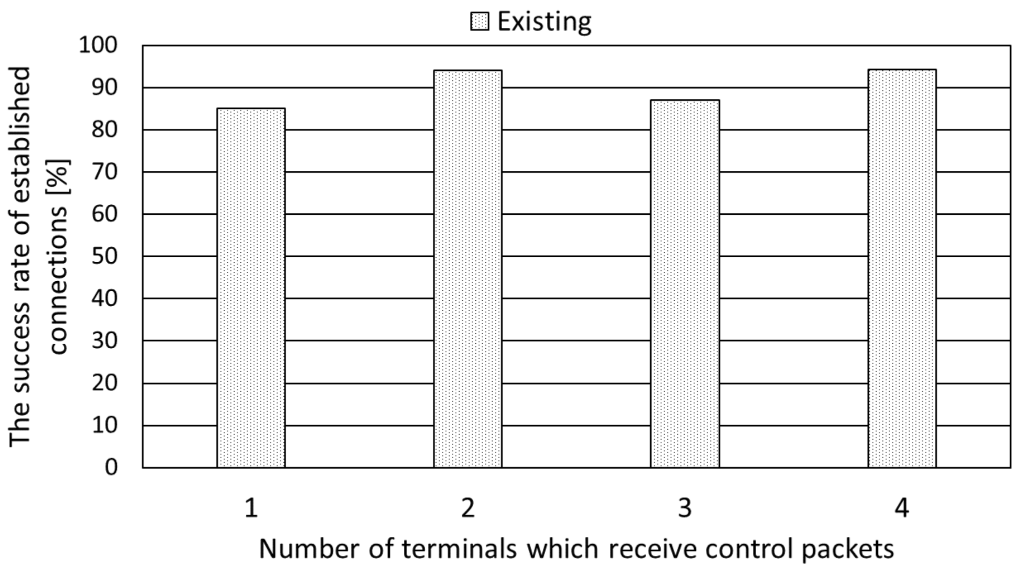

Figure 12: Terminal B's success rate of established connections for terminals C to F

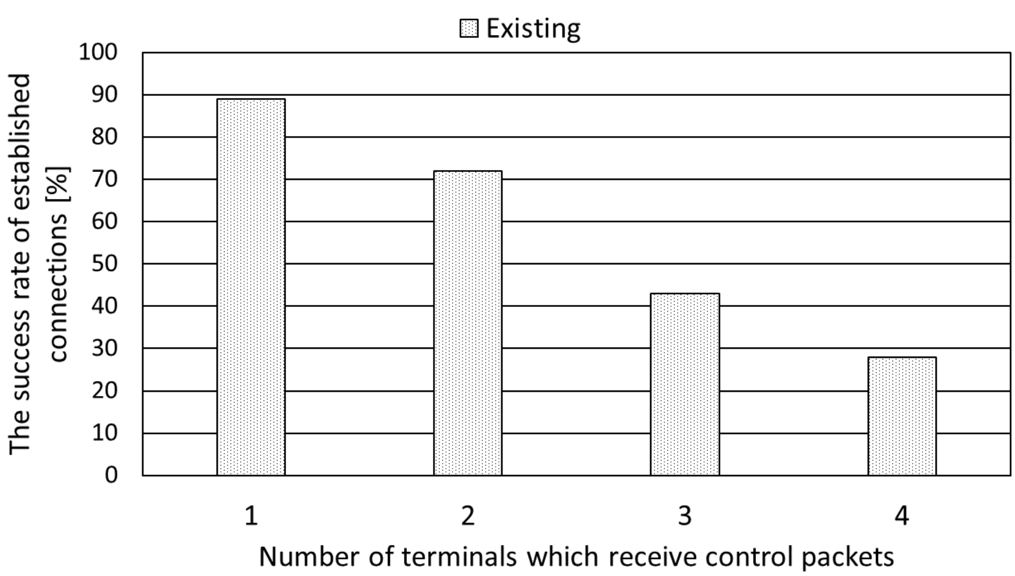

Figure 13: Terminal A's success rate of established connections for terminal B

6Byte	1Byte	1Byte
PDA	Capability of sending control packets	Capability of receiving a control packet

Figure 14: Payload format of advertising packets in our proposed method

control packets, which mitigates the concentration of control packets on terminals and mitigates contention in the connection establishment process. In this chapter, we describe the state of the terminal and the payload format of the advertising packet for our proposed method, and then describe the connection establishment procedure in our proposed method. After that, we discuss the effectiveness of our proposed method through experiments using actual terminals.

5.2 State of terminal

From the results of the preliminary experiments in section 4, the success rate of established connections does not decrease when one terminal sends control packets to multiple terminals and performs the connection establishment process with multiple terminals in the existing method, but when multiple terminals send control packets to one terminal and perform the connection establishment process with one terminal, the success rate of established connections decreases. Therefore, it is necessary to prevent multiple terminals from sending control packets by arbitrating the timing of sending them when the connection establishment process is performed with many neighbor terminals. In our proposed method, we set the following two states for terminals.

1. Capability of sending control packets
2. Capability of receiving a control packet

The first is the capability of sending control packets. In our proposed method, a terminal can send control packets if it has not established connections with 2 or more terminals as the master, and it cannot send control packets if it has established connections with 2 or more terminals. We set the threshold to 2 so that the received data packets are multi-hop forwarded to at least one terminal. By setting this threshold, only terminals that need to establish a connection can send control packets.

The second is the capability of receiving a control packet. In our proposed method, a terminal can receive control packets if it has not established a connection as the slave. If a terminal has established connections with one or more terminals as the slave, it cannot receive control packets. By setting this state, a terminal that sends control packets can send control packets only to terminals that can receive a control packet. Therefore, our proposed method prevents a situation in which specific terminals try to establish connections with many terminals and some terminals cannot receive data packets because they cannot establish connections.

5.3 Payload format of advertising packet

In our proposed method, the state of the terminal, which is set described in section 5.2, is announced to neighbor terminals. For this purpose, terminals store their own state in an advertising packet and broadcast it using BLE. Figure 14 shows the payload format of the redesigned advertising packet to transmit the information necessary to implement our proposed method. The “PDA” in Figure 14 indicates the PDA of the source terminal. The advertising packet is labeled with two flags, each indicating its capability to send or receive control packets respectively. The flag is set to 1 when it is capable.

5.4 Connection Establishment Procedure

The connection establishment procedure of our proposed method is explained using Figure 15. Figure 15 shows the connection establishment procedure when 6 terminals exist within the communication

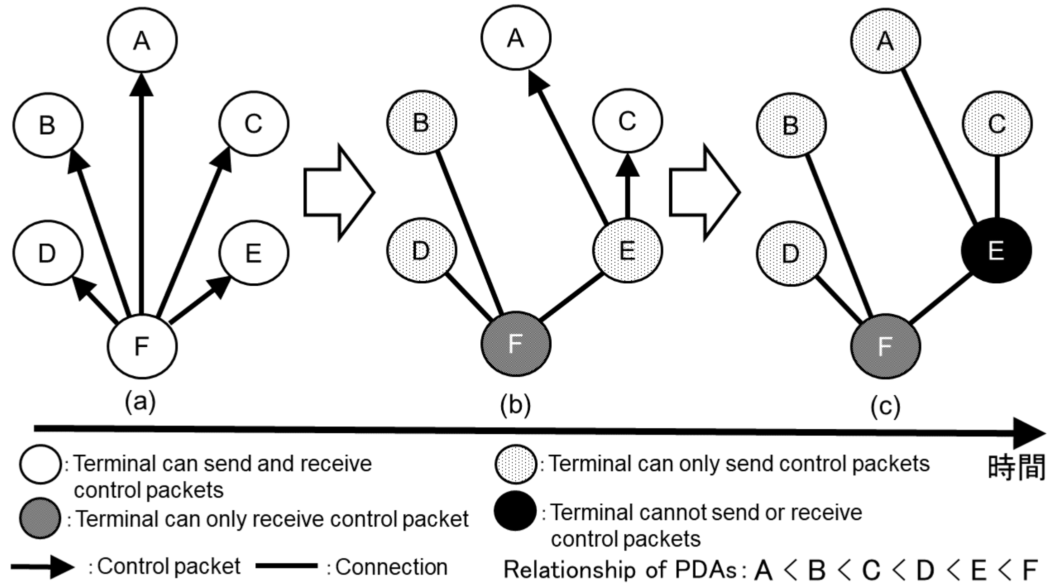

Figure 15: Connection establishment procedure of our proposed method

range of each terminal. It is assumed that the PDA of terminal A is the smallest and PDA of terminal F is the largest. It is also assumed that terminals A to F are already paired with all other terminals. In our proposed method, the terminal that sends the control packet at the time of connection establishment becomes the master and the terminal that receives the control packet becomes the slave, as in the existing method. Furthermore, in Figure 15, it is assumed that each terminal has not established a connection with any terminal in the initial state and terminals can send and receive control packets. Each terminal constantly broadcasts advertising packets in the format described in section 5.3.

1. Terminals A to F discover neighbor terminals by receiving advertising packets from the neighbor terminals, and collect the status of the neighbor terminals.

2. Terminals A to F determine whether their PDA is the largest among the terminals that can send control packets based on the collected status of the neighbor terminals.

3. In Figure 15(a), since any terminal can send and receive control packets, only terminal F, which has the largest PDA, sends control packets to the five terminals with previously acquired the control packets. In the example in this figure, it is assumed that the process of establishing a connection between terminals A and C does not proceed for some reason. There are various possible reasons why terminals A and C do not establish connections with each other even though they are within range of the control packets from terminal F that requests connection establishments. For example, the processing load of terminals A and C may be high temporarily, or the reception power of the control packets at them may be low due to the distance from terminal F. As a result, terminal F established connections with terminals B, D, and E as shown in Figure 15(b). At this time, terminal F cannot send control packets because it has established connections with 3 terminals as the master, and can only receive a control packet. Because terminals B, D, and E have established connections with terminal F as the slave, they cannot receive a control packet and can only send control packets.

4. Terminals A to E discover neighbor terminals by receiving advertising packets from neighbor terminals and collecting the status of the peripheral terminals. In this case, terminal F does not discover neighbor terminals because terminal F cannot send control packets.

5. Terminals A to E determine whether their PDAs are the largest among the terminals that can send control packets.

6. In Figure 15(b), terminal F is unable to send control packets. Therefore, terminal E, which has the second largest PDA among the 6 terminals, sends control packets. Since terminals B and D cannot receive control packets, terminal E sends control packets only to terminals A and C which are able to receive control packets.

7. As a result, connections are established as shown in Figure 15(c). At this time, terminal E has established a connection with terminal F as the slave, and terminal E has established connections with terminals A and C as the master. Therefore, terminal E cannot send or receive control packets. Since terminals A and C have established connections with terminal E as the slave, terminals A and C cannot receive a control packet and can only send control packets.

6 Experiments

6.1 Overview

We conducted experiments on actual Android terminals that implemented both the existing method and our proposed method in order to confirm the effectiveness of our proposed method. In this experiment, the following four items were evaluated.

Number of successfully established connections
This is the sum of the number of times the terminal established connections as both the master and the slave during the experiment.

Number of control packets
This is indicates the number of times the terminal sent control packets during the experiment.

Data packet delivery ratio
This is the ratio of the number of data packets received by the terminal to the total number of data packets sent by all terminals during the experiment.

Latency of data packet forwarding
This is the elapsed time for each data packet to be received during the experiment.

6.2 Configuration of experiments

Table 3 shows the experimental environment. In this experiment, 6 Android terminals were used, and all terminals were placed within the communication range of each terminal. The duration of the connection was set to 15-25 seconds. This was because the average duration of the connection was 20.4 seconds when the distance between the terminals was about 20 meters. The duration of the connection reproduced the situation where the connection was disconnected and re-established due to the movement of the terminal. The buffer size of each terminal was set to 100, and each terminal held a maximum of 100 data packets. This was to prevent an increase in the number of data packets transmitted due to an increase in the number of data packets retained. For this reason, the SV size was also 100, and each terminal periodically exchanged a list of the maximum 100 data packets it held. The multi-hop waiting time was set to 5 seconds, and data packets were forwarded using the wireless multi-hop forwarding method to terminals that have established a connection when data packets were received, or to terminals that had established a connection within 5 seconds of receiving data packets. Each terminal generated a data packet at an arbitrary interval of 1-10 seconds during the experiment, with a maximum of 30 data packets. In addition, each terminal generated data packets containing 30 bytes of text. Each terminal was scanning for 3 seconds in a 10 second cycle and received the advertising packets. In addition, each terminal always broadcasted advertising packets.

Table 3: Experiment environment

Number of trials	10
Experiment time	5
Number of terminals	6, 8, 10, 12, 14
Terminals used	AQUOS SH-M05
OS version	7.1.2
Bluetooth version	4.2
Predetermined random time[sec]	15 - 25
Scanning state time[sec]	3
Scanning state cycle[sec]	10
Buffer size	100
SV size	100
SV transmission interval[sec]	6
Number of data packets	5
Data packet transmission interval[sec]	1 - 10
Multi-hop waiting time[sec]	5

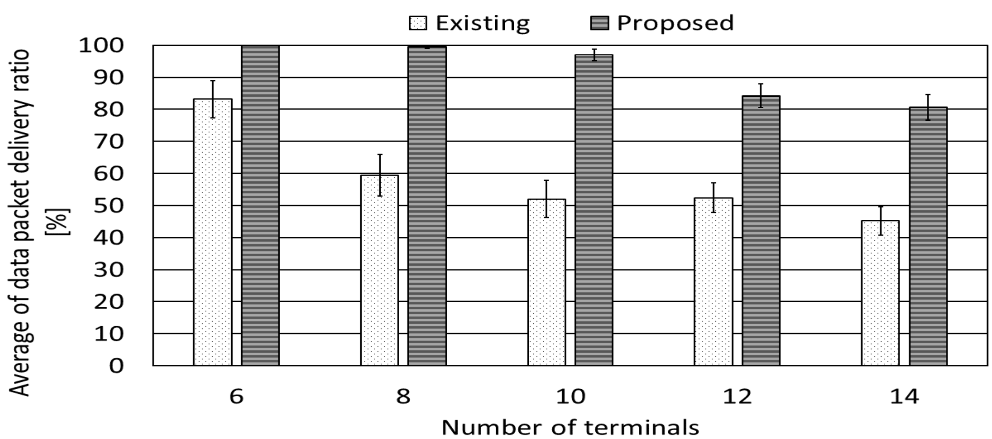

Figure 16: Average of data packet delivery ratio by the number of terminals used

In this experiment, each evaluation item was measured in the span of 10 minutes, and the average value of each trial was calculated. The experiment was conducted indoors at Hiroshima City University, where Bluetooth and radio waves in the same frequency band (IEEE802.11g/n in the 2.4GHz band) were mixed.

6.3 Results

Figure 16 shows the average data packet delivery ratio for each number of terminals used. The vertical axis represents the data packet delivery ratio, and the horizontal axis represents the number of terminals. Each value represents the average of the data packet delivery ratio for all terminals used. Figure 16 shows that the average of data packet delivery ratio was about 95% when the number of used terminals was 8 or 10, and about 80% when the number of used terminals was 12 or 14.

Figure 17 shows the average number of control packets for each number of terminals used. The vertical axis represents the number of control packets, and the horizontal axis represents each terminal, which was arranged in ascending order of their PDAs. From Figure 17, we confirmed that

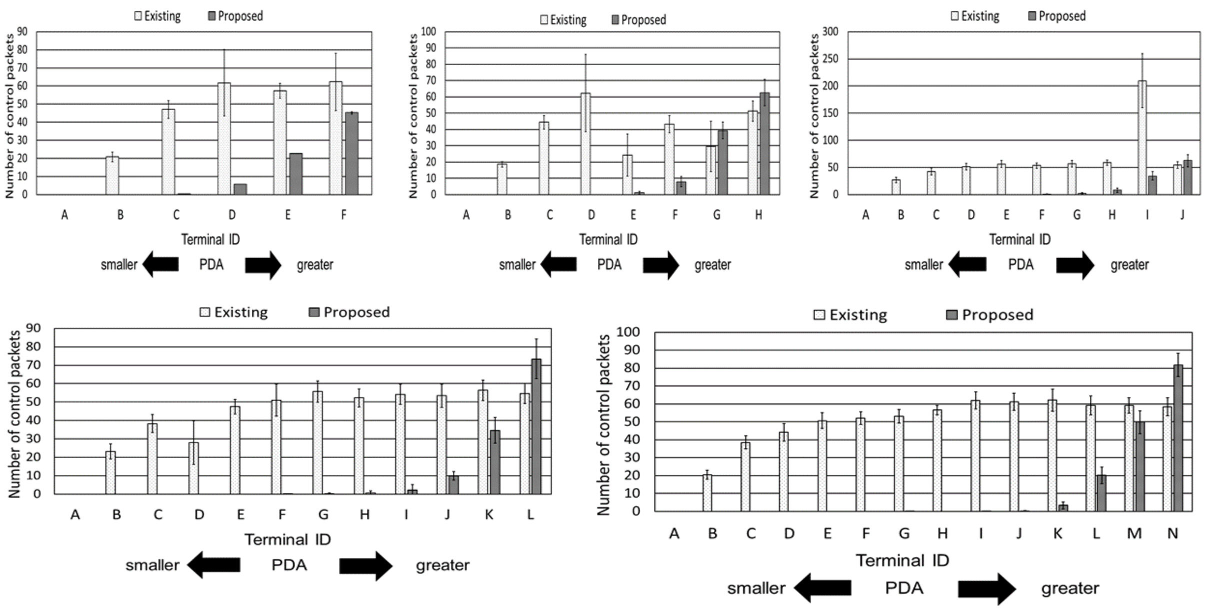

Figure 17: Average number of control packets sent by each terminal by number of terminals used in the configuration

the terminals with the larger PDA mainly send control packets for all the terminals.

Figure 18 shows the average number of successfully established connections for each number of terminals used. The vertical axis represents the number of successful connections, and the horizontal axis represents the terminal, which were arranged in ascending order of their PDAs. From Figure 18, we confirmed that the terminal with the larger PDA mainly sends control packets and established connections more reliably for any number of terminals used. However, when the number of terminals used was 12 or 14, terminals with a relatively lower number of established connections were more likely to appear than in situations with fewer terminals. This is because, in our proposed method, the control packets sent to discovered 5 terminals, the control packets could not reach the 5 terminals.

Figure 19 shows the average of the latency of data packet forwarding for each number of terminals used. The vertical axis represents the latency of data packet forwarding, and the horizontal axis represents the number of terminals. Data packet latency for each terminal configuration, (6, 8, 10, 12, 14), was measured as an avarage of all packet forwarding latencies for each configuration for the duration of the experiment. Figure 19 shows that the average of the latency of data packet forwarding is about 15 seconds when 8 terminals were configured, and we confirmed that the value of the average of the latency of data packet forwarding decreased compared to the existing method. In addition, we confirmed that the average of the latency of data packet forwarding increased as the number of terminals increased. This may be due to the fact that some terminals took time to establish a connection as the number of terminals increased, and the number of hops until data packets are received increased. From these experimental results, we confirmed that our proposed method can disseminate data packets even when the number of terminals increases, although the number of successfully established connections may be much smaller than that of the neighbor terminals compared to the existing method.

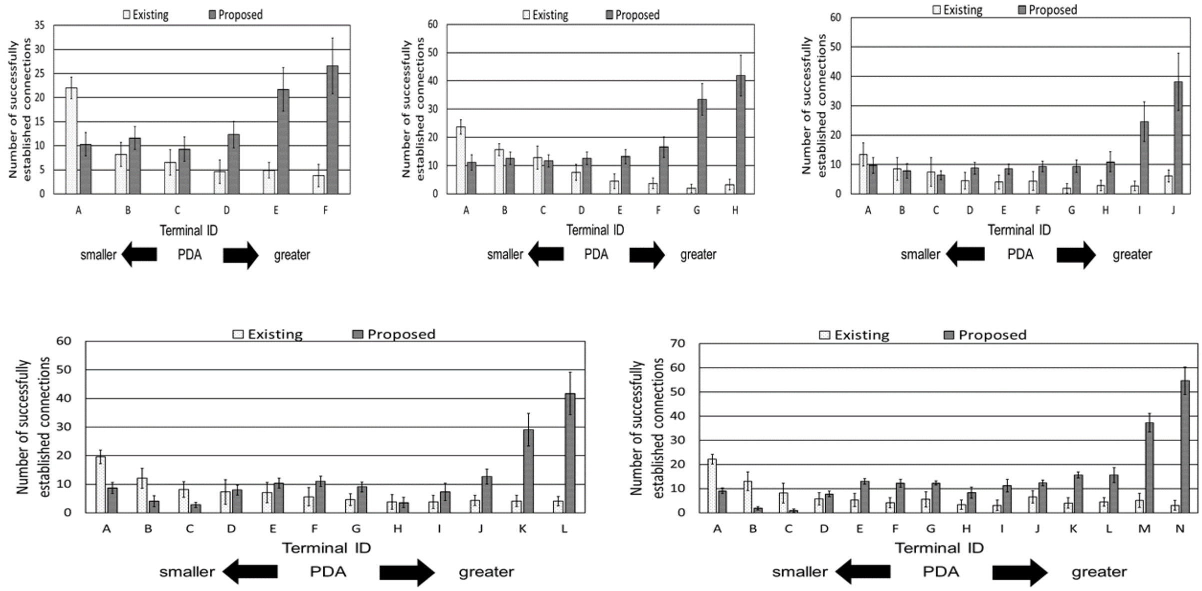

Figure 18: Average number of successfully established connections for each terminal by number of terminals used in the configuration

7 Discussion

We evaluate our proposed method based on the results of the evaluation experiments conducted described in section 6.2. By sending control packets one by one, our proposed method can increase the number of successfully established connections for terminals with large PDA. In Fig. 17, when the number of terminals is 6, the amount of control packets sent by our proposed method for terminal F with the largest PDA value is smaller than that of the existing method. However, for other cases, such as terminal H with the largest PDA value when there are eight terminals, the amount of control packets sent by our proposed method is larger than that by the existing method. This is because the increase in the number of terminals in the network causes interference with the packets sent by other terminals, and the control packets from the terminal with the largest PDA in the network are retransmitted. However, Figure 18 shows that the number of connections established between terminals increases due to the retransmission of control packets from the terminal with the largest PDA in the network, suggesting that our proposed method is effective.

In the evaluation experiment, since specific terminals mainly sent control packets, the terminals with relatively small PDAs had fewer opportunities to send control packets and the number of successfully established connections decreased. These results suggest that our proposed method can establish connections more reliably by controlling the timing of sending control packets when there is a possibility of establishing connections with multiple neighbor terminals. In addition, we confirmed that the average of the data packet delivery ratio increased in spite of the fact that the number of successfully established connections decreased in our proposed method. We believe that this is because the terminals with relatively larger PDA established connections and forwarded the received data packets only to the terminals that had not established connections. Therefore, terminals with a relatively small PDA establish a connection only with a terminal with a relatively larger PDA, while terminals with a larger PDA behave as relay terminals and forward data packets. As a result, the average of the data packet delivery ratio is considered to have increased even though the number of successfully established connections decreased. In addition, our proposed method can reduce the average of the latency of data packet forwarding compared to the existing method. We believe that this is because the number of successfully established connections increased in many terminals.

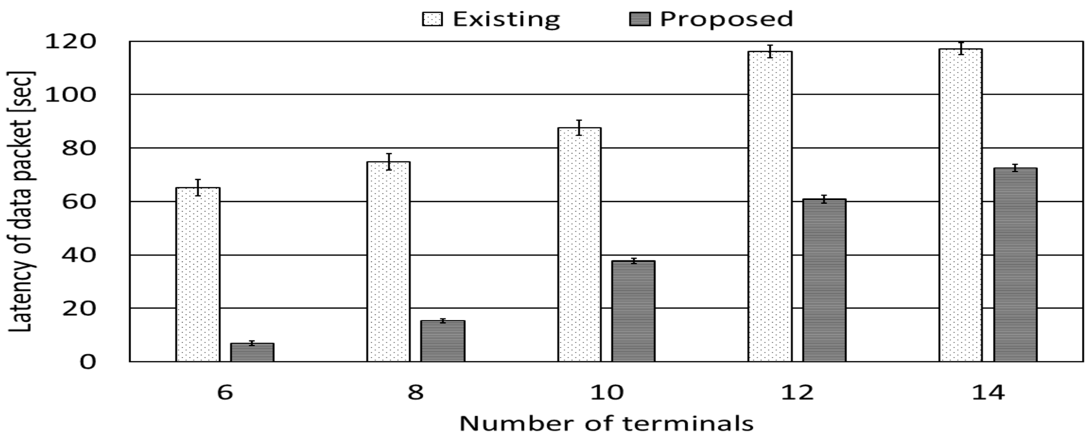

Figure 19: Average the latency of data packet forwarding by number of terminals used

In conclusion, our proposed method can establish connections and disseminate data packets more reliably compared to the existing method.

8 Conclusion

The existing method has the problem that when there are multiple neighbor terminals, there is a high concentration of control packets, which makes it easy for connection establishment to fail and makes it difficult to disseminate data packets. To solve this problem, we proposed a method to mitigate the concentration of control packets by controlling the transmission timing of each terminal's own control packets based on the information of neighbor terminals. From the evaluation experiments, we confirmed that our proposed method increases the average of data packet delivery ratio compared to the existing method, and the data packet delivery ratio average was about 80% even when 14 Android terminals are used. As future work, it is necessary to evaluate our proposed method in a real environment when terminals with different PDA join and leave within communication range. As for the data dissemination of our proposed method, since it was evaluated in an environment where the number of data packets is relatively small, it is necessary to evaluate our proposed method in an environment where the number of data packets sent and received by terminals is large.

Acknowledgment

This research is supported by JSPS KAKENHI Grant Number (C) (No. 17K00130, 17K00131, and 20K11775).

References

[1] Bluetooth Core Specification. Bluetooth SIG, Dec. 2016. Covered Core Package version: 5.0.

[2] RFCOMM WITH TS 07.10 Serial Port Emulation. BLUETOOTH DOC, Nov. 2012.

[3] SERIAL PORT PROFILE. BLUETOOTH DOC, July. 2012.

[4] Specification of the Bluetooth system. Bluetooth SIG, Dec. 2014. Covered Core Package version: 4.2.

[5] S. Basagni, M. Conti, S. Giordano, and I. Stojmenovic. Mobile Ad Hoc Networking, John Wiley & Sons, August 2004.

[6] C. S. R. Murthy and B. S. Manoj. *Ad Hoc Wireless Networks – Architectures and Protocols –*. Prentice Hall, 2004.

[7] A. Ito, Y. Kakuda, T. Ohta, and S. Inoue. New Safety Support System for Children on School Routes Using Mobile Ad Hoc Networks. In *IEICE Transactions on Communications (IEICE/IEEE Joint Special Section on Autonomous Decentralized Systems Technologies and Their Application to Networked Systems), vol.E94-B, no.1*, pages 18–29, January 2011.

[8] Y. Kitaura, Y. Tsutsui, K. Taketa, E. Kohno, S. Inoue, T. Ohta, and Y. Kakuda. The Assessment Information Acquisition and Dissemination System Based on Delay and Disruption Tolerant MANETs for the Hiroshima National Confectionery Exposition. In *Proc. First International Symposium on Computing and Networking (CANDAR 2013), 6th International Workshop on Autonomous Self-Organizing Networks (ASON 2013)*, pages 476–479, December 2013.

[9] C. Marco and G. Silvia. Mobile Ad Hoc Networking: Milestones, Challenges, and New Research Directions. *IEEE Communications Magazine*, 52(1):85–96, 2014.

[10] Y. Minami, N. Kajikawa, R. Saka, Y. Nakao, E. Kohno, and Y. Kakuda. Arbitration-Based Deadlock Mitigation Mechanism for Fast Connection Establishment in Autonomous Self-Organized Bluetooth MANETs. In *Proc. 2018 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computing, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovations, at the Seventeenth International Workshop on Assurance in Distributed Systems and Networks (ADSN2018)*, pages 1611–1616, October 2018.

[11] Y. Minami, Y. Kitaura, Y. Tsutsui, E. Kohno, S. Inoue, T. Ohta, and Y. Kakuda. Implementation and Evaluation of Dual-Purpose Normal and Disaster Situations System Based on Delay and Disruption Tolerant Bluetooth MANETs. *International Journal of Communications, Network and System Sciences*, 8(9):342–357, September 2015.

[12] Y. Minami, R. Saka, E. Kohno, and Y. Kakuda. Data Forwarding Schemes to Adapt Connectivity among Terminals for Bluetooth MANET with Delay and Disruption Tolerance. *Transactions of the Institute of Electronics, Information and Communication Engineers B*, 102(5):356–365, 2019 (in Japanese).

[13] A. Nomasaki, E. Kohno, and Y. Kakuda. On the Connection Establishement Request Control Method to Improve Data Packet Delivery Ratio at Densely Populated Areas for Bluetooth MANETs. In *Proc. Eighth International Symposium on Computing and Networking (CANDAR 2020)*, pages 188–194, Naha, Japan, 11 2020.

[14] T. Ohtani, E. Kohno, A. Nomasaki, and Y. Kakuda. An Adaptive Connection-Establishment Timeout Configuration Method for Bluetooth MANETs in Control Packet Loss Environments. *International Journal of Networking and Computing*, 10(1):25–43, January 2020.

[15] R. Saka, T. Ohtani, K. Fujita, E. Kohno, and Y. Kakuda. On the Design, Feasibility, and Implementation of a Bluetooth MANET-Based Routing Application. *IEICE Communications Express*, 8(12):628–633, 2019.

[16] B. Stefano, C. Marco, G. Silvia, and S. Ivan. *Mobile ad hoc networking: Cutting edge directions*, volume 35. John Wiley & Sons, 2013.

[17] C. K. Toh. Ad Hoc Mobile Wireless Networks, Prentice Hall, 2002.

[18] A. Vahdat and D. Becker. Epidemic Routing for Partially-Connected Ad Hoc Networks. Duke Tech. Report. CS-2000-06, Jul 2000.