A Novel Information Sharing Architecture Constructed by Broadcast Based Information Sharing System (BBISS)

Keisuke Utsu¹, Chee Onn Chow², Hiroaki Nishikawa³, and Hiroshi Ishii¹
¹School of Information and Telecommunication Engineering, Tokai University, Minato, Tokyo, Japan
²Faculty of Engineering, University of Malaya, Kuala Lumpur, Malaysia
³Graduate School of Systems and Information Engineering, University of Tsukuba, Tsukuba, Ibaraki, Japan

Abstract - Existing communication infrastructure may be unavailable in disaster situations. Under the situations, it is difficult to share information composed of multiple packets, such as text, image, and audio data in the communication infrastructure unavailable areas. To enable information sharing without using existing communication infrastructure in the areas, we have proposed a novel system “Broadcast-Based Information Sharing System (BBISS)”. The paper evaluates the performance of BBISS by the network simulations. The simulation results can conclude that the proposed method achieves the high information reachability without significantly increasing of the number of packet exchanges.

Keywords: Broadcast, Ad hoc communication, Information sharing

1 Introduction

Communication carriers equip backup systems and batteries to prevent the communication infrastructure disruptions due to disasters. However, the above approaches taken by communication carriers are not sufficient to endure traffic congestions by confusion owing to the disasters, it is difficult to share the information among victims and to execute the rescue and recovery activities. In the infrastructure unavailable situations, damage and safety information are necessary to be shared by text, image, voice, or video data in the area. The information to be shared is as follows:

1. Announcement of damage information and evacuation instructions: The announcement may be broadcast by voices using microphones and speakers by local governments, polices, or firefighters in usual.

2. Sharing information such as safety information, searching, buzzes in the area: The information may be shared by notices and posters among the victims, the local governments, the police, or the firefighters. However, there is no convenient and efficient alternative way using the notices and posters.

Applications that enable the above information sharing are to disseminate generated information to the whole area. For the objective, we have proposed Broadcast-Based Information Sharing System (BBISS). That is a novel information sharing system which uses ad hoc communication in [1].

The paper evaluates the performance of BBISS through the network simulations and shows the information sharing architecture constructed by BBISS. The paper is organized as follows. Section 2 explains outlines and problems of existing information delivery methods using broadcast based communication. Section 3 revisits BBISS proposed in [1]. Section 4 shows the effectiveness of the proposed methods through the performance evaluation by network simulations. Lastly, Section 5 concludes the study.

2 Existing information delivery methods using broadcast communication

The ad hoc network architecture has been studied as a networking technique in the infrastructure unavailable situations, and many routing protocols have been proposed. In general, available IP addresses must be assigned at nodes in the network to communicate using routing protocols. However, the IP addresses are often not available in the infrastructure unavailable situations owing to the large-scale disaster. Moreover, some servers must be required to collect and disseminate the information to adopt the existing client-server applications. Considering the above problems, the existing ad hoc network architecture cannot be applied to the applications discussed in the Section 1.

2.1 Simple flooding

The Simple Flooding (SF) has been implemented in ad hoc network routing protocols to deliver routing messages in the broadcasting manner [2]. Although SF is one solution to disseminate the information to the area, it has following problems. In this method, when a node receives a packet, the packet is broadcast if it has never received before (non-identical packet). Therefore, since many nodes broadcast the packets and data frame collisions occur, the packet reachability is degraded. Probabilistic scheme and Counter-based scheme have been proposed as the methods without HELLO packet exchanges and dedicated devices such as GPS.
2.2 Counter-based scheme

In this method, whether to relay or not is determined by how many times an identical packet has been received. The basic procedure is explained below.
1. On receiving a packet, a node sets its counter at 1 if the packet is non-identical to ever received packet. Identical packets are rejected.
2. The counter value is incremented by 1 if an identical packet has been received during an arbitrary period of time (decision_time).
3. If the counter reaches a threshold value (c_threshold) after expiration of decision_time, broadcasting is canceled.

The assumed information types for applications shown in Section 1 among the nodes are file download type information such as text, image, voice, and video, which are composed of multiple packets, in addition to single-packet messages. Since, the above delivery methods shown in Sections 2.1 and 2.2 cannot complement unreached packets, they are not suitable to the applications.

3 Proposed method, BBISS

Here, we revisit and show the idea of BBISS which we have already proposed in [1] for facilitating understanding.

3.1 Assumed environment

The nodes are wireless communication devices such as smartphones, tablet PCs, or laptop PCs. The nodes in the area have no available IP addresses and gateway information, and they can communicate only by broadcast communication of IEEE802.11 series wireless LAN. Since the proposed system can be implemented over the general UDP/IP platform by socket programming, the proposed system is easily implemented with flexibility on the existing terminals. The shared information is assumed text information (in several Kbytes size) and image information (in several hundred Kbytes size), both of which consist of multiple packets.

3.2 Requirements

We consider the system needs to meet the following requirements #1 to #3.
- Requirement #1: IP addresses, gateway information, and servers at the nodes are not used.
- Requirement #2: Unreached packets must be complemented without TCP (Transmission Control Protocol) and unicast transmission to assure the communication reliability.
- Requirement #3: The battery charge consumption of the communication terminals should be saved.

3.3 Design of the proposed system

3.3.1 Packet format

Three types of packet formats: Normal packet, Retransmission packet, Retransmission request packet are defined in this method as shown in Fig.1. The packet type is recognized by the “a. Packet type field”. The packet formats for the Normal packet and for the Retransmission packet are same except for the “a. Packet type field”. The roles of the each field are explained below:

a. Packet type (packet_type), 2Byte: The packet type (Normal packet, Retransmission packet, Retransmission Req packet) is recognized by this field.

b. Initiator node ID (init_id), 8Byte: The ID of the information initiator node, i.e. the information originator, is recognized by this field. Each node must be assigned a unique ID such as a MAC address of a wireless communication interface of the communication terminal.

c. Information ID (info_id), 2Byte: The field shows the ID of the information, which is given by the initiator node, is unique for each initiator node, and may be overlapped with that of other initiator nodes.

d. Packet sequence number (packet_seq), 4Byte: The sequence number of the packet in the information is recognized by the field, which is unique for each information.

e. Packet total (packet_total), 4Byte: The number of packets of which the information is recognized by the field.
f. Relaying node ID (relay_node_id), 8Byte: When a relaying node (that is a node except for the initiator node) relays the packet, the field is overwritten with the node ID of the relaying node. The field is used when a relaying node counts how many other nodes are relaying the information on the Relay decision state described later. Here, the field is empty when the packet is generated at an initiator node.

g. Data (data_payload): Divided data is contained in the field. The size of the field is determined by the parameter payload_size.

In addition, Retransmission Req packets contain the following.

h. Unreached packet sequence numbers: If a node finds that some packets are missing for received information, all the sequence numbers of unreached packets are described here, and informed to the neighboring nodes.

3.3.2 Operation of the proposed method

The operation of the proposed method is explained below. The proposed method operates according to the state transition diagram shown in Fig. 2. In the figure, the transition conditions are shown at the side of the arrows. Here, the “transmission” means broadcast transmission.

i. Initiation state: information generation: When a node generates information, the state transits from “Null state” to “Initiation state”. Initiator divides information into multiple packets according to payload_size. Each packet contains Initiator ID and Information ID. Then, the state transits to “Sending state” after the waiting time which is determined by Random(min, max).
Table 1 Parameters for each delivery method

<table>
<thead>
<tr>
<th>c_threshold</th>
<th>relay_threshold</th>
<th>send_interval</th>
<th>Range of Random(min, max)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SF</td>
<td>-</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td>4</td>
<td>-</td>
<td>0.064 s (0.064, 0.640) s</td>
</tr>
<tr>
<td>C3</td>
<td>3</td>
<td>-</td>
<td>0.064 s (0.064, 0.640) s</td>
</tr>
<tr>
<td>C2</td>
<td>2</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Ba4</td>
<td>-</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Ba3</td>
<td>-</td>
<td>3</td>
<td>0.064 s (0.064, 0.640) s</td>
</tr>
<tr>
<td>Ba2</td>
<td>-</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Ba1</td>
<td>-</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Bb4</td>
<td>-</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Bb3</td>
<td>-</td>
<td>3</td>
<td>0.032 s (0.032, 0.320) s</td>
</tr>
<tr>
<td>Bb2</td>
<td>-</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Bb1</td>
<td>-</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

ii. Sending state: information sending and relaying: The initiator node sends the packets sequentially with a fixed time interval which is determined by send_interval. After sending all the packets, the state transits to “Retrans. wait. state”.

iii. Packet receiving state: As shown in Fig.3, when a node receives a packet which is a part of information which has not been received yet (That is distinguished by the combination of init_ID and info_ID.), the state transits from “Null state” to “Receiving state”. Then the node waits for req_wait_time which is determined by the expression (1). Here, num_of_packets means the total number of packets for the information, which is described on packet_total field in the each packet. Random_max means the maximum value of Random(min, max).

\[
req\_wait\_time = 2 \times (send\_interval \times num\_of\_packets) + Random\_max \tag{1}
\]

During the period, if the node received all the packets of the information, the node break req_wait_time immediately. Then the state transits to “Relay decision state” shown in Fig.4. If the node does not receive the packets of the information (owing to data frame collisions or other problems) after expiration of req_wait_time, the state transits to “Retrans req state” shown in Fig.5 to request to retransmit the unreached packets. Here, the number of times the state transits to “Retrans req state” is limited to req_threshold. If the number of times reaches the threshold, the state transits to “Null state”.

This means that the retransmission requests are not transmitted anymore, and the information receiving is failed.

iv. Relay decision (the node decides whether to relay the information or not): The node waits for relay_wait_time which is determined by Random(min, max). During the period, the node counts the number of nodes relaying the same information (relaying node), which is distinguished by the combination of init_ID, info_ID, and relay_node_ID. After the period, if the number of relaying nodes is not reached to relay_threshold, the state transits to “Sending state”. As shown in Fig.6, the node relays the information.

v. Retrans. req. state: The node waits a period which is determined by Random(min, max). Then the node transmits Retransmission Req packet(s) in which sequence number(s) of unreached packet(s) shown in Fig.7, and the state transits “Receiving state”.

vi. Retrans. wait. State: The node waits for retrans_wait_time which is determined by the expression (2).

\[
retrans\_wait\_time = 2 \times num\_of\_packets \times send\_interval + 2 \times Random\_max \tag{2}
\]

vii. Retrans. send. State: The node transmits retransmission packet(s) are indicated in retransmit by Retransmission Req packet(s) with the fixed interval time send_interval. Here, the node retransmits the packet(s) only once even if it receives multiple Retransmission Req Packet(s), during retrans_wait_time period at the previous state “Retrans. wait. state”. When the node finishes retransmitting the retransmission packets, the state transits to “Retrans. wait. state”.

4 Evaluation by the simulations

Although we have shown the possible effectiveness of our proposal in [1], the evaluation made was preliminary, to make the evaluation more in detail, this section shows the performance comparison of BBISS with the existing methods through the network simulator OPNET [5].

4.1 Operation of the proposed method

The simulation conditions are described below. The simulation area is defined as 1000m×600m. We assume networks with 100, 200, 400, and 800 nodes. The initial positions of the nodes are set to be random. All nodes move at a speed of [0.00, 4.00] m/s according to the random waypoint model. This model is based on the human walking and running speed. The configuration of each node is described below. The node Mac layer is IEEE802.11b with the data rate...
of 11 Mbps. The communication range is 150m radius. The information generated by initiator node is delivered by Simple Flooding (SF), Counter-based scheme (CF), or BBISS, using the parameter shown in Table 1. Here, in CF, 3 sets of parameters, C4-C2 are determined, and in BBISS, 8 sets of parameters, Ba4-Ba1 and Bb4-Bb1 are determined. In SF and C2, the interval time of Random(mix, max) is inserted between packets when the node relays a packet. packet_size is 1024Byte.

In addition to the above, the following 2 conditions, Simulations #1 and #2, are assumed. In Simulation #1, 5% nodes of all nodes are information initiator nodes. Each initiator node generates one information data of 20kByte (i.e. 20 packets). In Simulation #2, 1% nodes of all nodes are information initiator nodes. Each initiator node generates one information data of 100kByte (i.e. 100 packets). Total number of packets initiated is same between Simulations #1 and #2.

4.2 Evaluated items

The average values of the following items (i)-(v) are found for 100 simulation runs at every value of the random seed. Here, the information data generated by only one initiator node is evaluated, so the information data generated by the other initiator nodes are as background traffic.

(i) The % of the information receiving nodes: The % of nodes that received the information to the total nodes successfully except for the initiator node of the information is shown. The higher the percentage is, the better the performance is.

(ii) The num. of receiving packets in the area: The total number of packets that are received by the nodes in the area is shown. Here “receiving” means that a data frame is received and transferred to upper layer. The smaller the value is, the better performance is.

(iii) (ii) / The number of information receiving nodes: To normalize (ii), (ii) is divided by the number of the information receiving nodes, \( \{ (i) \times \text{(total num. of nodes)} / 100 \} \). The smaller the value is, the better the performance is.

(iv) the num. of transmitted Retransmission Req packets in the area

(v) The average time of information delivery: The time between the information is generated at the initiator node and received at the receiving node (all packets that are composing the information are received) is calculated. The nodes could not receive the information are eliminated. The times at all receiving nodes are averaged. The smaller the value is, the better the performance is.

4.3 Simulation results and discussion

The simulation results are shown in Figs. 8–12.

(i) The % of the information receiving nodes

The simulation results for Simulation #1 and #2 are shown in the Figs. 8 (a) and (b), respectively. The % for BBISS was higher than that for SF and CF regardless of the number of nodes and simulation cases.

The results can be explained as follows. In SF and CF, a node decides whether to relay a receiving packets or not at each time when the node receives the packet. Since the operation caused many redundant relay-transmissions, many packets were lost due to data frame collisions. To make the matter worse, these methods did not equip the function to complement the lost packets. Therefore, these methods proved to have low percentage. On the other hand, BBISS outperformed SF and CF. In BBISS, the node determines on the performance or nonperformance of the information relaying at the time when the node completes the information receiving. The operation made the packet transmissions dispersive. In addition, BBISS performed the retransmission attempts. Therefore, BBISS proved to have the higher percentage than those of SF and CF.

(ii) the num. of receiving packets in the area

The simulation results for Simulation #1 and #2 are shown in the Figs. 9 (a) and (b), respectively. Although the results in BBISS depended on the parameter setting, some cases for BBISS proved to have the larger number than that for SF and CF. The reason seemed that the (i) for BBISS was increased. The more specific discussion will be shown in the evaluation of (iii).

(iii) (ii) / The number of information receiving nodes

The simulation results for Simulation #1 and #2 are shown in the Figs. 10 (a) and (b), respectively. Ba2, Ba1, Bb3, Bb2, and Bb1 for BBISS performed the same value as the best existing method C2 in both Simulations #1 and #2. Considering the results of (i), Ba2, Ba1, Bb3, Bb2, and Bb1 for BBISS performed high percentage of information receiving nodes without increasing the packet reception (traffic).

(iv) The num. of transmitted Retransmission Req packets in the area

The simulation results for Simulation #1 and #2 are shown in the Figs. 11 (a) and (b), respectively. In the cases where the numbers of nodes in the area were 100 and 200; those were small numbers, or low node density, the smaller the values of relay_threshold were, the more the Retransmission Req packets transmitted. The result can be
explained as follows. Since the node density was low, the cases with the small relay_threshold saved relaying, and then the packets were not delivered to the whole area. As the result, the number of Retransmission Req packets was increased. On the other hand, in the cases where the numbers of nodes in the area were 400 and 800; those were large numbers, or high node density, an opposite result to the above was given. The larger values of relay_threshold were, the more the Retransmission Req packets were transmitted. The result can be explained as an opposite reason to the above. Since the node density was high, the cases with the large relay_threshold did not save relaying, and the relay transmissions performed, and the packet losses due to data frame collisions were increased. As the result, to complement the packet losses, the number of Retransmission Req packets was increased.

As mentioned in the evaluation (i), since BBISS outperformed the existing method, we can conclude that the retransmission operation on BBISS can contribute to the information reachability. Although we have not discussed the number of Retransmission Req packets specifically, we have to optimize the parameters considering the number of Retransmission Req packets sent in the companion paper. That is because parameters which perform the smaller number of Retransmission Req packet may perform the less unreached packets or the higher information reachability and low traffic load.

(v) The average time of information delivery:

The simulation results for Simulation #1 and #2 are shown in the Figs. 12 (a) and (b), respectively. In Simulation #1 where the generated information size was small (20kByte), the results for Bb4-Bb1 showed the smaller results than the results for Ba4-Ba1 because Bb4-Bb1 set smaller values of Random (min, max) than Ba4-Ba1. Although the results for
BBISS were the same as or 1-2 seconds longer than the results for the best existing method C2, the differences were small, so that can be negligible. On the other hand, in Simulation #2, where the generated information size was large (100kByte), the results for BBISS showed larger than the results for the existing methods. Bb4-Bb1 for BBISS showed 5-10s longer than the results for the best existing method C2.

Considering the evaluation (i), we can conclude that BBISS showed the longer delivery times than the existing methods in return for the improvement of the information reachability. The shortening the delivery time for BBISS by adjustment of the parameters is our future issue to be tackled.

5 Conclusions

This study evaluates the performance of BBISS through the network simulations. The simulation results can conclude that BBISS outperform the existing flooding methods in terms of the information reachability without increasing on traffic. The parameter optimizations are our future issue to be tackled, some of them are shown in the companion paper.

6 Acknowledgments

This study has been supported by the Telecommunications Advancement Foundation, Japan.

7 References


