PAPER
Rescue Support System with DTN for Earthquake Disasters

Raito MATSUZAKI[†a], *Student Member*, Hiroyuki EBARA[††], *and* Noriaki MURANAKA[††], *Members*

SUMMARY   In a previous paper, we proposed a rescue support system for victims buried in an earthquake disaster by constructing an *ad-hoc* network using home-server based smart homes. However, this system has the following two problems: i) it cannot ensure sufficient density of home servers to realize adequate WLAN coverage, ii) the system does not consider areas in which home servers cannot be used such as parks and factories, for example. In this paper, we propose a new method using a delay tolerant network (DTN) technique. In this method, rescuers (such as rescue teams) with mobile devices relay information between disconnected networks by walking around during rescue activities. For a performance evaluation, we performed simulation experiments using a map of Abeno-ku, Osaka. From our results, we show that the proposed method increases the information acquisition rate, and that the network can be maintained. We also quantitatively show the penetration rate of smart homes needed for our system. In addition, we show that the rescue request system is more effective than other systems, and the method with the mobile device relay is better than without this method.

key words: ad-hoc network, delay tolerant network(DTN), smart home, emergency rescue, earthquake

1. Introduction

In Japan, earthquake disasters are frequent. Problems during an earthquake disaster include communication blackouts and damaged infrastructure, which cause traffic jams. Because of these problems, security companies and public institutions, such as the fire department, often experience delays in their rescue activities. In the Great Han-Shin Awaji Earthquake, the survival rate within 24 hours was about 75%, but was only 15% in 72 hours (3 days) [1]. Making quick rescue activities is essential because the survival rate decreases with the passage of time. When there is no communication infrastructure due to power outages and the breakdown of communication lines, there are two main types of systems that can detect buried victims and can support to rescue them.

First, a sensor network is proposed based on previous research [2], [3] as a system that can detect the location of buried victims. This system requires a large number of sensors and beacon nodes placed in buildings, and it will be expensive to build such a system. In addition, this system cannot work properly without a private power generator system in the building if there is a power outage. In the research

of [2], vital sign sensors measure the state of disaster victims, and this information is sent to mobile devices used by emergency medical technicians (EMT). The communication pathway between these sensors and receiving devices is established in an *ad-hoc* network.

Second, a vehicular *ad-hoc* network (VANET), used as an information-sharing system is proposed based on previous research [4]–[6]. In this system, vehicle nodes which have communication functions can relay the information necessary to rescue buried victims without an existing communication infrastructure. However, in earthquake disasters, vehicles cannot move freely due to blocked streets because of rubble from collapsed buildings, sinking roads, and vehicles left unattended. Also, due to traffic congestion during an earthquake disaster, it often takes time to arrive at the disaster area.

Our paper specifically deals with rescue support for earthquake disasters. In previous research [7], we proposed a smart home network, built with *ad-hoc* networks using home servers for smart homes, and a rescue support system for buried victims, by making a rescue request map that uses the smart home network in an earthquake disaster. Smart homes manage various systems utilizing home servers to create a comfortable home environment through sensing at normal times. In the smart home network, each home server contains a wireless LAN (WLAN) and a battery function. Thus, this network will work even in the case of infrastructure blackouts and outages in the area, because each home server can form a wireless *ad-hoc* network just after earthquake disasters. To use this network, a home server estimates the presence of buried victims by using information on their presence in collapsed homes and buildings, and by sending this information to neighboring home servers. Each home server makes a rescue request map based on this information. The rescue request map displays location-based information about buried victims on the map. In addition, rescuers (such as rescue teams) can get this map from nearby home servers. Therefore, we can identify the location of buried victims and rescue them quickly. The advantages of our proposed system are as follows:

1. The system can quickly construct *ad-hoc* networks using home servers just after earthquake disasters, and home servers can communicate with each other.
2. The system can only be built by home servers in each house and mobile devices (it is not necessary to install other equipment such as repeaters).

3. The system can maintain the network by the battery functions of both home servers and mobile devices and so offsets infrastructure blackouts and outages in the disaster area.

The smart home network system (our original proposal), however, has a few problems. First, we have to ensure sufficient density of home servers to remain within the WLAN communication range, so introducing this service into rural areas is difficult. Therefore, this system is useful only in urban residential areas, and then only with high penetration rates. Second, some places in urban residential areas (such as parks, factories, schools, and national highways etc.) hinder the network connection between home servers because the length of these places is longer than the WLAN communication range. Thus, we proposed to set repeaters to aid communication between home servers in [7], but such repeaters are highly expensive.

In this paper, we propose mobile devices for rescuers that can acquire information by communicating with disconnected home servers. Mobile devices are introduced as a data exchange method in a delay tolerant network (DTN). The DTN communication system can transfer data with a long delay, but with some confidence even in environments of interruption and disconnection for communication, such as an earthquake disaster. The 'carry and forward' technique commonly used with DTNs can exchange information between nodes. In a smart home network, this technique can relay information among disconnected home server groups because rescuers with mobile devices walk around and obtain information during a rescue activity. In this way, we can solve the problem of communication between disconnected home servers without repeaters. Thus, we show that our proposed method can improve the performance of the information acquisition rate in the overall network by evaluation using our original simulation environment implemented in Java. This simulation uses actual map information taken from the area of Abeno-ku, Osaka. In this map area, the movement of rescuers with mobile devices follows a mobility model based on the Random Waypoint (RWP), which considers obstacles such as buildings and waterways.

2. Delay Tolerant Network

Delay Tolerant Networks (DTNs) [8] were originally proposed to contact and transmit between planets. DTNs are designed to overcome issues affecting connectivity, such as mobility, poor infrastructure, and the short-range of radios. In fact, there are experimental projects such as TIER [9] and KioskNet [10] with the purpose of providing an Internet connection in sparsely populated areas without communicational infrastructures. However, missed contact opportunities among nodes decrease throughput and increase delays in the network. Thus, there are a few proposed methods to increase contact opportunities. For example, by using Capacity Enhancement with Throwboxes in DTNs [11], the authors of this article have proposed the use of throwboxes

to enhance network capacity in mobile DTNs. Furthermore, this enhancement increases the transmission opportunities, as well as throughput between nodes. The authors performed an extensive evaluation of their algorithms by varying both the underlying routing and the mobility models.

There has been much research in DTN previously regarding routing protocols, mobilities and transmission schemes; nevertheless, to date no research ideal for this situation has been conducted. Each DTN performance evaluation of the proposed schemes is highly dependent on routing protocols and node mobilities. In the research of [12], [13], the authors propose and implement the Opportunistic Networking Environment (ONE) simulator. The ONE simulator can use six kinds of routing algorithms (Direct Delivery/Transmission [14], First Contact [15], Spray-and-Wait [16]–[18], Epidemic [19], [20], PRoPHET [21], [22] and MaxProp [23], [24]), and synthetic movement models (Random Movement: Random Walk [25], Random Waypoint (RWP) [26]; Map-Constrained Random Movement: Helsinki City Scenario [13]; Human Behavior Based Movement: Working Day Movement Model (WDM) [27]). Moreover, we can add new routing protocols and node mobilities, such as experiment movement traces.

We describe a few state-of-the-art DTN research initiatives as follows. First, one type of research regarding the DTN routing technique is Data Routing for DTN Environments According to Data Size and Deadline [28]. The authors propose a technique using 'carry and forward' to enable asynchronous communication between two points without infrastructures by using mobile devices, which users move around, for data transport. They assume that each node introduces a small battery-powered wireless server called an Infobox, which transmits mobile devices and has sufficient storage. This research is not only an improvement in data arrival rate and reduction of delay, but also in data size and transmission deadline. This proposed method can send and receive higher value data and improve performance such as the data arrival rate as shown by their simulation evaluation.

Second, the article in [29] describes research for a load-balancing mechanism in participatory DTNs that disseminates content evenly. Based on the difference of mobility patterns, each node changes data capacity and transmission frequency in participatory DTNs. Thus, the battery assumption of nodes becomes uneven, and the assumption collapses in that participation of each node in the network can be taken for granted. In this research, the load-balancing mechanism whereby nodes maintain local estimates of the network workload achieves a high delivery ratio without lacking evenness. The authors propose the CoHabit protocol using the Habit routing protocol [30] applied to this mechanism. From simulation results using actual mobility traces, the CoHabit improved performance in comparison with existing protocols.

Third, the research in forwarding strategies for DTNs is the Quasi-Fair Forwarding Strategy [31]. Existing strategies aim to provide forwarding opportunities to more reach-

able destinations to maximize the overall network throughput. However, these strategies have long end-to-end delay and low throughput for less reachable destinations. To overcome these problems, the authors propose two kinds of new forwarding strategies, which provide data as evenly as possible for each destination. The Max-Fair forwarding strategy provides the reception ratio equally for each node, and the Quasi-Fair forwarding strategy relieves the reception ratio equally a little for each node. From results using the ONE simulator, the Max-Fair forwarding strategy improves the equal reception ratio almost completely although throughput is low, while the Quasi-Fair forwarding strategy improves both results.

Finally, the research that combined DTNs with the Mobile *Ad-hoc* Network (MANET) is the Hybrid DTN-MANET routing protocol (HYMAD) [18], which uses both DTN and MANET routing protocols. In this hybrid protocol, each node is grouped and communicates changes within the group and between groups. When nodes communicate within the group, nodes follow the MANET routing algorithm using a distance vector. When nodes communicate between groups, nodes follow the DTN routing algorithm using improved Spray-and-Wait. From evaluations using the ONE simulator [12], the performance of HYMAD improves the existing Spray-and-Wait. Moreover, the time of arrival from the source to the destination is shorter using HYMAD than the existing Spray-and-Wait.

We discuss our research position in comparison to the existing research discussed above. First, our research uses an original simulation implemented in Java. In this simulation, we can use actual map data. In particular, we consider the width of roads in our simulation environment, unlike the ONE simulator. In our research, we assume that the wireless communication range between home servers and mobile devices is not more than 30 m because of both the influence of walls in houses and our previous experiments [7]. Thus, it is important to consider the width of roads in our simulation, and we do not utilize the ONE simulator in this paper. Second, we assume that each node (both home servers and mobile devices) wants to obtain the information from all other home servers in the network, in order to quickly make an accurate rescue request map in the area. We think that our proposed network does not need a load-balancing mechanism [29], because each home server keeps this map information, like the Infobox [28], and has enough battery life to operate for three days in our assumption (Sect. 3.1). Moreover, we consider that our approach, which uses home servers, is more practical than the Infobox [28]. Because our research does not have priority on the information, as discussed in Sect. 3.2, our research does not need the Quasi-Fair forwarding strategy. Finally, we can characterize the routing paradigms of mobile wireless networks (such as MANET and DTN etc.) along the 2 main parameters in [18]. Figure 1 shows that both MANET and DTN are related to node density and node mobility. From Fig. 1, we consider that both of our research approaches (this paper and [7]) are characterized as Assisted DTNs (A-DTN)
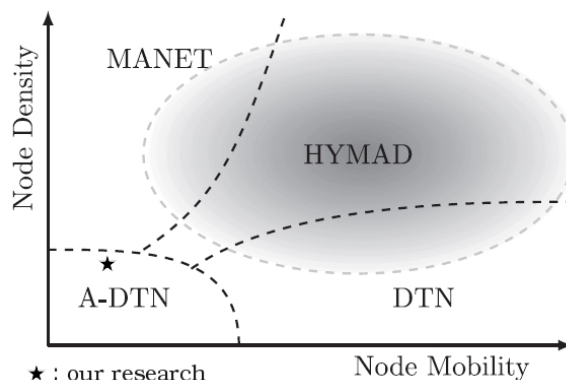
Fig. 1 Mobility vs. density: when different paradigms apply [18].

because of the low (or no) mobility and overall low network density. Thus, we can say that our research is opposite to HYMAD [18].

3. Smart Home Network with Delay Tolerance

In this section we present the smart home network and the rescue support system, and propose a new method to improve connectivity in the smart home network. For more information about the smart home network and the rescue support system, see [7].

3.1 Smart Home Network

In the smart home network, each home server operates a smart home system in the case of normal (non-emergency situations), and constructs *ad-hoc* networks by wireless functions between neighboring home servers in the case of a disaster. The smart home system (sometimes referred to as the smart house) shown in Fig. 2 represents a system that creates a secure home environment in terms of monitoring, consumer electronics operation by sensing, and security. Each home server needs a battery to run this network when the electricity supply stops. Due to the impact of the Tohoku Earthquake, a power self-sufficiency function (such as a high capacity battery, solar power generation, etc.) has been attached to the smart home in recent years. Each home server also needs a wireless LAN (We assume the wireless standard is IEEE 802.11 because this standard is most common) for communicating the *ad-hoc* network.

The smart home network uses Earthquake Early Warning (EEW), which exists in two types [32]. One type is for advanced users and the other type is for the general public. We use both EEWs in the smart home network, and we call the EEW for advanced users the first EEW, and the second type for the general public, the second EEW.

We describe the behavior of the smart home network in an earthquake disaster. Receiving the first EEW, each home server checks connections between neighboring home servers. Receiving the second EEW, each home server sends and shares information (such as home rescue information, etc.) using existing infrastructure such as the Internet. When
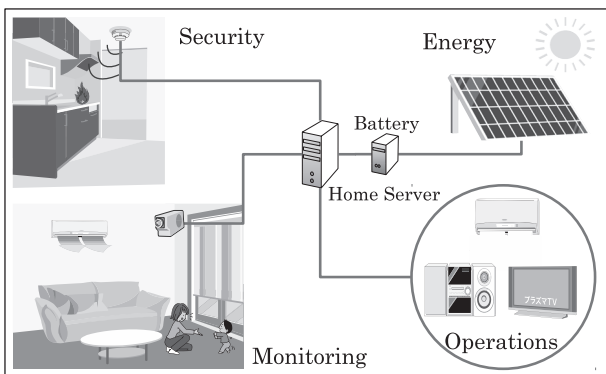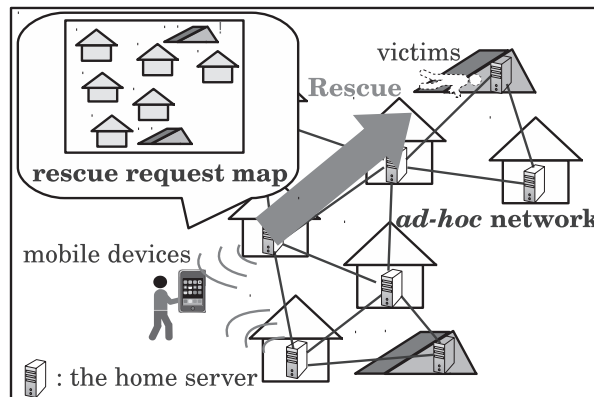
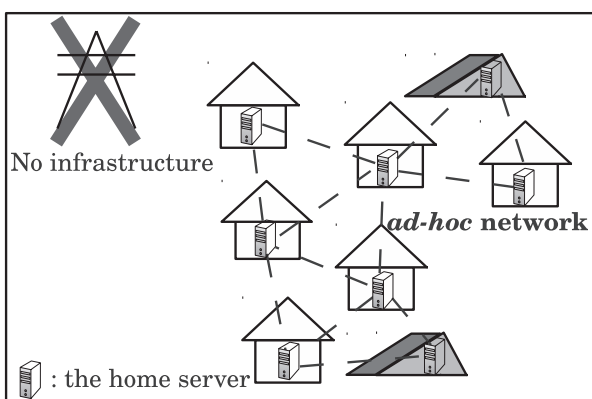**Fig. 2** Example of services by smart homes.

Fig. 3 Construction of *Ad-hoc* networks.

a blackout occurs after an earthquake, each home server connects and communicates with neighboring home servers to maintain the system, as shown in Fig. 3.

3.2 Rescue Support System

It is very important for rescuers to rescue buried victims in an earthquake. However, many rescue operations, such as identifying the location of buried victims, are often difficult. In fact, the way rescuers now often find buried victims is by hearing these victims when they call for help. Therefore, we proposed a rescue support system in [7]. We described this system model, the rescue request information, and the operation of this system as follows.

3.2.1 System Model

In the rescue support system, to create a rescue request that quickly identifies the location of buried victims, home servers share the information for the rescue request map in the smart home network. Rescuers with mobile devices can obtain this information by communicating with home servers, and mobile devices create a rescue request map, as shown in Fig. 4. Then, mobile devices create the rescue request map, and rescuers can quickly identify the location of buried victims and rescue them.

Fig. 4 Obtaining a rescue request map.

Each communication device (home servers and mobile devices) have a map of the surrounding area (assumed to be about 1 km square) in advance, and these devices create a rescue request map by displaying the information for the rescue request map on the map. Each piece of information for the rescue request map is generated by each home server and is not sent too far (assumed to be within 1 km in this case). Therefore, a rescue request map becomes a suitable map for this region.

We assume the communication method is a pure P2P in the rescue support system. In this system, each home server creates information for the rescue request map when earthquake disasters occur (by receiving the first EEW). Then, home servers send this information to neighboring home servers at a fixed time and share it in the smart home network. If a home server updates information or receives new information, the home server sends it to a neighboring home server. The home server doesn't send duplicated information. In addition, each home server checks the behavior of neighboring home servers by confirming periodically the connection of the surrounding home servers.

Each home server estimates the location of buried victims by the information for the rescue request map. We describe the information for the rescue request map as follows.

3.2.2 Information for Rescue Request Map

In the rescue support system, each home server obtains various information to estimate the location of buried victims, and each mobile device creates a rescue request map to display the estimated location of buried victims on the map application in the mobile device. The type of information needed to estimate the location of buried victims by the home servers is as follows:

- Rescue status information
  This information is the status of buried victims in each home. The value is 0 if victims have been rescued or there are no victims, and the value is 1 if the system estimates the presence of victims.
- Rescue request information
  This information is a direct rescue request by the buried

victim. The victim can send the rescue request information (the value is 1 if he is buried alive, and 0 otherwise) to neighboring home servers by the mobile device.

- Home information
  The number of people in a home in earthquake disasters is information that the smart home system obtains through sensors.
- Information from various sensors
  Emergency information such as the detection of a fire, the collapse of a house, etc., is obtained from the sensing signals of the sensors in the smart home system. In the rescue support system, the value is 0 without abnormal emergencies, and the value is 1 with abnormal emergencies.
- Information from disconnected home servers
  This information is estimated if neighboring home servers that have been able to communicate just before the disconnection suddenly disconnect from the network because of abnormal emergencies (such as home server collapse, dead batteries, etc.).
- Coordinate information
  This information consists of longitude and latitude, which are positional information using the GPS from each home server.

In the rescue support system, rescuers can see the location of buried victims by the display on a map application in the mobile devices. At this time, mobile devices send support information to efficiently rescue buried victims by communication with home servers in this proposed system. The type of information needed to support a rescue activity for buried victims is as follows:

- Rescue request information
  This information is a direct rescue request by buried victims (This information is the same as the necessary information for the location estimation of the buried victims by the home servers.).
- Complete rescue information
  The rescuer needs to inform other rescuers that he has rescued a buried victim so that other rescuers can efficiently rescue other buried victims. Thus, this information is sent to the home servers by the mobile device after the rescuers rescue victims.
- Coordinate information of mobile devices
  The location estimation of other rescuers allows each rescuer to know which rescuer rescues which victim. Thus, each rescuer obtains the positional information of other rescuers (consisting of longitude and latitude) using the GPS from each mobile device.

Note that complete rescue information is in the home server's ID. In the rescue support system, we assume that rescuers can see the location of buried victims by map markers that display on a map application in the mobile devices. At this time, the location of the displayed map markers is based on the coordinate information of the home servers

Table 1 The amount of data needed to create a rescue request map.

information	amounts [Byte]
data ID information (ID)	12
rescue status information (RS)	1
rescue request information (RR)	1
home information (HI)	2
information of various sensors (SI)	5
coordinate information (XY)	19 (longitude:10, latitude:9)
update time information (Date)	10

Table 2 The amount of data that each mobile device sends to the home servers.

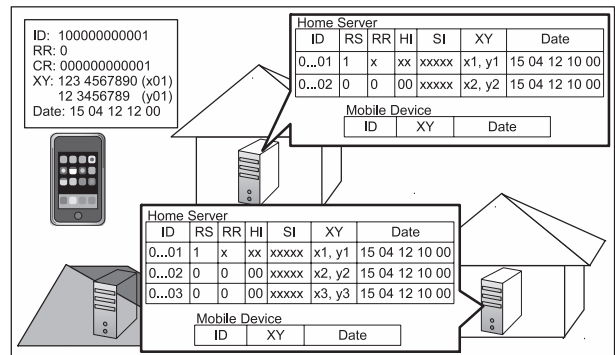| information	amounts [Byte]
data ID information (ID)	12
rescue request information (RR)	1
complete rescue information (CR)	12 (home server's ID)
coordinate information (XY)	19 (longitude:10, latitude:9)
update time information (Date)	10

Fig. 5 An example of the necessary data to update the information.

(Namely, are there any buried victims in this home?). Thus, the rescuer sends the complete rescue information to neighboring home servers by touching the map marker in our proposed system.

Deciding the amount of data for each piece of information needed to create a rescue request map is necessary. In Table 1, we show the amount of data needed to create a rescue request map. Moreover, in Table 2, we show the amount of data that each mobile device sends to home servers. From Table 1, we assume the information required for a single home server to create a rescue request map needs 50 Bytes at the minimum. From Table 2, we also assume the information required for a mobile device to send to home servers needs 54 Bytes at the minimum. In this study, home servers communicate with other home servers by a multi-hop in order to share information for a rescue request map. We assume that each home server sends information which is about 10,000 pieces at the maximum. Therefore, each home server needs to send a plurality of these pieces of information. We assume the amount of information that home servers send is less than 1 MByte.

In Fig. 5, we show an example of the data needed to update the information in our proposed system. Each data is the same as the data in both Tables 1 and 2. In our proposed
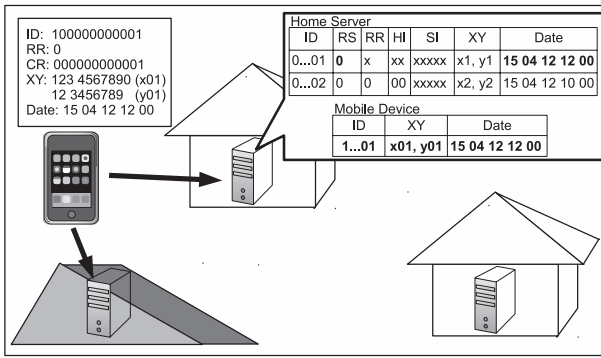
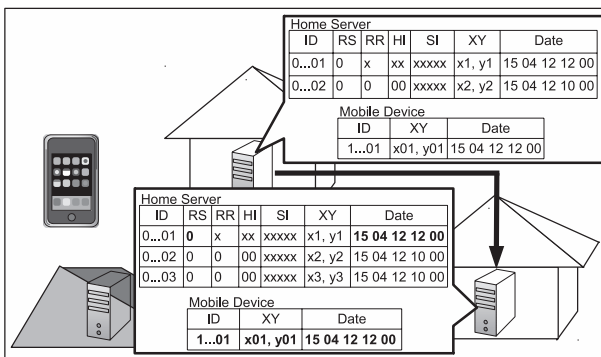**Fig. 6** The home server updates the data from the mobile device.

Fig. 7 The home server updates the data from other home servers.

system, we assume that each of the home servers and mobile devices independently updates the data. Moreover, we also assume that home servers communicate with each other at regular time intervals (several minutes), and that mobile devices communicate with neighboring home servers at regular time intervals (some # of minutes). In addition, the mobile devices communicate with home servers when the mobile devices send the rescue request information or the complete rescue information to these home servers. In Fig. 6, the mobile device sends the data to the home server, and the home server updates the data. Concurrently, this home server determines whether or not to update the duplicated data by the newness of the update time information. In this example, the home server updates the data from the mobile device. In Fig. 6 and well as in Fig. 7, the home server also updates the data from the received data from another home server. If the home server receives the old data from another home server, the home server discards this data without updating this server.

3.3 Operations of the Smart Home Network and Rescue Support System

We describe below the operation of the smart home network and rescue support system in earthquake disasters. In this paper, we assume that a large earthquake occurs from midnight until dawn, such as in the Great Han-Shin Awaji Earthquake, and many people are in their own homes. Moreover,

rescuers who belong to the rescue institution start to rescue buried victims after a few hours. In this situation, rescuers consisting of local residents and local ambulance crews rescue buried victims, and to rescue these buried victims, they must find them first. However, without the proposed system, these rescuers cannot identify the location of the buried victims except when the victims cry for help. Through our proposed system, rescuers can identify the location of buried victims by the display on the map application in their mobile devices, and they can quickly rescue these buried victims.

1. If home servers receive the first EEW, each home server checks the connection with neighboring home servers.
2. If home servers receive the second EEW, each home server shares information for a rescue request map generated by home servers with neighboring home servers. At this time, home servers communicate with each other using the wired network (the normal network line) as a priority to reduce communication loads in the smart home network until the infrastructure is stopped by earthquakes.
3. If home servers cannot communicate by wired networks, each home server establishes a connection with neighboring home servers, and the home servers communicate with each other using the wireless network (the smart home network).
4. If a home server updates information or receives new information, the home server sends it to a neighboring home server. The home server does not send duplicated information.
5. Rescuers obtain the information for the rescue request map using mobile devices when rescuers move the range that mobile devices can communicate with neighboring home servers by WLAN.

These systems need to keep running for at least three entire days, because it is important for rescuers to rescue victims for at least three days as mentioned previously.

Next, we describe how the home server estimates the location of buried victims based on these three types of information as follows:

1. When a home server receives rescue request information (RR), the home server estimates the location of the buried victims by this rescue request information and the coordinate information (XY).
2. When a home server detects abnormal emergencies through sensing (SI), and also detects the presence of people in a home through the home information (HI) after an earthquake, the home server estimates the presence of buried victims in this home (At this time, the home server uses the coordinate information (XY), which represents the location of the buried victims.).
3. When a home server disconnects a neighboring home server and receives home information (HI) with a value more than 1 in the neighboring house just before the disconnection, the home server estimates the occurrence of buried victims by the collapse of the neigh-
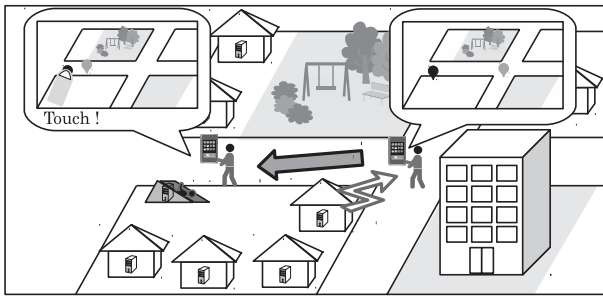
Fig. 8   The usage of the rescue request map.

Fig. 9   The mobile device relay (MDR).

boring house (Also, the home server uses the coordinate information (XY), which represents the location of the buried victims.).

In a smart home system, there are many sensors and cameras for monitoring. As in existing research, the various sensors (such as infrared sensors for person detection [33], [34], differential pressure sensors for examining doors opening and closing [35] etc.), and monitoring cameras [36] can estimate the accurate movements for multiple people in a home environment. Thus, detecting only the number of people in each home is relatively easy because home servers need not estimate the accurate movements for each person, and each home server can estimate the presence of buried victims in the home by the home information (the information of the number of people) before and after earthquake disasters. Moreover, this home server also estimates the risk of buried victims by the condition of the house through the detection of various sensors in the smart home system (for example, the home server detects the collapse of the house by a collapse and crash detector, and this home server also detects a fire by fire sensors and gas sensors). In our proposed system, each home server estimates the presence of buried victims by these methods, and mobile devices display the location of these victims based on the information received from the home servers through the map marker.

Finally, we describe how the rescuers use the rescue request map. The rescuers can use the rescue request map as shown in Fig. 8. At this time, the rescuer identifies the location of buried victims through the display on the map application in mobile devices, and the rescuer goes to rescue this victim. After the rescuer rescues the victim, this rescuer touches the map marker displayed on the map application in his mobile device and other rescuers can see that this victim has already been rescued.

3.4   Proposed Method

3.4.1   Mobile Device Relay

To achieve the smart home network, we consider two major problems. First, because the communication range of the WLAN is small, each home server cannot easily connect to neighboring home servers, especially in rural areas. Thus, we assume that the smart home network can be introduced
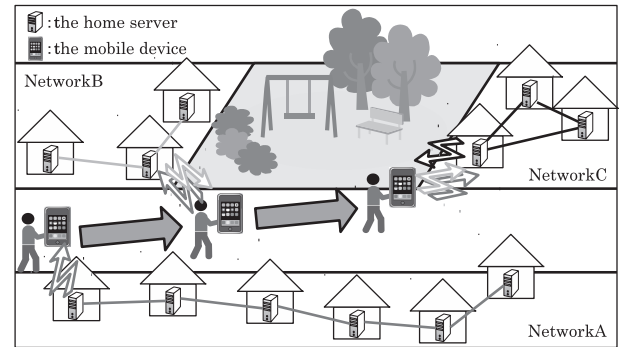
in urban residential areas. Second, urban residential areas which have large areas (such as parks, factories, schools, etc.) have low residential density, and therefore the home server cannot be used. Not putting a home server in these areas causes several small home server groups to form. These problems cause a reduction in the connection rate if a smart home network is introduced. As a possible solution, we set repeaters in these areas, to relay communication between disconnected home servers. However, this solution has two problems as follows:

- It requires a large number of repeaters to improve the connectivity between home servers in the disaster area [7], and it requires high costs to build the smart home network.
- It is necessary to examine the placement of repeaters that can relay communication between home servers.

Therefore, we propose that the mobile device relay (MDR) method communicates between disconnected home servers in the smart home network, as shown in Fig. 9. Mobile devices have a data exchange method used in DTN called, a 'carry and forward' technique, so mobile devices can save information to create a rescue request map and pass it to other home servers. If a rescuer with mobile devices gets information to create a rescue request map from one home server group and moves to another home server group, the mobile devices can communicate with the home server group and share information between the home server groups. The advantages of the MDR method are as follows:

- This method can only be built by a mobile device to relay communication between home servers, and it requires no cost by introducing other devices in each area.
- This method can support the same role as repeaters to relay the smart home network without examining the placement of devices such as repeaters because mobile devices are moved freely through the action of the device users in the area.

In Fig. 9, assume that there are several disconnected home server networks named, NetworkA, NetworkB, and NetworkC, respectively. First, a rescuer gets information given in NetworkA. Second, the rescuer moves and gets
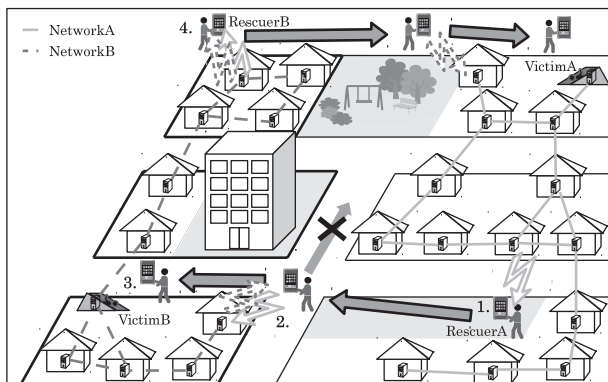
Fig. 10 An effective example of the MDR method in the rescue support system.

information given in NetworkB. At this time, this rescuer sends NetworkB information given in NetworkA. Finally, the rescuer gets information given in NetworkC and sends NetworkC information given in NetworkA and NetworkB. Moreover, this rescuer sends NetworkA information given in NetworkB. In this case, NetworkA has information only given in NetworkA, and NetworkB has information given in NetworkA and NetworkB, and NetworkC has all the information.

We show an effective example of the MDR method in Fig. 10. This example shows the usefulness of our proposed system, and we describe this example as follows:

1. RescuerA gets rescue request information about VictimA from NetworkA.
2. RescuerA moves to rescue VictimA. At this time, this rescuer shares information between other smart home networks. In this case, RescuerA sends NetworkB information about VictimA.
3. If Rescuers get rescue request information about a closer victim, rescuers rescue this victim as a priority. In this case, RescuerA rescues VictimsB as a priority.
4. RescuerB enters the area of the closest victim and gets rescue request information about both VictimA and VictimB from NetworkB. Then, this rescuer selects the closest victim to rescue. In this case, RescuerB selects VictimA and moves to his position to rescue him.

In this example in Fig. 10, RescuerA can rescue VictimB more quickly with the MDR method. Without the MDR method, RescuerA cannot get rescue request information about VictimB, but RescuerB can rescue VictimB. However, the rescue of VictimA and VictimB is greatly delayed because both rescuers rescue victims become more and more distantly located. Thus, the MDR method is effective.

3.4.2 An Algorithm for the Proposed Method

We describe an exchange algorithm for sending and receiving the rescue request map information data in the proposed method. Here, the rescue request map information is data columns as shown in Table 1 needed to create the rescue

request map in Fig. 5. In the exchange algorithm, home servers and mobile devices update information from the update time information (Date) and the data ID information (ID) as described in Sect. 3.3. In this research, we assume that the data ID information is generated and identified by a MAC address. In addition, we assume that each home server and mobile device has the information data pre-sorted by the data ID information. We describe an exchange algorithm for sending and receiving the rescue request map information data as follows:

Step1) When the rescuer with a mobile device moves within the communication range of a home server, the mobile device sends all rescue request map information data (D_{MD}) to the home server.

Step2) The home server compares the received information data (D_{MD}) and its own data (D_{HS}). If the home server does not have the received data, these data are new for the home server. Moreover, if the home server already has the data, and this update time information which the home server has is older than that which the mobile device has, this information data from the mobile device is update data for the home server. Thus, the home server updates from the data D_{HS} to the data (D'_{HS}), and sends it to the neighboring home servers as necessary information.

Step3) The home server compares the received data (D_{MD}) and its own update data (D'_{HS}). Therefore, this home server can determine the required data for the mobile device ($D'_{HS} - D_{MD}$). Specifically, if the home server has data not included in the received data from the mobile device, these data are new for the mobile device. Moreover, if the mobile device already has the data, and the update time information which the home server has is newer than that which the mobile device has, these data are update data for the mobile device. When the home server sends these data, the mobile device updates from the data D_{MD} to the data (D'_{MD}).

In this algorithm, we define data in the home server as D_{HS} and that in the mobile device as D_{MD}. Moreover, we also define the updating data in the home server as D'_{HS} and that in the mobile device as D'_{MD}, and define the differential information data as expressed by $D'_{HS} - D_{MD}$.

Figures 11, 12 and 13 show an example of the algorithm in the proposed method. In Fig. 13, we describe the example of the differential information data ($D'_{HS} - D_{MD}$). The home server compares the received data (D_{MD}) with its own update data (D'_{HS}). Therefore, this home server can determine the required data for the mobile device ($D'_{HS} - D_{MD}$), as shown in Fig. 13. In this example, $D'_{HS} - D_{MD}$ represents the new data column for the mobile device (for example, 0...04 in Fig. 13), and the update data column (for example, 0...02 in Fig. 13).
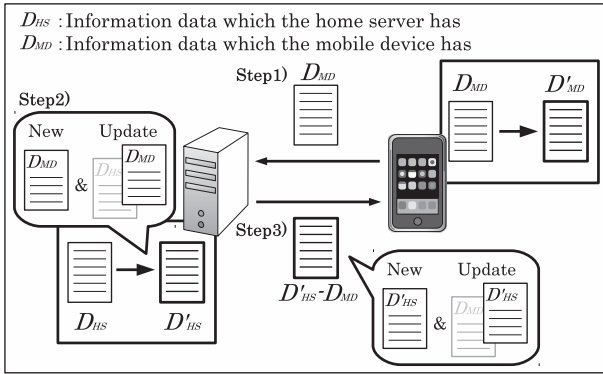
D_{HS} : Information data which the home server has
D_{MD} : Information data which the mobile device has

Fig. 11 An example of the algorithm in the proposed method.

D_{HS}

ID	RS	RR	HI	SI	XY	Date
0...01	1	x	xx	xxxxx	x1, y1	15 04 12 10 00
0...02	0	0	00	xxxxx	x2, y2	15 04 12 10 30
0...04	0	0	00	xxxxx	x4, y4	15 04 12 10 00

D_{MD}

ID	RS	RR	HI	SI	XY	Date
0...01	0	x	xx	xxxxx	x1, y1	15 04 12 12 00
0...02	1	0	0x	xxxxx	x2, y2	15 04 12 10 00
0...03	0	x	00	xxxxx	x3, y3	15 04 12 11 00

Step2)

D'_{HS}

ID	RS	RR	HI	SI	XY	Date	
0...01	0	x	xx	xxxxx	x1, y1	15 04 12 12 00	Update
0...02	0	0	00	xxxxx	x2, y2	15 04 12 10 30	
0...03	x	0	00	xxxxx	x3, y3	15 04 12 11 00	New
0...04	0	0	00	xxxxx	x4, y4	15 04 12 10 00	

Fig. 12 The home server updates the information by the received data from the mobile device.

D_{MD}

ID	RS	RR	HI	SI	XY	Date
0...01	0	x	xx	xxxxx	x1, y1	15 04 12 12 00
0...02	1	0	0x	xxxxx	x2, y2	15 04 12 10 00
0...03	x	0	00	xxxxx	x3, y3	15 04 12 11 00

D'_{HS}

ID	RS	RR	HI	SI	XY	Date	
0...01	0	x	xx	xxxxx	x1, y1	15 04 12 12 00	
0...02	0	0	00	xxxxx	x2, y2	15 04 12 10 30	Update
0...03	x	0	00	xxxxx	x3, y3	15 04 12 11 00	
0...04	0	0	00	xxxxx	x4, y4	15 04 12 10 00	New

D'_{HS}

ID	RS	RR	HI	SI	XY	Date
0...01	0	x	xx	xxxxx	x1, y1	15 04 12 12 00
0...02	0	0	00	xxxxx	x2, y2	15 04 12 10 30
0...03	x	0	00	xxxxx	x3, y3	15 04 12 11 00
0...04	0	0	00	xxxxx	x4, y4	15 04 12 10 00

Compare Step3)

D'_{HS}-D_{MD}

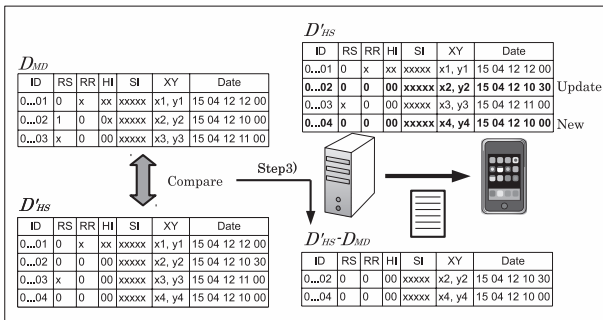| ID	RS	RR	HI	SI	XY	Date
0...02	0	0	00	xxxxx	x2, y2	15 04 12 10 30
0...04	0	0	00	xxxxx	x4, y4	15 04 12 10 00

Fig. 13 The home server creates the data that is sent to the mobile device.

3.5 Assumptions Necessary for Achieving the Proposed Method

To achieve the proposed method, it is necessary to consider several assumptions, described below.

3.5.1 Home Server

The home server has several functions in the smart home network as follows:

- Each home server has WLAN (IEEE 802.11) communications.
- Each home server can save all necessary information to create a rescue request map.
- Each home server sends the differences in information to create a revised rescue request map.
- Each home server broadcasts between home servers, and broadcasts to mobile devices.

3.5.2 Mobile Device

The mobile device of a rescuer has several functions as follows:

- Each mobile device has WLAN (IEEE 802.11) communications.
- Each mobile device gets information to create a rescue request map once.
- Each mobile device broadcasts to home servers (Not to transportable mobile devices).

The 'carry and forward' technique needs to set a transmission deadline to avoid network congestion and waste of bandwidth. Nonetheless, rescuers with mobile devices do not discard information to create a rescue request map during the rescue activity. In addition, the size of the information to create a rescue request map is small because it is less than the product of 8 Bytes times the number of home servers sharing information. Therefore, we do not need to worry about the transmission deadline for information to create a rescue request map, but only need to update new information sent to the home server.

In our proposed method, the mobile device can communicate with home servers, but cannot communicate with other mobile devices; this is called message passing communication. Using passing communication, mobile devices must always have an active communication function, which wastes the battery. Therefore, we assume that the mobile device only communicates with home servers.

3.5.3 Battery

When a large earthquake occurs, the electricity supply is stopped because of damage to infrastructure. Because each home server needs to supply its own power, it needs a large battery, or private power generator such as solar power. In addition, the home server must work for a minimum of three days. The sensors in the smart home and in the mobile devices also need to have enough battery to last for three days. We think, however, that battery capacity will increase through future technological developments. Thus, we assume home servers, sensors, and mobile devices will have enough power supply to achieve the proposed method.

3.5.4 Movement of Mobile Device Users

We assume that rescuers are rescue teams consisting of local residents and local ambulance crews. The leaders of rescue teams consisting of more than one person have mobile devices, and these devices communicate with home servers. In

addition to the rescuers, there are moving people such as patrol teams and local residents in disaster areas. These people walk around (to check disaster situations and the state of their own homes), but they cannot directly rescue victims. However, they can relay communications between home servers with mobile devices. Therefore, we define these walkers as people who can communicate with neighboring home servers and share information. Moreover, we also define rescuers and walkers collectively as mobile device users[†].

We assume that rescuers and walkers move as freely as possible in areas. In other words, they can move onto roads and into parks, but cannot pass through obstacles such as buildings, factories, waterways, and so on. The free action of rescuers and walkers follows the Random Waypoint Model [26] in our research. In the Random Waypoint Model, they move directly to random destinations which they decided on previously. However, Random Waypoints cannot work effectively in areas with obstacles because they cannot move in dead ends nor pass through obstacles. In our model, rescuers and walkers move around obstacles when they cannot move through areas.

Rescuers move freely in areas according to the Random Waypoint when mobile devices cannot get the location of a victim. When the mobile device gets the location of a victim, the rescuer can move there. If rescuers finish the rescue by finding all victims, then the rescuers can move freely again.

3.5.5 Communication Range of Home Servers and Mobile Devices

In our research, we assume that the communication range of the home servers and mobile devices is the same. We also assume that the communication range is equal in a circle. We assume that the communication range for the WLAN is up to 30 m because we measured this range in [7].

4. Performance Evaluation

To show the effectiveness of the proposed method, we performed simulation experiments on an actual map. The simulator is implemented in Java.

4.1 Simulation Setting

Tables 3 and 4 show the parameters of our simulations. We assume the simulation model is in the residential area of Abeno-ku, Osaka, and the simulation area is 1 km square. The map of the simulation area is from the Geographical Survey Institute (GSI) [37]. The location of the home servers and the communication range are in pixels[††]. Note that both the communication range and the measurement range are radiuses of a circle. The density of households in

---

[†]Note that rescuers consist of multiple persons, and walkers consist of single or multiple persons.

[††]Range of 1 pixel is 1.3 m (speed of people).

Table 3 Constant parameters in simulations.

items	parameters
simulation area	Osaka Abeno-ku
simulation range	1 km^2
the number of households	8,000
communication protocol	WLAN (IEEE 802.11)
moving speed	1.3 m/s
communication interval (mobile devices)	60 s
the number of victims	30
the number of the location of victims	10
wait time	360 s
rescue time for a victim	1,800 - 3,600 s (at random)
the range of voice call for help	50 m

the simulation area is obtained from the number of households divided by the area of Abeno-ku. The penetration rate (P_{hs}) [%] is obtained from the number of installed home servers (HS) divided by the number of households (HS_{holds}) in the simulation area, as shown in Eq. (1).

$$P_{hs} = \frac{HS}{HS_{holds}} \times 100 \tag{1}$$

The communication interval for mobile devices is the time when mobile devices do not communicate with home servers. We also assume that the mobile devices run sufficiently. In this simulation, we suppose more than one victim becomes buried in the same house. Thus, we separate the location of victims and number of victims into separate definitions because we suppose that the rescue time is changed by the number of victims. We assume the number of victims is three times the number of the location of victims by parameter tuning. The time that rescue teams wait when rescue teams arrive at their random destinations in the Random Waypoint (RWP) is 360 s. Therefore, the rescue time for victims is randomly from 1800 s to 3600 s. The range of voice calls for help uses the simulation to identify the location of victims without a rescue support system.

Figure 14 shows the actual map of the simulations. The map is divided into 4 areas. The black areas represent places where mobile device users can move, such as roads, parks, schools, and so on. The white areas represent residential areas where home servers can be set. Both of the gray areas represent obstacles. The light gray areas are waterways (examples of rivers, ponds, lakes, seas, etc.). On the other hand, the dark gray areas are other obstacles (examples of factories, buildings, etc.). In our simulations, we assume that mobile device users can enter schools to evacuate victims, but cannot pass through buildings.

We simulate each information acquisition rate of the home servers and mobile devices in the simulation area. In this simulation, the information acquisition rate shows the rate of the information that home servers and mobile devices can obtain from other home servers to create the rescue request map as shown in Table 1. Note that the subject of the collecting information is only the data sending from home servers, but is not the data sending from mobile devices (as shown in Table 2) for simplicity. In the mobile device relay method, if a mobile device relays between disconnected

Table 4 Variable parameters in simulations.

items	Fig. 15	Fig. 16	Fig. 17	Fig. 18	Fig. 19	Fig. 20 and 21
the number of installed home servers	800 - 8,000	4,000	4,000	4,000	800 - 8,000	800 or 4,000
communication range	15 m and 30 m	15 m and 30 m	30 m	30 m	15 m and 30 m	30 m
the number of walkers	10	10	2 - 10 and 20	a total of 10	7	7
the number of rescuers	-	-	-	1 - 9	3	3
simulation time	4 hours	0.25 - 2 hours	0.25 - 2 hours	0.25 - 2 hours	20 hours	1 - 20 hours
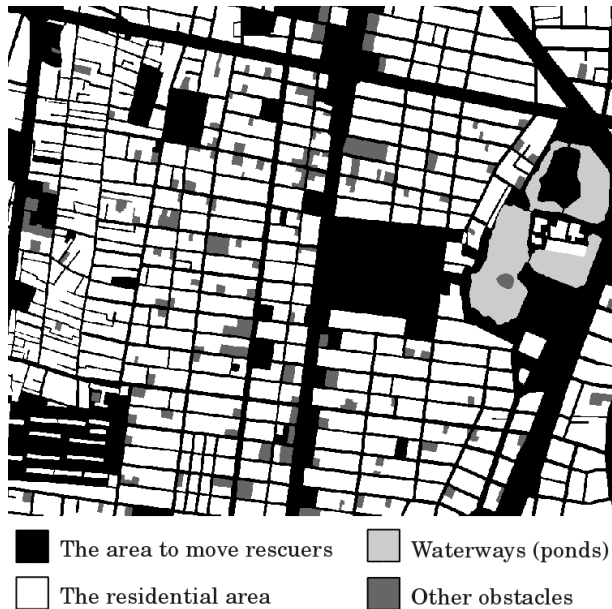
Fig. 14 Map of simulation area in Abeno-ku.

home servers, information acquisition rates of these home servers are added. The amount of data for each home server getting from other home servers in a simulation area is defined by $Nacq_i$, and the amount of data for each mobile device getting from home servers in a simulation area is also defined by $Nacq_j$. We determine the information acquisition rate for home servers (Acq_{HS}) and mobile devices (Acq_{MD}) in a simulation area by the maximum value of each $Nacq_i$ and each $Nacq_j$, as shown in Eqs. (2) and (3).

$$Acq_{HS} = \max\{\frac{Nacq_i}{HS}\} \times 100 \quad (i = 1, 2, \cdots, HS) \quad (2)$$

$$Acq_{MD} = \max\{\frac{Nacq_j}{HS}\} \times 100 \quad (j = 1, 2, \cdots, MD) \quad (3)$$

At this time, the total number of home servers in a simulation area is HS, and the total number of mobile devices in this area is MD.

Then, we simulate the total number of rescued victims for 20 hours in the simulation areas. These parameters are the main metrics of this study. We compare 3 types of methods: 1) a rescue support system with the mobile device relay (MDR) method, 2) a rescue support system without the MDR, and 3) no rescue support system. Without a rescue support system, rescue teams can identify the location of victims if rescue teams move within the range of the voice call for help. In this simulation, however, rescue teams do

not move until they rescue victims if rescue teams arrive at the location of victims. The time when rescue teams do not move is followed by the rescue time for victims, and this time is determined randomly from 1800 s to 3600 s. Moreover, if victims are at the position, the rescue time for victims is multiplied by their number (up to seven). If rescue teams rescue the same victims, the rescue time for victims is reduced X times (X equals the number of rescuers) as early as a single rescue team.

We do simulations of both the information acquisition rate and the total number of rescued victims a hundred times, and then take the average.

4.2 Mobility Model for Mobile Device Users

The movement of mobile device users follows the RWP Model in the case of normal rescue (rescue teams do not find victims), and the shortest path model in the case of rescue (rescuers find victims) in our simulation. In addition, in our simulation model, rescue teams move around obstacles from necessity. In our RWP model, we consider the self-avoiding path, which follows the self-avoiding walk. The self-avoiding walk, which is similar to the simple Random Walk in the two-dimensional lattice path, prohibits same points. Note that the mobile device users follow the RWP method by moving around obstacles to move in a temporary self-avoiding way, and not to select the previous position and its adjacent positions. Thus, strictly speaking, our proposed method does not follow the complete self-avoiding path because rescuers are permitted to pass through the same position after a certain lapse of time in our method. We describe the algorithm for our mobility model using the pseudo code as follows.

Algorithm 1 shows the algorithm for our RWP mobility model. In Algorithm 1, line 12 shows whether the mobile device users move around obstacles in the case of selecting diagonal paths (upper-left, upper-right, bottom-left, and bottom-right), while line 16 shows whether the mobile device users move around obstacles in the case of selecting straight paths (such as left, right, up, and down). When moving in diagonal paths, note that this algorithm selects a diagonal direction for the rescuers to move either in horizontally or vertically adjacent pixels of this diagonal pixel. Moreover, Algorithm 1 do not select the direction not to satisfy self-avoiding path, but this algorithm selects this direction when rescuers cannot move other directions with only a handful of exceptions.

In the shortest pass model, rescue teams can change the destination for the victims quickly if they can identify

Algorithm 1 The RWP model that moves around obstacles.

Set M to the number of mobile device users
for $i = 0$ to $M - 1$ **do**
    //Exclude paths that do not satisfy the self-avoiding path
    Exclude directions that move to the previous position and its adjacent
    positions to avoid selecting these directions.
    //Moving Method
    if Move in either a horizontal or vertical direction to the destination
    using the self-avoiding path **then** {Normally RWP}
        Select the direction in accordance with the normal RWP.
    else {Move around obstacles}
        //Select the next path
        if Move in a diagonal direction using the self-avoiding path **then**
        {The diagonal path}
            Select the direction of the movable diagonal path in accordance
            with moving around obstacles.
        else if Move in a straight direction using the self-avoiding path
        then {The straight path}
            Select the direction of the movable straight path in accordance
            with moving around obstacles.
        else {a handful of exceptions}
            if Move in a diagonal direction from the previous position **then**
            {the previous diagonal path}
                Select a diagonal direction that does not satisfy the self-
                avoiding path.
            else {the previous straight path}
                Select a straight direction that does not satisfy the self-
                avoiding path.
            end if
        end if
    end if
end for

the location of some victims and more victims are closer. Moreover, the rescue teams can determine the number of other rescue teams available to rescue each victim, and the rescue teams do not select a victim until other victims are rescued if many rescue teams move to rescue the victim as shown in Fig. 10 at Sect. 3.4.1. In this simulation, we assume rescue teams determine which other rescue victims to help preferentially as follows:

- If a single rescue team moves to rescue less than three victims.
- If two rescue teams move to rescue less than five victims.
- If three rescue teams move to rescue less than seven victims.

4.3 Results of the Simulations

We describe the results of the simulation. First, we show the result of the information acquisition rate for home servers and mobile devices. In the rescue support system, each home server obtains the information needed to create the rescue request map by communication with the other home servers, and the mobile devices obtain these pieces of information from the neighboring home servers so that rescuers can identify the location of buried victims by the display on a map application in their mobile devices. In other words, the more the information acquisition rate increases,
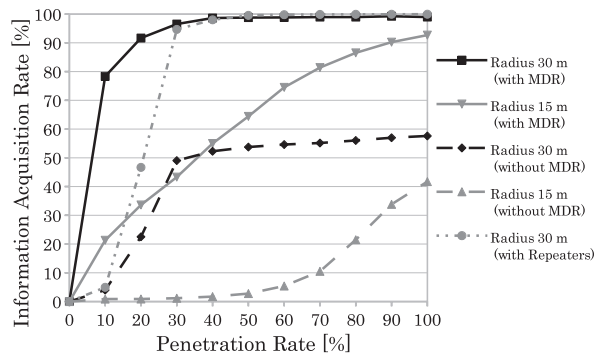

Fig. 15 Characteristics of the average information acquisition rate for the home servers.

the higher the number of buried victims identified by the rescuers increases. Thus, we show that the information acquisition rate for both home servers and mobile devices increases through the mobile device relay method in the simulation. In this way, the rescue support system helps rescuers to identify the location of many buried victims and allows them to rescue these victims quickly.

Figure 15 shows the information acquisition rate for home servers in the smart home network. In Fig. 15, the horizontal axis represents the penetration rate, and the vertical axis represents the information acquisition rate. The communication range of both home servers and mobile devices is 15 m and 30 m. In the MDR method, the number of mobile device users is 10 (composed entirely of walkers), and the simulation time is 4 hours. In Fig. 15, the result of no mobile device relay (MDR) method is measured the information acquisition rate of only home servers. To compare the existing method using repeaters, we simulate the installed repeaters as 190 as shown [7]†. Fig. 15 shows that the information acquisition rate without mobile devices for 30 m was less than 60%, even if the penetration rate was 100%. Because the area in which home servers cannot be used is too large, home servers are divided into several small or middle areas, and the rate for both 15 m and 30 m is very low. However, the information acquisition rate of the proposed methods (with mobile devices) is significantly improved compared to the methods without mobile devices for both 15 m and 30 m. This rate for 30 m is especially high at nearly 100% when the penetration rate is only 50%. In addition, the result of the MDR is much the same as the result of the existing method using repeaters as in [7] for more than a 30% penetration rate. The result of the MDR is greater than the result of the existing method for less than a 30% penetration rate. This result is related to the necessity for examining the placement of repeaters, which is described in both the advantages of the MDR methods and the problem of existing methods in Sect. 3.4.1.

In Fig. 16, we show the information acquisition rate for mobile devices by focusing on a 50% penetration rate

---

†We assume that each repeater can communicate 50 m between other repeaters, and can communicate 30 m to home servers to create a full duplex communication with one another.
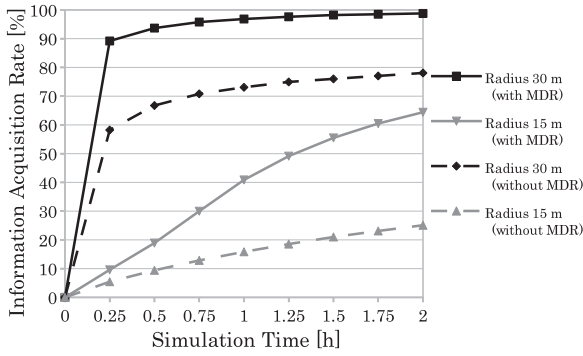
Fig. 16 Characteristics of the average information acquisition rate for the mobile devices.

Fig. 17 The information acquisition rate for mobile device users.

Fig. 18 The information acquisition rate for rescuers.

(the number of home servers is 4,000). In Fig. 16, the horizontal axis represents the simulation time, and the vertical axis represents the information acquisition rate. Other variable parameters in Fig. 16 are the same as those in Fig. 15. In Fig. 16, the information acquisition rates with MDR are significantly higher than these rates without MDR for both 15 m and 30 m. Fig. 16 in particular shows that the information acquisition rate for 30 m with MDR is nearly 100% within 2 hours. In comparison with Figs. 15 and 16, by focusing on a 50% penetration rate, the information acquisition rate for home servers in 4 hours is less than that rate for mobile devices in 2 hours. The reason for these results is that mobile devices can relay communications between home servers, but some home servers do not relay all of the information obtained by the mobile devices as shown in Fig. 9. In other words, the first home server that communicates with the mobile device does not necessarily obtain all of the information obtained from the second or later home server that communicates with the mobile devices. Thus, the acquisition rate for home servers is less than that rate for the mobile devices. To rescue victims quickly, rescue teams must obtain more information for the rescue request map in a short time. Thus, we show that the MDR method is effective for obtaining and quickly sharing information from these results. In addition, this system needs about a 50% penetration rate to get a nearly 100% information acquisition rate in a communication range of 30 m for 2 hours. Figs. 15 and 16 show that these values are enough to maintain the smart home network.

To understand the relationship of smart home network connectivity with the number of mobile device users (walkers + rescuers), we show 2 types of results in Figs. 17 and 18. In these figures, the horizontal axis, the vertical axis, and penetration rate are the same as in Fig. 16, and the communication range is 30 m. Fig. 17 shows the information acquisition rate for mobile devices by the number of walkers. In Fig. 17, the number of mobile device users is from 2 to 10 in 2 increments and 20 (composed entirely of walkers). Fig. 17 shows that the more the information acquisition rates increase when the simulation time and the number of walkers increase. In particular, Fig. 17 shows that the results for both 10 and 20 walkers are nearly 100% within 2 hours,
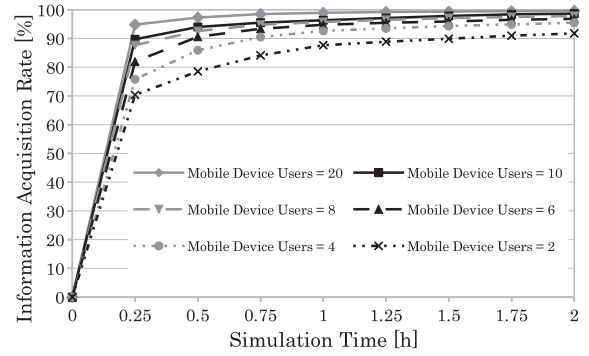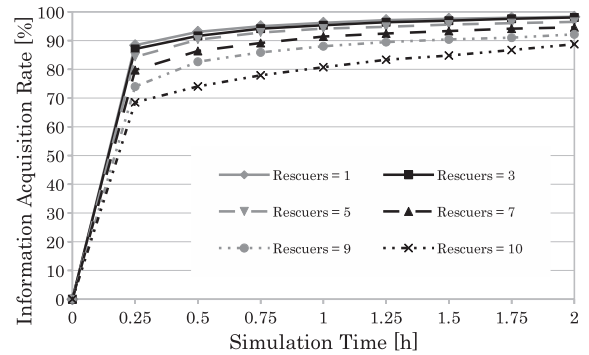
and that the result for 10 walkers is almost same as that for 8 walkers. In other words, the information acquisition rate hardly changes when the number of walkers is more than 10. Thus, we assume that the number of walkers is 10 in subsequent simulations.

Figure 18 shows the information acquisition rate for mobile devices by the number of rescuers. In Fig. 18, there are 10 mobile device users, and they are divided into rescuers (including rescue teams) and walkers. From Fig. 18, when the number of rescuers decrease (namely, the number of walkers increase), the more the information acquisition rates increase. In particular, Fig. 18 shows that the results for both 1 and 3 rescuers are similar. In other words, the information acquisition rate hardly changes when the number of rescuers is less than 3. Thus, we assume that there are 3 rescuers, and 7 walkers in subsequent simulations.

Next, we show the results of the total number of rescued victims with and without our proposed system. In this simulation, we assume that a large earthquake occurs from midnight until dawn such as in the Great Han-Shin Awaji Earthquake, and rescuers who belong to the rescue institution start to rescue buried victims after a few hours. In this situation, rescuers consisting of local residents and local ambulance crews rescue the buried victims, and to rescue them, these rescuers must find buried victims first. However, without our proposed system, these rescuers cannot identify the location of buried victims except for those who cry for help. Through our proposed system, rescuers can identify the lo-
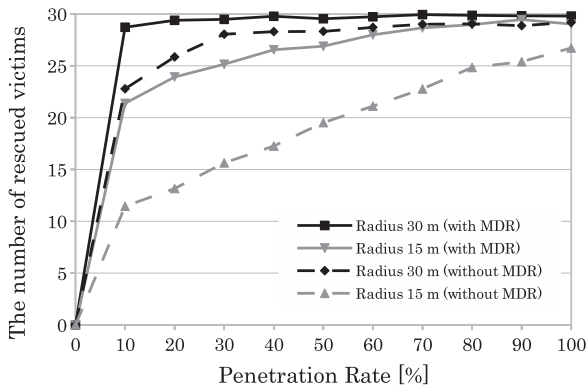
Fig. 19 Characteristics of the average total number of rescued victims in 20 hours.

Fig. 20 Characteristics of the average total number of rescued victims for 10% penetration rate.

Fig. 21 Characteristics of the average total number of rescued victims for 50% penetration rate.

cation of these buried victims by the display on the map application in their mobile devices, and they can quickly rescue these buried victims. From the simulation results as follows, we show that our proposed system works effectively in realistic disaster situations.

Figure 19 shows the total number of rescued victims for penetration of the home server. In Fig. 19, the horizontal axis represents the penetration rate, and the vertical axis represents the total number of rescued victims. We simulate the communication range as both 15 m and 30 m, and we compare the rescue support systems with MDR and without MDR. There are 10 mobile device users (3 rescue teams and 7 walkers). There are 30 victims (the number of their positions is 10). Fig. 19 shows the results in 20 hours. The reason why the simulation time is up to 20 hours in Fig. 19 is because this is the maximum activity time in one day that rescuers can continuously rescue victims. Fig. 19 shows that the total number of rescued victims with MDR is larger than that without MDR in each home server penetration rate. Fig. 19 shows that the more the home server penetration rate increases, the more the total number of rescued victims increases in both results with MDR and without MDR for 15 m. However, both results with MDR and without MDR for 30 m does not follow the increase for more than a 30% penetration rate. For this reason, each result highly depends on the movement of mobile device users because the average information acquisition rate is a nearly 100% within 4 hours for more than a 30% penetration rate, as shown in Fig. 15. From those results, it is clear that the rescue system with MDR is effective because the total number of rescued victims is larger than the number without MDR for each penetration rate.

Figures 20 and 21 show that the total number of rescued victims each time by focusing on the 10% and 50% penetration rates. In Figs. 20 and 21, the horizontal axis represents the simulation times up to 20 hours, and the vertical axis represents the total number of rescued victims. We simulate the communication range as only 30 m, and we compare the 3 types of systems: a rescue support system with MDR, a rescue support system without MDR, and no rescue support system. Other variable parameters in Figs. 20
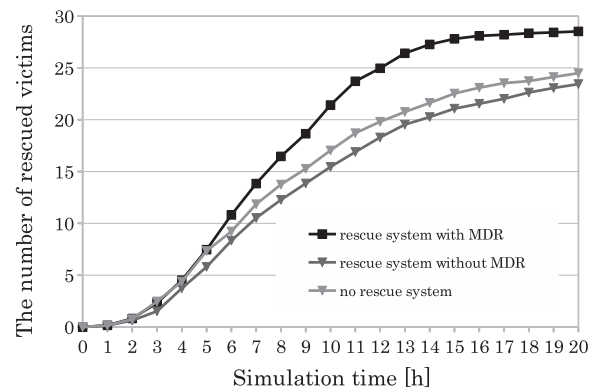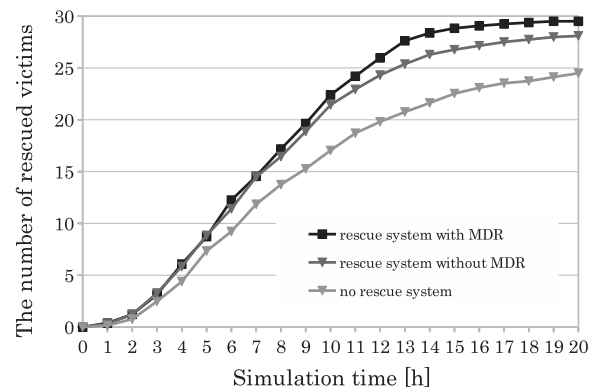
and 21 are same as those in Fig. 19. In both Figs. 20 and 21, the total number of rescued victims with the rescue system is larger than without the rescue system. Moreover, the total number of rescued victims with MDR is larger than that without MDR, and in Fig. 20, for more than 10 hours. Note that the results of the rescue system with MDR and without MDR within 5 hours are not much different in Fig. 20, because each rescue team soon finds nearby victims and rescues them in the beginning of the simulation. In Fig. 21, the results within 7 hours and over 11 hours exhibit a similar tendency similar to the results in Fig. 20. In Fig. 21, in particular, we can see that the total number of rescued victims with MDR is nearly 30 (almost all victims at 100 times the average) for more than 15 hours. In addition, for the 10% penetration rate in Fig. 20, the result of the rescue system with MDR is over 28 within 16 hours, while the result of the rescue system without MDR is less than the result of no rescue system, which is less than 25 for 20 hours. For this reason, the required information cannot be shared between home servers for a 10% penetration rate as shown in Fig. 15, and mobile device users could hardly use this information. In addition, the rescue system without MDR did not use the parameter of help voice (the range of voice call for help as shown Table 3) to ensure the effectiveness of this system itself, and mobile device users without MDR cannot find the

location of buried victims better than without the rescue system. Thus, we believe that the rescue system with the MDR method is effective for rescuing more victims quickly.

5. Conclusion

Buried victims must be rescued within 3 days because this is generally how long victims can survive without help. To do this, rescuers must identify the location of buried victims. We proposed a rescue support system that creates a rescue request map. In our research, we proposed a method for a mobile device relay using the movement of the mobile device users and the 'carry and forward' technique in DTNs to communicate between disconnected home servers in the smart home network during an earthquake disaster. For a performance evaluation, we simulated our proposed method using our original simulation implemented in Java. We showed that our proposed method is effective by performing an evaluation of the information acquisition rate and the total number of rescued victims using an actual map of Abeno-ku, Osaka.

We are currently implementing a communication environment to achieve a smart home network and a rescue support system. Moreover, we need to use more accurate simulators to evaluate the characteristics of wireless communication between home servers using MANET and mobile devices using DTN in our research as future work.

Acknowledgement

References

[1] G.o.J. Cabinet Office, "The lesson data of han-shin awaji earthquake disaster (in Japanese)." http://www.bousai.go.jp/kyoiku/kyokun/hanshin_awaji/data/detail/1-1-2.html

[2] K. Lorincz, D.J. Malan, T.R.F. Fulford-Jones, A. Nawoj, A. Clavel, V. Shnayder, G. Mainland, M. Welsh, and S. Moulton, "Sensor networks for emergency response: Challenges and opportunities," IEEE Pervasive Comput., vol.3, no.4, pp.16–23, 2004.

[3] D. Malan, T. Fulford-Jones, M. Welsh, and S. Moulton, "Codeblue: An ad hoc sensor network infrastructure for emergency medical care," MobiSys 2004 Workshop on Applications of Mobile Embedded Systems (WAMES 2004), vol.5, pp.12–14, June 2004.

[4] J. Sun, X. Zhu, C. Zhang, and Y. Fang, "RescueMe: Location-based secure and dependable VANETs for disaster rescue," IEEE J. Sel. Areas. Commun., vol.29, no.3, pp.659–669, 2011.

[5] H. Chenji, W. Zhang, R. Stoleru, and C. Arnett, "DistressNet: A disaster response system providing constant availability cloud-like services," Ad Hoc Networks, vol.11, no.8, pp.2440–2460, 2013.

[6] S. George, W. Zhou, H. Chenji, M. Won, Y. Lee, A. Pazarloglou, R. Stoleru, and P. Barooah, "DistressNet: A wireless ad hoc and sensor network architecture for situation management in disaster response," IEEE Commun. Mag., vol.48, no.3, pp.128–136, 2010.

[7] R. Matsuzaki and H. Ebara, "Ad-hoc networks using smart homes in an earthquake disaster — delivery of rescue request map data for buried victims (in Japanese)," IPSJ Trans. Mathematical Modeling

and its Applications (TOM), vol.6, no.1, pp.64–78, March 2013.

[8] F. Warthman, "Delay- and disruption-tolerant networks (DTNs)." http://ipnsig.org/wp-content/uploads/2012/07/DTN_Tutorial_v2.03.pdf, July 2012.

[9] "Tier." http://tier.cs.berkeley.edu/drupal/

[10] "Kiosknet." http://blizzard.cs.uwaterloo.ca/tetherless/index.php/KioskNet

[11] W. Zhao, Y. Chen, M. Ammar, M. Corner, B. Levine, and E. Zegura, "Capacity enhancement using throwboxes in DTNs," 2006 IEEE International Conference on Mobile Ad Hoc and Sensor Sysetems, pp.31–40, 2006.

[12] A. Keränen, J. Ott, and T. Kärkkäinen, "The ONE simulator for DTN protocol evaluation," Proc. Second International ICST Conference on Simulation Tools and Techniques, 2009.

[13] A. Keränen and J. Ott, "Increasing reality for DTN protocol simulations," July 2007.

[14] T. Spyropoulos, K. Psounis, and C.S. Raghavendra, "Single-copy routing in intermittently connected mobile networks," 2004 First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, 2004. IEEE SECON 2004., pp.235–244, 2004.

[15] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," ACM SIGCOMM Comput. Commun. Rev., vol.34, no.4, pp.145–158, Oct. 2004.

[16] T. Spyropoulos, K. Psounis, and C.S. Raghavendra, "Spray and wait: An efficient routing scheme for intermittently connected mobile networks," Proc. 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking — WDTN'05, pp.252–259, 2005.

[17] T. Spyropoulos, K. Psounis, and C.S. Raghavendra, "Efficient routing in intermittently connected mobile networks: The multiple-copy case," IEEE/ACM Trans. Netw., vol.16, no.1, pp.77–90, 2008.

[18] J. Whitbeck and V. Conan, "HYMAD: Hybrid DTN-MANET routing for dense and highly dynamic wireless networks," Comput. Commun., vol.33, no.13, pp.1483–1492, 2010.

[19] Y.-P. Kim, K. Nakano, K. Miyakita, M. Sengoku, and Y.-J. Park, "A routing protocol for considering the time variant mobility model in delay tolerant network," IEICE Trans. Inf. & Syst., vol.E95-D, no.2, pp.451–461, Feb. 2012.

[20] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," April 2000.

[21] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," Service Assurance with Partial and Intermittent Resources, Lecture Notes in Computer Science, vol.3126, pp.239–254, Springer Berlin Heidelberg, Berlin, Heidelberg, 2004.

[22] A. Lindgren and K.S. Phanse, "Evaluation of queueing policies and forwarding strategies for routing in intermittently connected networks," 2006 1st International Conference on Communication Systems Software & Middleware, pp.1–10, 2006.

[23] J. Burgess, B. Gallagher, D. Jensen, and B.N. Levine, "MaxProp: Routing for vehicle-based disruption-tolerant networks," Proc. IEEE INFOCOM 2006. 25TH IEEE International Conference on Computer Communications, pp.1–11, 2006.

[24] E.M. Sammou and A. Abdali, "Routing in delay tolerant networks (DTN)," Int. J. Commun., Netw. Syst. Sci., vol.4, no.1, pp.53–58, 2011.

[25] I. Yu and R. Newman, "A topology-aware random walk," IEICE Trans. Commun., vol.E95-B, no.3, pp.995–998, March 2012.

[26] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," Wirel. Commun. Mob. Comput., vol.2, no.5, pp.483–502, Aug. 2002.

[27] F. Ekman, A. Keränen, J. Karvo, and J. Ott, "Working day movement model," Proc. 1st ACM SIGMOBILE Workshop on Mobility Models — MobilityModels'08, pp.33–40, 2008.

[28] W. Sun, Y. Ishimaru, K. Yasumoto, and M. Ito, "Data routing for DTN environments according to data size and deadline (in Japanese)," IPSJ SIG Notes 2010-MPS-80, pp.1–6, 2010.

[29] A.J. Mashhadi, S.B. Mokhtar, and L. Capra, "Fair content dissemination in participatory DTNs," Ad Hoc Networks, vol.10, no.8, pp.1633–1645, Feb. 2012.

[30] A.J. Mashhadi, S.B. Mokhtar, and L. Capra, "Habit: Leveraging human mobility and social network for efficient content dissemination in delay tolerant networks," 2009 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks & Workshops, pp.1–6, 2009.

[31] S.-K. Ko, H. Bang, K. Kang, and C.-S. Park, "Quasi fair forwarding strategy for delay tolerant networks," IEICE Trans. Commun., vol.E95-B, no.11, pp.3585–3589, Nov. 2012.

[32] J.M. Agency, "Earthquake early warnings." http://www.jma.go.jp/jma/en/Activities/eew1.html

[33] C.R. Wren and E.M. Tapia, "Toward scalable activity recognition for sensor networks," Location- and Context-Awareness, Lecture Notes in Computer Science, vol.3987, pp.168–185, Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[34] D.H. Wilson and C. Atkeson, "Simultaneous tracking and activity recognition (STAR) using many anonymous, binary sensors," Pervasive Computing, Lecture Notes in Computer Science, vol.3468, pp.62–79, Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.

[35] S.N. Patel, M.S. Reynolds, and G.D. Abowd, "Detecting human movement by differential air pressure sensing in HVAC system ductwork: An exploration in infrastructure mediated sensing," Pervasive Computing, Lecture Notes in Computer Science, vol.5013, pp.1–18, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[36] S.L. Dockstader and A.M. Tekalp, "Multiple camera tracking of interacting and occluded human motion," Proc. IEEE, vol.89, no.10, pp.1441–1455, 2001.

[37] G.I.A. of Japan, "Geographical survey institute (GS)." http://www.gsi.go.jp/ENGLISH/index.html

Noriaki Muranaka received the B.E. and M.E. degrees from the Kansai University, Japan in 1972, and 1974 respectively and the Dr.E. degree from Himeji Institute of Technology, Japan in 1990. He is a Professor in the Faculty of Engineering Science, Kansai University. Membership in Learned Societies: He is a member of IEEJ (The Institute of Electrical Engineers of Japan), IPSJ (Information Processing Society of Japan), ITE Japan (The Institute of Image Information and Television Engineers), JSKE (Japan Society of Kansei Engineering), and IEEE (The Institute of Electrical and Electronics Engineers, Inc.).

Raito Matsuzaki received the B.S., and M.S. degrees in Faculty of Engineering Science from Kansai University, Osaka, Japan in 2011, 2013. Currently, he is Ph.D. student in the Computer System Engineering Lab. at Kansai University. His research interests are in emergency network in an earthquake disaster using ad-hoc network, DTN, and so on. He is a student member of IPSJ (Information Processing Society of Japan).

Hiroyuki Ebara received the B.S., M.S., and Ph.D. degrees in Communications Engineering from Osaka University, Osaka, Japan, in 1982, 1984, and 1987, respectively. In 1987 he became Assistant Professor of Osaka University. Since 1994 he has been with Kansai University, where he is currently Associate Professor. His main research interests are in computational geometry, combinatorial optimization, and parallel computing. He is a member of IEEE (The Institute of Electrical and Electronics Engineers, Inc.), ACM (Association for Computing Machinery), SIAM (Society for Industrial and Applied Mathematics), IPSJ (Information Processing Society of Japan), and OR Society of Japan.