

Received December 27, 2017, accepted January 19, 2018, date of publication March 6, 2018, date of current version March 19, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2808356

A Neighbor-Based Probabilistic Broadcast Protocol for Data Dissemination in Mobile IoT Networks

WEI LIU ^{id}, KIYOHIDE NAKAUCHI, (Member, IEEE), AND YOZO SHOJI, (Member, IEEE)

Social-ICT Innovation Laboratory, National Institute of Information and Communications Technology, Tokyo 184-8795, Japan

Corresponding author: Wei Liu (wei_liu@nict.go.jp)

ABSTRACT The recent trend of implementing Internet of Things (IoT) applications is to transmit sensing data to a powerful data center and try to discover the valuable knowledge behind “Big Data” by various intelligent but resource-consuming algorithms. However, from the discussion with some industrial companies, it is understood that disseminating real-time sensing data to their nearby network-edge applications directly would produce a more economical design and lower service latency for some important smart city applications. Therefore, this paper proposes an efficient broadcast protocol to disseminate data in mobile IoT networks. The proposed protocol exploits the neighbor knowledge of mobile nodes to determine a rebroadcast delay that prioritizes different packet broadcasts according to their profits. An adaptive connectivity factor is also introduced to make the proposed protocol adaptive to the node density of different network parts. By combining the neighbor knowledge of nodes and adaptive connectivity factor, a reasonable probability is calculated to determine whether a packet should be rebroadcasted to other nodes or be discarded to prevent redundant packet broadcast. Extensive simulation results have validated that this protocol can improve the success ratio of packet delivery by 13% ~ 28% with a similar end-to-end transmission delay and network overhead of the most state-of-art approaches.

INDEX TERMS Internet of Things, mobile networks, neighbor coverage, probabilistic broadcast.

I. INTRODUCTION

Japan is one of the advanced countries in the world and it is facing to various representative social issues in healthcare, nursing-care, disaster prevention, energy saving and so on. The concept of smart city that utilizes Internet-of-Things (IoT) technologies to strengthen social infrastructures opens a new door for innovative solutions to the aforementioned issues and also creates a big commercial market. By observing this trend, the National Institute of Information and Communications Technology (NICT) of Japan is cooperating with some industrial companies to deploy a large-scale wireless IoT platform in the Sumida ward of Tokyo [1]. In this IoT platform, hundreds of different social “Things” like vending machines, beverage delivery trucks, and taxis are equipped with Wi-SUN [2] communication units that have a transmission range of several hundred meters in the real field. Thus, these wireless-enabled “Things” comprise large scale mobile wireless networks that cover a service region of 10 km², i.e., more than 70% area of the Sumida ward [1]. This kind of networks consists of social “Things” is named mobile IoT

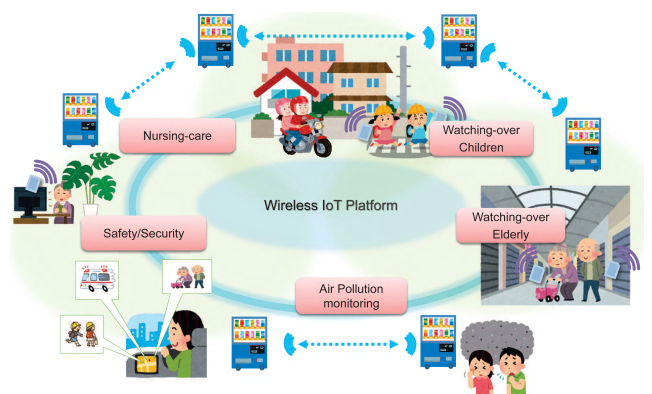


FIGURE 1. Wireless IoT platform.

networks in this paper. As depicted in Figure 1, this under constructed platform is expected to play an important role on gathering sensing data for different smart city applications like nursing care, traffic safety, air pollution monitoring, and watching over elderly/children.



FIGURE 2. An application scenario of traffic safety.

The recent trend of implementing IoT systems is to gather sensing data greedily, and transmit them to a powerful data center for analysis [3]. Since the amount of such data is typically too huge to be perceived, acquired, managed, and processed by traditional software/hardware tools within a tolerable time, they are named “Big Data” [4]. Various intelligent but resource-consuming approaches have been proposed by researchers to discover the valuable knowledge behind this kind of “Big Data” [5]–[8]. However, from the discussion with our industrial partners, it is understood that many kinds of sensing data are valuable only to human acting (living) in a local area and only when they are fresh. Figure 2 indicates an example scenario of alerting rushing children to nearby vehicle drivers. Assume that some sensing device is carried by a child (e.g., a sensor embedded in school-bag) and it transmits an alert signal when detecting the child is rushing in street. By disseminating this real-time signal in the local mobile IoT networks, nearby drivers can be notified to slow down their vehicles and avoid traffic accident. Thus, instead of simply uploading data to a data center, disseminating real-time sensing data to their nearby edge applications in local IoT networks directly would produce a more economical design and lower service latency for some important smart city applications.

Broadcasting is a suitable strategy to meet the previous demand. In a naive protocol named simple flooding (SF) [9], every wireless node simply rebroadcasts a newly received data packet and this operation can deliver the packet to all nodes in the networks (assuming that there is no separation). Therefore, this strategy is not only easy to be implemented in resource-constrained wireless nodes like vending machines and smart meters, but also adaptive to the dynamic network topology of mobile nodes like vehicles. However, it has been realized that SF protocol suffers an issue of “broadcast storm” [10], i.e., the massive redundant broadcasts of packets incur many collisions and interference in wireless channel which lead to excessive packet loss. Many improved protocols have been proposed by researchers to overcome the “broadcast storm” issue. Williams and Camp categorized these protocols into three classes: “probability-based”, “area-based”, and “neighbor-based” protocols [9], and Kim *et al.* further indicated that neighbor-based protocols perform best among them [11].

Since limiting the number of redundant packet broadcast is the key to alleviate the “broadcast storm” issue, and the neighbor-based strategy performs better than the area-based and probability-based ones, this paper proposes a

neighbor-based probabilistic broadcast (NPB) protocol to facilitate the data dissemination in mobile IoT networks. The main contributions of this paper are:

- (1) A novel scheme for determining the rebroadcast delay of a packet is proposed. The node which has more neighbors that have not received a data packet (named uncovered neighbors in this paper) triggers a shorter delay to rebroadcast the packet. This delay scheme not only decreases the probability for nodes to rebroadcast redundant packets, but also alleviates the collisions and interference in wireless channel that lead to excessive packet loss.
- (2) A novel scheme for determining the rebroadcast probability of a packet is also proposed. This scheme considers the statistics of uncovered neighbors and network connectivity to determine the probability for nodes to rebroadcast a packet. The resulting probability is composed of two parts:
 - a. Additional coverage ratio, that is the ratio of a transmitter’s neighbors that will be newly covered by one rebroadcast to its total amount of neighbors;
 - b. Adaptive connectivity factor, that considers the local density of nodes in different parts of networks to maintain the connectivity of packet delivery.

The rest of this paper is organized as follows: Section II reviews previous works. Section III describes the proposed NPB protocol in detail. Section IV shows the evaluation results of NPB protocol and its comparisons with other state-of-art approaches. Section V briefly discusses the deployment of protocol. Finally, conclusions are drawn and future work is mentioned in the last section.

II. RELATED WORK

Broadcasting is an effective strategy to disseminate data, but its overhead can be quite large, especially in highly dynamic mobile networks [12]. Tseng *et al.* studied conventional broadcast protocols analytically and experimentally [10], and showed that SF protocol incurs large network overhead and causes many problems such as redundant broadcasts and channel collisions. Keshavarz-Haddad *et al.* proved that solving the optimal broadcast strategy that minimizes the number of packet broadcast is equivalent to find out the minimum connected dominating set (MCDS) of network topology [13]. However, this optimal solution is little used in real situations, mainly because of two reasons: (1) Solving the MCDS of a given network topology is NP-hard [13] that indicates its complexity increases exponentially with the size of networks; and (2) It is hard to maintain an accurate MCDS of mobile wireless networks since their topologies constantly change according to the movements of network nodes. As a result, many researchers proposed their efficient heuristic algorithms to approximate the optimal broadcast behavior in mobile wireless networks.

Haas *et al.* proposed a gossip-based broadcast protocol [14], in which each node rebroadcasts a newly received packet with a pre-configured probability. Their

protocol can save up to 35% overhead compared with SF protocol. However, when the network density is high or the network traffic load is heavy, the improvement of their protocol is quite limited [12]. Kim *et al.* proposed a probabilistic broadcast protocol based on the coverage area and neighbor confirmation of wireless nodes [11]. Their scheme uses the coverage area of nodes to set rebroadcast probability, and uses the neighbor confirmation of nodes to guarantee the reachability of data packets. A considerable number of works such as [15]–[17], were proposed under a counter-based strategy. In these protocols, a counter C is initiated to record the total number of the same data packet received by a wireless node. This C is compared to a pre-configured threshold. If C reaches or exceeds the threshold, the probability for the node to rebroadcast such packet is zero, else its value approaches to one. Although these protocols can reduce network overhead, they also increase the end-to-end delay of packet delivery significantly. A location-based strategy was suggested in [18] and [19]. In these works, data packets are rebroadcasted based on the locations of wireless nodes, mainly by using GPS. As a result, many nodes can be excluded from the rebroadcast process that in turn enhances network performance. In contrast, using GPS not only increases the cost of wireless nodes, but also prevents indoor nodes from adopting these protocols.

Two recent effective broadcast protocols that inspire our work are the neighbor coverage-based probabilistic rebroadcast (NCPB) protocol proposed by Zhang *et al.* [20] and the neighbor-based dynamic connectivity factor routing (DCFR) protocol proposed by Ejmaa *et al.* [21]. NCPB protocol proposes a scheme for calculating the rebroadcast delay of a data packet, and allows a node which has more common neighbors with the previous transmission node to rebroadcast the packet first. After the delay expires, it decides whether a packet should be rebroadcasted or discarded based on the node's uncovered neighbors. DCFR protocol can be treated as a revision of NCPB protocol that eliminates the need of configuring global network parameters and improves the success ratio of packet delivery by 2% ~ 4% in most situations. Compared with the previous two protocols, our proposed NPB protocol adopts a reverse strategy to determine the rebroadcast delay of a packet. That is, the priority of a node to rebroadcast a packet decreases, when it has more common neighbors with the previous transmission node. As illustrated in Sect. IV, this novel delay scheme and its accompanied scheme for determining the rebroadcast probability of a packet enable NPB protocol to improve the success ratio of packet delivery by 13% ~ 28%, while retaining a similar end-to-end packet transmission delay and network overhead of NCPB and DCFR protocols. Ruiz and Bouvry provided a comprehensive survey of this research topic [22].

III. NEIGHBOR-BASED PROBABILISTIC BROADCAST PROTOCOL

As described in Sect. I, the key to optimize a broadcast protocol is to limit the number of redundant packet

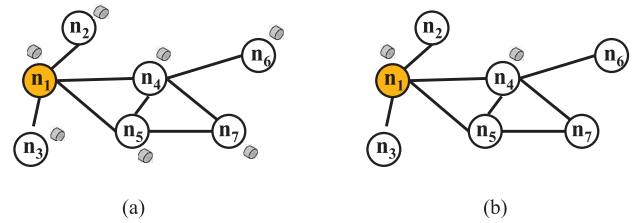


FIGURE 3. The comparison between two broadcast protocols. (a) Simple flooding protocol. (b) The optimal broadcast protocol.

broadcast. Figure 3 compares the performances of SF protocol and an optimal broadcast protocol that minimizes redundant packet transmission. By using SF protocol, when the node n_1 broadcasts a packet, its neighbors n_2 , n_3 , n_4 , and n_5 will receive this packet. Since this is a new packet to n_2 , n_3 , n_4 , and n_5 , they will rebroadcast this packet. Similarly, n_6 and n_7 will receive the packet and rebroadcast it again. Therefore, a total of 7 packets are broadcasted by 7 nodes in the network. However, Figure 3(b) illustrates that only two times of packet broadcast from n_1 and n_4 can disseminate this packet to all nodes in the network. It is easy to validate that n_1 and n_4 constitute the MCDS of the network topology shown in Figure 3, and this observation is consistent to the conclusion made in [13]. Due to the reasons explained in Sect. II, it is not smart to directly apply this optimal solution to dynamic mobile networks. Therefore, this section proposes a novel NPB protocol to implement efficient data broadcast in the mobile IoT networks discussed in this paper.

As scrutinized in the following parts of this section, NPB protocol uses the number of uncovered neighbors to determine the rebroadcast delay of a packet, and uses the additional coverage ratio of a packet and a metric named adaptive connectivity factor to determine its rebroadcast probability. The same to other neighbor-based broadcast protocols, NPB protocol requires a data packet to carry the neighbor list of its transmitter. Since this mechanism is not the main focus of this paper, NPB protocol adopts the same efficient mechanism used by NCPB [20] and DCFR [21] protocols to reduce the overhead of neighbor data transmission. The details of this mechanism are available in [20].

A. REBROADCAST DELAY

When a node n_i receives a data packet p_k (the suffix k indicates the sequence number of packet) from its previous transmitter n_p , it can use the neighbor list of n_p contained in p_k to estimate how many of its neighbors have not been covered by the p_k broadcasted by n_p . To quantify this, the uncovered neighbor set of n_i for p_k is given by

$$U_k(n_i) = N(n_i) - \{N(n_i) \cap N(n_p)\} - \{n_p\}, \quad (1)$$

where $N(n_i)$ and $N(n_p)$ are the neighbor sets of nodes n_i and n_p .

Due to the characteristic of broadcasting a packet, the node n_i may receive duplicate p_k from its neighbors other than n_p . Therefore, n_i can continuously update $U_k(n_i)$ with the

duplicate p_k from other neighbors. It is obvious that when $U_k(n_i)$ becomes empty, there is no need for n_i to rebroadcast p_k since all of its neighbors have received p_k already. This introduces a tradeoff on the rebroadcast delay of p_k . When rebroadcast delay is long, there is a long period for n_i to eavesdrop duplicate packets and empty its $U_k(n_i)$ to prevent redundant packet rebroadcast, while this also increases the end-to-end delay of data dissemination. Thus, NPB protocol determines the rebroadcast delay of p_k as

$$D_k(n_i) = \begin{cases} \text{cancel this rebroadcast,} \\ \quad |U_k(n_i)| = 0; \\ \frac{d_{max}}{1 - a(n_i)} \times |U_k(n_i)| - \frac{d_{max} \times a(n_i)}{1 - a(n_i)}, & (2) \\ \quad 1 \leq |U_k(n_i)| < a(n_i); \\ 0, & |U_k(n_i)| \geq a(n_i), \end{cases}$$

where $|\cdot|$ is the size of a set, d_{max} is the upper limit of rebroadcast delay, and $a(n_i)$ is the average number of n_i 's neighbors. Due to the mobility of n_i , $a(n_i)$ is a dynamic variable and the moving window average strategy described in [23] is used to calculate it:

$$a_{t+\Delta t}(n_i) = \beta \times a_t(n_i) + (1 - \beta) \times a_{\Delta t}(n_i), \quad (3)$$

where $a_t(n_i)$ indicates the average number of n_i 's neighbors at the time point t , and $\beta \in [0, 1]$ is a convergence factor. It can easily be proved that, as time passes, the moving window average strategy ensures the convergence of $a(n_i)$, and this proof is independent of β [23]. Indeed, the value of β represents a trade-off, i.e., a larger β reduces the error of $a(n_i)$ but increases the time to reach its steady state, and vice versa.

The scheme presented by Eq. (2) uses a truncated linear function to determine the rebroadcast delay of p_k . When n_i has a relatively large number of uncovered neighbors (i.e., $|U_k(n_i)| \geq a(n_i)$), it will rebroadcast p_k without any delay. Otherwise, the rebroadcast delay of p_k linearly increases to d_{max} (when $|U_k(n_i)| = 1$) with the decrease of its uncovered neighbors. This scheme can be understood in the following way: When n_i receives p_k but $|U_k(n_i)|$ is small, the profit for n_i to rebroadcast p_k is small since it can only deliver p_k to a few additional nodes. From another point of view, since $|U_k(n_i)|$ is small, there is a higher possibility to empty $U_k(n_i)$ by letting n_i eavesdrop duplicate p_k for a longer time. Thus, this scheme assigns a longer delay to n_i when $U_k(n_i)$ is small, and vice versa. Taking the network topology depicted in Figure 3(b) as an example, and assumes that $a(n_4) = a(n_5)$ at the time of packet transmission. When n_4 and n_5 receive p_k from n_1 simultaneously, n_4 will rebroadcast p_k earlier than n_5 , because it has two uncovered neighbors (i.e., n_6 and n_7), while n_5 has only one (i.e., n_7). After eavesdropping the p_k rebroadcasted by n_4 , n_5 realizes that there is no need to broadcast p_k again because all of its neighbors have received p_k already, i.e., $|U_k(n_5)| = 0$.

Finally, it is interesting to note that this scheme for determining the rebroadcast delay of a packet is reverse to the one

adopted by NCPR [20] and DCFR [21] protocols. Both NCPR and DCFR protocols let a node which has more common neighbors with the previous transmission node to rebroadcast a data packet first. According to Eq. (1), more common neighbors (i.e., a larger $|N(n_i) \cap N(n_p)|$) usually indicates a smaller uncovered neighbor set (i.e., a smaller $|U_k(n_i)|$). According to Eq. (2), a smaller $|U_k(n_i)|$ leads to a longer rebroadcast delay in NPB protocol. As illustrated in Sect. IV, this completely different strategy leads to better performances of data dissemination in different network settings.

B. REBROADCAST PROBABILITY

A node that has a longer rebroadcast delay may eavesdrop duplicate data packets from other nodes that have shorter ones. For example, when the node n_i receives a duplicate p_k from its neighbor n_j , it knows that how many of its neighbors have been additionally covered by this packet. Thus, n_i can update its uncovered neighbor set according to the neighbor list of n_j , by using Eq. (1). After updating $U_k(n_i)$, n_i discards this duplicate p_k since it has been broadcasted already. It should be noted that NPB protocol does not try to update the rebroadcast delay of p_k according to the updated (i.e., smaller) $U_k(n_i)$. This is because of two reasons: (1) It prevents the continuous increase of 1-hop rebroadcast delay when n_i receives duplicate packet every time; and (2) It prevents the frequent computation of protocol timer and keeps NPB protocol simple and stupid, i.e., the general principle of industrial development. When the timer of rebroadcast delay expires, n_i obtains the final $U_k(n_i)$ that contains its neighbors which have not received p_k yet. This final $U_k(n_i)$ is used to determine the probability for n_i to rebroadcast p_k .

Let the additional coverage ratio of n_i to rebroadcast p_k denote the ratio of n_i 's neighbors that will newly receive p_k to the total number of n_i 's neighbors,

$$C_k(n_i) = \frac{|U_k(n_i)|}{|N(n_i)|}. \quad (4)$$

With the increase of $C_k(n_i)$, more nodes will benefit from n_i 's rebroadcast. Thus, the probability for n_i to rebroadcast p_k should be increased accordingly. Besides, Xue and Kumar have derived that when every node connects to more than $5.1774 * \log(N_{node})$ of its neighbors, the probability of a network being connected is approaching one with the increase of N_{node} [24], where N_{node} denotes the total number of nodes in the network. This means, in order to disseminate p_k to all nodes in the network, at least $5.1774 * \log(N_{node})$ neighbors of n_i need to receive and rebroadcast this packet, i.e., $|N(n_i) - U_k(n_i)| > 5.1774 * \log(N_{node})$. Thus, a metric named adaptive connectivity factor is defined as

$$A_k(n_i) = \frac{5.1774 * \log(N_{node})}{|N(n_i) - U_k(n_i)|}. \quad (5)$$

The value of N_{node} used in this equation can either be configured by a system administrator in advance as Zhang *et al.* suggested in NCPR protocol [20], or be estimated by the distributed algorithm proposed by Ejmaa *et al.* in DCFR

protocol [21]. From Eq. (5), it can be observed that when the number of n_i 's neighbors that have received p_k is larger than $5.1774 * \log(N_{node})$, $A_k(n_i)$ is less than one. This means that n_i moves in the dense area of network, and p_k can be disseminated to the other parts of network without the rebroadcast from n_i . Otherwise, it means that n_i moves in the sparse area of network, and it prefers rebroadcasting p_k to maintain the connectivity of packet dissemination.

The probability for n_i to rebroadcast p_k is given by the multiplication of its additional coverage ratio and adaptive connectivity factor,

$$P_k(n_i) = C_k(n_i) \times A_k(n_i). \quad (6)$$

In Eq. (6), $C_k(n_i)$ indicates how many next-hop nodes will benefit from the rebroadcast of p_k from n_i , but it does not consider the connectivity of packet dissemination. $A_k(n_i)$ compensates this drawback by increasing the rebroadcast probability of n_i when it moves in the sparse area of network, while decreasing this probability when it moves in the dense area of network. As illustrated by the evaluation results in Sect. IV, the combination of these two factors makes NPB protocol adaptive to different network settings. It should be noted that although the value of $P_k(n_i)$ in Eq. (6) may become greater than 1 in some cases, this will not affect the behavior of NPB protocol. It only indicates that the local node density of network around n_i is so low that n_i should rebroadcast this packet to maintain the connectivity of packet dissemination.

Algorithm 1 NPB Protocol

```

1  if ( $n_i$  receives a new packet  $p_k$ )
2  initializes the uncovered neighbor set by using Eq. (1);
3  calculates the rebroadcast delay by using Eq. (2);
4  sets a timer to rebroadcast  $p_k$  according to the delay;
5  end if;
6  while ( $n_i$  receives duplicate  $p_k$  during the delay)
7  updates the uncovered neighbor set by using Eq. (1);
8  discards duplicate  $p_k$ ;
9  end while;
10 if (the timer for rebroadcasting  $p_k$  expires)
11 calculates the additional coverage ratio by using
    Eq. (4);
12 calculates the adaptive connectivity factor by using
    Eq. (5);
13 calculates the rebroadcast probability by using
    Eq. (6);
14 if ( $random(0, 1) \leq$  the rebroadcast probability)
15 rebroadcasts  $p_k$ ;
16 else
17 discards  $p_k$ ;
18 end if;
19 end if;
```

Finally, since network nodes with NPB protocol only make rebroadcast decisions according to their local neighbor information and Eqs. (1 - 6) are linear or truncated linear functions,

TABLE 1. The default parameters of simulations.

Simulator	ns-3 v3.26
Simulation area	1000 × 1000 m
Number of nodes	100
Packet size	100 bytes
Packet transmission interval	1 sec
Transmission range of nodes	250 m
Link bandwidth	1 Mbps
Mobility model	Random way point
Mobility speed	1 m/s
Pause time	0 sec
Convergence factor β	0.8
Number of generated packets	200 packets per node

the overall computational complexity of NPB protocol is linear to the number of nodes in the networks, i.e., $O(N_{node})$. The pseudo-codes of NPB protocol are listed in Algorithm 1.

IV. PERFORMANCE EVALUATIONS

This section presents the evaluation results of the proposed NPB protocol. In the following evaluations, NPB protocol was compared with NCPR [20], DCFR [21], and SF [9] protocols by using the ns-3 network simulator. NCPR and DCFR are two state-of-art broadcast protocols introduced in recent literatures, while SF is a conventional one and can be treated as the benchmark for other protocols.

Mobile nodes were uniformly distributed in a rectangular area at the beginning of simulations. The distributed coordination function (DCF) of IEEE 802.11 standard was used as MAC layer protocol, and a radio channel model of 1Mbps was used to verify the performances of different protocols in resource-constrained IoT environment. The transmission range of every node was 250 meters, and they moved at a speed of 1 m/s according to the random waypoint mobility model [25]. The convergence factor β in Eq. (3) for calculating the average number of neighboring nodes was set to 0.8. According to the preliminary simulation results, the performance of NPB protocol is not sensitive to β when it is greater than 0.5. Thus, the impact of β is not further discussed in the remaining parts of this section. The max delay (d_{max}) used by NPB, NCPR and DCFR protocols to determine their rebroadcast delay was 128 ms, which is equivalent to the upper limit of the random back-off time in many IEEE 802.11 implementations. In every trial of simulation, each mobile node generated 200 data packets with a default interval of 1 second and disseminated them to other nodes by using different broadcast protocols. Table 1 summarizes the default parameters of simulations, and the following results are an average of 100 experimental trials.

The previous mentioned four broadcast protocols were compared based on three criteria:

Average packet delivery ratio: the average ratio of the data packets that are successfully received by nodes to the total amount of data packets that are generated in simulation.

Average end-to-end delay: the average delay of successfully received data packets from the time they are originally generated in simulation.

Network overhead: the average amount of data transmitted by nodes in simulation. To preserve the fairness of comparison, the total amount of data transmission is measured instead of the number of transmitted data packets. This is because neighbor-based broadcast protocols like NPB, NCPR, and DCFR require a data packet to carry the neighbor information of its transmitter.

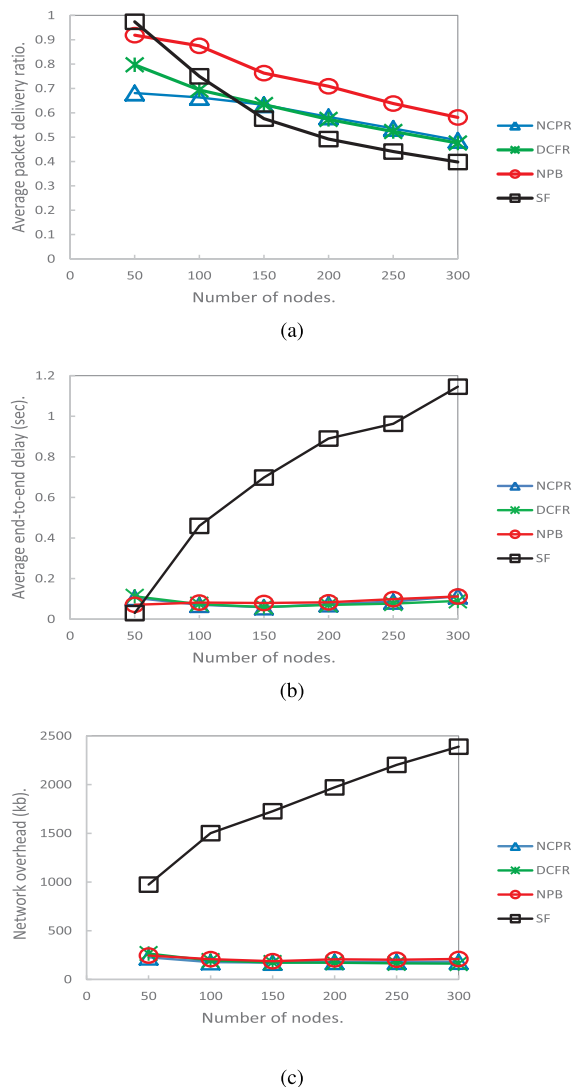


FIGURE 4. Node density. (a) The packet delivery ratios of different protocols. (b) The end-to-end delays of different protocols. (c) The network overheads of different protocol

Figure 4 presents the performances of different broadcast protocols with varied node density. When the number of nodes in the area was small, there was little collision and interference in wireless channel. SF protocol achieved the best packet delivery ratio since it broadcasted more redundant packets than other improved protocols, and these redundant packets improved the possibility of successful packet

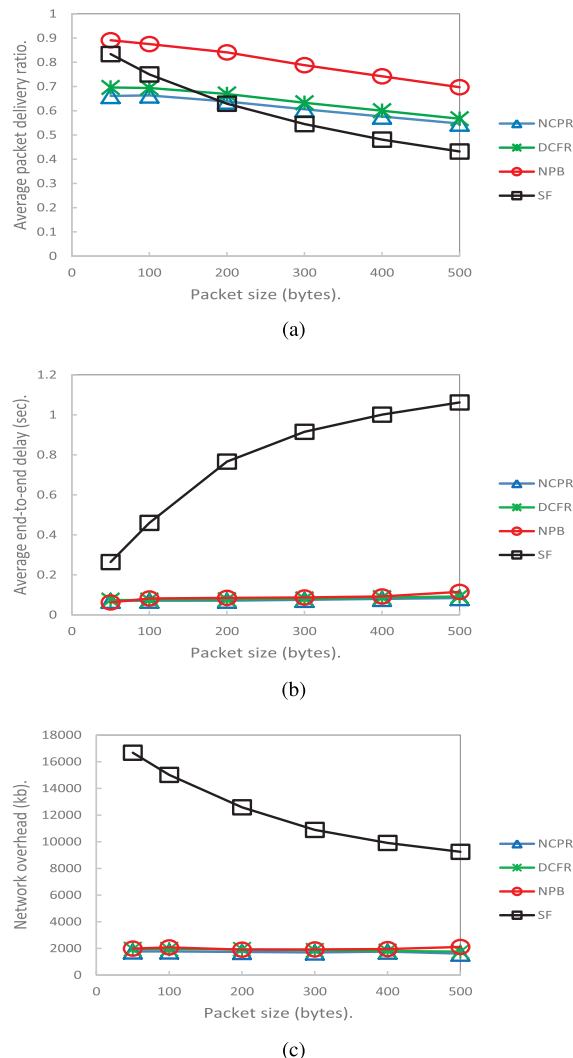
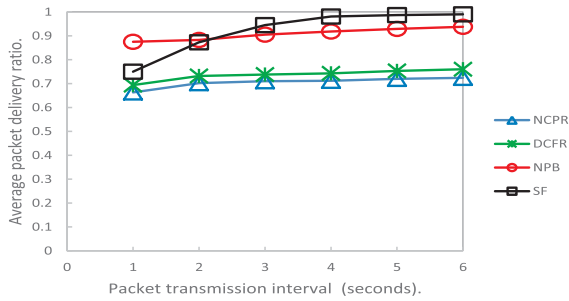
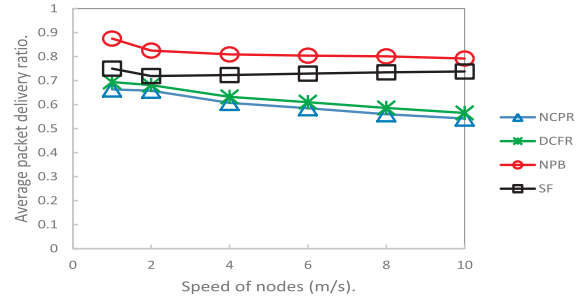


FIGURE 5. Packet size. (a) The packet delivery ratios of different protocols. (b) The end-to-end delays of different protocols. (c) The network overheads of different protocols.

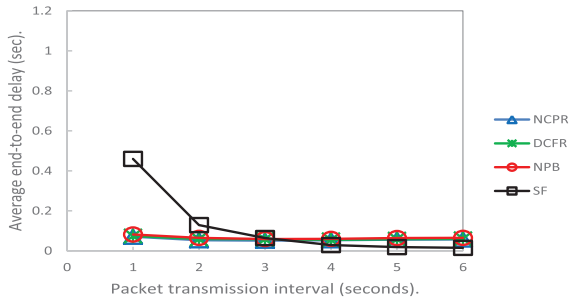
delivery. However, with the increase of node density, many redundant packets produced by SF protocol led to severe channel collisions and interference that not only degraded the success ratio of packet delivery but also increased the end-to-end delay of packet delivery drastically. Compared with the state-of-art NCPR and DCFR protocols, NPB protocol improved the success ratio of packet delivery by 13% ~ 25%, and retained a similar level of network overhead. It proves that NPB protocol is more effective on managing different packet broadcasts, e.g., it prioritizes a broadcast that can deliver a packet to more uncovered neighbors, and it increases broadcast probability when a node moves in the sparse area of networks. Figure 4(b) shows that NPB protocol resulted in a shorter end-to-end delay than NCPR and DCFR protocols when the node density of network was low, and a little longer delay when the node density of network was high. Recall that both NCPR and DCFR protocols allow a node which has more common neighbors with the previous transmission node to rebroadcast a data packet shortly. With the increase



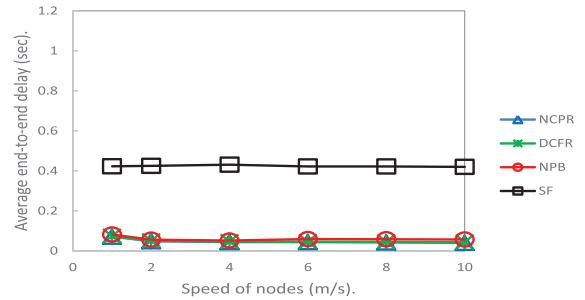
(a)



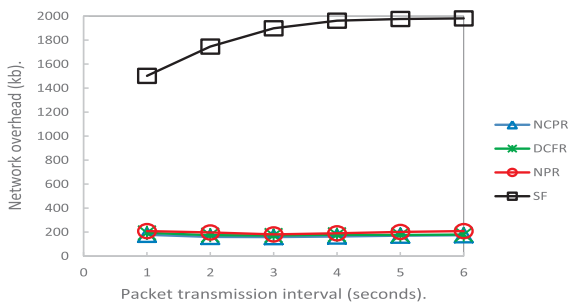
(a)



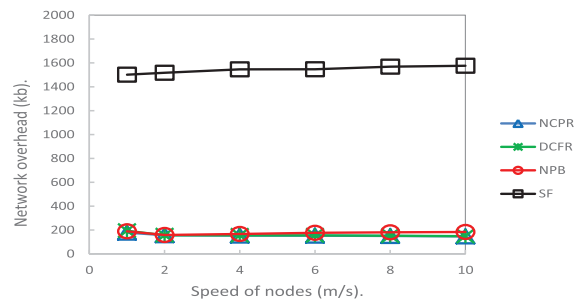
(b)



(b)



(c)



(c)

FIGURE 6. Packet transmission interval. (a) The packet delivery ratios of different protocols. (b) The end-to-end delays of different protocols. (c) The network overheads of different protocols.

of node density, the common neighbors between receiving and transmission nodes increase, and their schemes tend to reduce the delay of rebroadcasting a packet. As described in Sect. III, NPB protocol adopts a reverse scheme against NCPR and DCFR protocols, and it tends to increase the end-to-end delay of packet delivery when node density increases. Nevertheless, since NPB protocol can adaptively decrease the broadcast probability of nodes in dense networks, the end-to-end delays of these three protocols were at the same level in any setting of node density.

Figure 5 presents the performances of different broadcast protocols with varied packet size. The performance of SF protocol degraded drastically with the increase of packet size. This was because nodes need a longer time to broadcast a bigger packet, and many redundant packets produced by SF protocol severely congested wireless channel. NPB protocol achieved the best packet delivery ratio among four protocols. Compared with NCPR and DCFR protocols, it improved the packet delivery ratio by 16% ~ 22%, with a very little

FIGURE 7. Speed of nodes. (a) The packet delivery ratios of different protocols. (b) The end-to-end delays of different protocols. (c) The network overheads of different protocols.

higher and steady network overhead. It should be noted that the two metrics, i.e., packet delivery ratio and network overhead, must be jointly considered. Purely focusing on reducing network overhead is meaningless. For example, it is easy to achieve a very low network overhead if every node never rebroadcasts any data packet, while this strategy leads to an unacceptable packet delivery ratio. Similar to the simulation results of varied node density, there was no obvious difference among the end-to-end delays of these three protocols. It is clear that this simulation on increasing packet size is equivalent to the simulation on decreasing the link bandwidth of mobile nodes, since both of them increase the time to broadcast a packet.

Figure 6 presents the performances of different protocols with varied packet transmission interval. With the increase of packet transmission interval, there were less data packets need to be processed in a time unit and this also alleviated the collisions and interference in wireless channel. Similar to the previous discussions under a low node density, SF

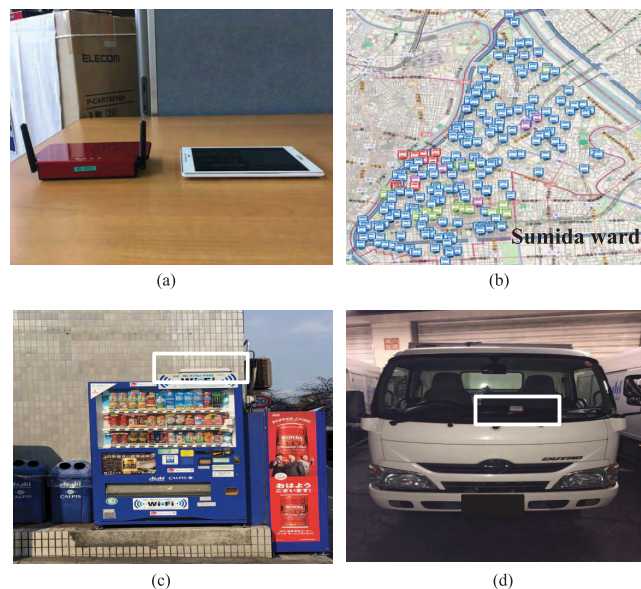


FIGURE 8. The under constructed IoT platform in Tokyo. (a) Wireless router (box version and tablet version). (b) Vending machine distribution in the Sumida ward of Tokyo. (c) Wireless-enabled vending machine. (d) Wireless-enabled vehicle.

protocol achieved a better packet delivery ratio than other improved protocols under the situation of a long interval, since its redundant packet broadcasts improved the possibility of successful packet delivery. The packet delivery ratio of NPB protocol was near to that of SF protocol. Compared with NCPD and DCFD protocols, NPB protocol improved the success ratio of packet delivery by 17% ~ 28%, and retained a similar level of end-to-end delay and network overhead.

Figure 7 presents the performances of different protocols with varied speed of mobile nodes. When the speed of mobile nodes increases, the neighboring node list maintained by NPB, DCFD, and NCPD protocols changes more frequently and becomes less accurate. Since these three protocols make their rebroadcast decisions based on the neighbor list of packet transmitter, their performances degraded with the increase of mobility speed. Nevertheless, NPB protocol achieved the best performance among all protocols even when nodes moved at a speed of 10 m/s, i.e., a usual upper limit of vehicle speed on urban roads. Compared with NCPD and DCFD protocols, it improved the success ratio of packet delivery by 18% ~ 27% under a similar level of end-to-end delay and network overhead.

V. DISCUSSIONS ON PROTOCOL DEPLOYMENT

Figure 8 shows an overview of our under constructed IoT platform in the Sumida ward of Tokyo. At the current stage of demonstration experiment, 100 vending machines and 65 vehicles including beverage delivery trucks and taxis are equipped with wireless routers that have a coverage radius of 250 ~ 300 meters in the real field. Every few second, these wireless nodes gather various sensing data from the sensors deployed in their surrounding environments, and encapsulate them into a small data packet of 100 bytes. These data packets

are disseminated in the mobile IoT networks by SF protocol and processed by different smart city applications deployed at the edges of networks. As proved by the previous simulation results, SF protocol is not scalable to the increase of network nodes in this platform. Thus, there is a strong intention for us to replace it with the proposed NPB protocol in near future. The ultimate objective of this project is to construct an IoT platform covers a service region of 10 km². Since it may be not necessary to disseminate local sensing data in such a wide area, the time-to-live (TTL) value of data packets can be used to constrain their dissemination scope within several hundred meters or 1 ~ 2 kilometers. Finally, to make this platform compatible with existing “Big Data” systems, a few gateway routers may be deployed in networks to gather disseminated sensing data and deliver them to a remote data center for further analysis. This kind of hybrid mobile IoT networks is also economical to industrial companies because it is very expensive to maintain an internet access for every wireless node separately.

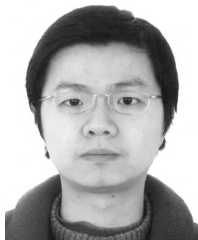
VI. CONCLUSIONS AND FUTURE WORK

This paper proposed a neighbor-based probabilistic broadcast protocol to facilitate the data dissemination in mobile IoT networks. The proposed protocol uses the number of uncovered neighbors to determine the rebroadcast delay of a data packet, and utilizes the additional coverage ratio of the packet and a metric named adaptive connectivity factor to determine its rebroadcast probability. Extensive simulation results have validated that the proposed protocol successfully improves packet delivery ratio by 13% ~ 28% with a similar end-to-end delay and network overhead of the most state-of-art approaches. For future work, there is a strong intention for us to implement the proposed protocol in an under constructed IoT platform in Tokyo.

REFERENCES

- [1] Y. Shoji, K. Nakauchi, and W. Liu, “Community-based wireless IoT infrastructure using ubiquitous vending machines,” in *Proc. Cloudification Internet Things*, Nov. 2016, pp. 1–6.
- [2] H. Harada, K. Mizutani, J. Fujikawa, K. Mochizuki, K. Obata, and R. Okumura, “IEEE 802.15. 4g based Wi-SUN communication systems,” *IEICE Trans. Commun.*, vol. 100, no. 7, pp. 1032–1043, 2017.
- [3] Q. Zhang, L. T. Yang, Z. Chen, P. Li, and M. J. Deen, “Privacy-preserving double-projection deep computation model with crowdsourcing on cloud for big data feature learning,” *IEEE Internet Things J.*, to be published, doi: 10.1109/JIOT.2017.2732735.
- [4] M. Chen, S. Mao, and Y. Liu, “Big data: A survey,” *Mobile Netw. Appl.*, vol. 19, no. 2, pp. 171–209, Apr. 2014.
- [5] Q. Zhang, M. Lin, L. T. Yang, Z. Chen, and P. Li, “Energy-efficient scheduling for real-time systems based on deep q-learning model,” *IEEE Trans. Sustain. Comput.*, to be published, doi: 10.1109/TSUSC.2017.2743704.
- [6] W. Liu, Y. Shoji, and R. Shinkuma, “Logical correlation-based sleep scheduling for WSNs in ambient-assisted homes,” *IEEE Sensors J.*, vol. 17, no. 10, pp. 3207–3218, May 2017.
- [7] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, “PPHOPCM: Privacy-preserving high-order possibilistic c-means algorithm for big data clustering with cloud computing,” *IEEE Trans. Big Data*, to be published, doi: 10.1109/TBDDATA.2017.2701816.
- [8] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, “An improved deep computation model based on canonical polyadic decomposition,” *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2017.2701797.

- [9] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proc. 3rd ACM Int. Symp. Mobile Ad Hoc Netw. Comput.*, 2002, pp. 194–205.
- [10] Y.-C. Tseng, S.-Y. Ni, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," *Wireless Netw.*, vol. 8, nos. 2–3, pp. 153–167, 2002.
- [11] J.-S. Kim, Q. Zhang, and D. P. Agrawal, "Probabilistic broadcasting based on coverage area and neighbor confirmation in mobile ad hoc networks," in *Proc. IEEE Global Telecommun. Conf. Workshops (GlobeCom Workshops)*, Nov./Dec. 2004, pp. 96–101.
- [12] J. Abdulai, M. Ould-Khaoua, and L. M. Mackenzie, "Improving probabilistic route discovery in mobile ad hoc networks," in *Proc. 32nd IEEE Conf. Local Comput. Netw. (LCN)*, Oct. 2007, pp. 739–746.
- [13] A. Keshavarz-Haddad, V. Ribeiro, and R. Riedi, "Color-based broadcasting for ad hoc networks," in *Proc. 4th Int. Symp. Modeling Optim. Mobile, Ad Hoc Wireless Netw.*, Feb./Mar. 2006, pp. 1–10.
- [14] Z. J. Haas, J. Y. Halpern, and L. Li, "Gossip-based ad hoc routing," *IEEE/ACM Trans. Netw.*, vol. 14, no. 3, pp. 479–491, Jun. 2006.
- [15] A. Mohammed, M. Ould-Khaoua, and L. Mackenzie, "An efficient counter-based broadcast scheme for mobile ad hoc networks," in *Proc. Eur. Perform. Eng. Workshop*, 2007, pp. 275–283.
- [16] M. B. Yassein, S. F. Nimer, and A. Y. Al-Dubai, "A new dynamic counter-based broadcasting scheme for mobile ad hoc networks," *Simul. Model. Pract. Theory*, vol. 19, no. 1, pp. 553–563, 2011.
- [17] M. Bakhouya, J. Gaber, and P. Lorenz, "An adaptive approach for information dissemination in Vehicular Ad hoc Networks," *J. Netw. Comput. Appl.*, vol. 34, no. 6, pp. 1971–1978, Nov. 2011.
- [18] Y.-B. Ko and N. H. Vaidya, "GeoTORA: A protocol for geocasting in mobile ad hoc networks," in *Proc. IEEE Int. Conf. Netw. Protocols*, Nov. 2000, pp. 240–250.
- [19] W. Liu, K. Nakauchi, and Y. Shoji, "A location-aided flooding mechanism in community-based IoT networks," in *Proc. IEEE 7th Annu. Comput. Commun. Workshop Conf. (CCWC)*, Jan. 2017, pp. 1–6.
- [20] X. M. Zhang, E. B. Wang, J. J. Xia, and D. K. Sung, "A neighbor coverage-based probabilistic rebroadcast for reducing routing overhead in mobile ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 12, no. 3, pp. 424–433, Mar. 2013.
- [21] A. M. E. Ejmaa, S. Subramaniam, Z. A. Zukarnain, and Z. M. Hanapi, "Neighbor-based dynamic connectivity factor routing protocol for mobile ad hoc network," *IEEE Access*, vol. 4, pp. 8053–8064, 2016.
- [22] P. Ruiz and P. Bouvry, "Survey on broadcast algorithms for mobile ad hoc networks," *ACM Comput. Surv.*, vol. 48, no. 1, 2015, Art. no. 8.
- [23] H. Dang and H. Wu, "Clustering and cluster-based routing protocol for delay-tolerant mobile networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 6, pp. 1874–1881, Jun. 2010.
- [24] F. Xue and P. R. Kumar, "The number of neighbors needed for connectivity of wireless networks," *Wireless Netw.*, vol. 10, no. 2, pp. 169–181, Mar. 2004.
- [25] D. B. Johnson and D. A. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*. Boston, MA, USA: Springer, 1996, pp. 153–181.



WEI LIU received the B.E. and M.E. degrees in software engineering from Chongqing University, China, in 2006 and 2009, respectively, and the Ph.D. degree in communications and computer engineering from Kyoto University, Japan, in 2015. He was with PICC Corporation as an ICT Engineer. Since 2015, he has been a Researcher with the National Institute of Information and Communication Technology, Japan. His research interests include mobile cloud computing, IoT, and smart city applications. He received the Best Paper Award for his paper at the IEEE CCWC 2017 and the Young Author Recognition for his paper at the ITU Kaleidoscope 2013.



KIYOHIDE NAKAUCHI (S'01–M'04) received the B.E., M.E., and Ph.D. degrees from The University of Tokyo, Tokyo, Japan, in 1998, 2000, and 2003, respectively, all in information and communication engineering. He joined the Communications Research Laboratory [currently National Institute of Information and Communications Technology (NICT)], Ministry of Posts and Telecommunications, Tokyo, in 2003, where he is currently a Senior Researcher with the Open Innovation Promotion Headquarters. Since 2003, he has been actively involved in research and development activities in the field of high-performance transport protocols, overlay network, network virtualization, and mobile networks. He is a member of the Association for Computer Machinery and the Institute of Electrical, Information and Communication Engineers.



YOZO SHOJI (S'98–M'99) joined the Communications Research Laboratory (CRL), Ministry of Posts and Telecommunications, Japan, in 1999. Since then, he has been involved in research in millimeter-wave and optical communications systems and made a lot of contributions to the standardization for the 60-GHz band in IEEE802.15.3c. In 2000, he invented the millimeter-wave self-heterodyne system and succeeded in a 60-GHz band wireless transmission of OFDM-based digital TV broadcast signals for the first time in the world. In 2010, he received the Excellent Young Researchers Overseas Visit Program Fellowship from the Japan Society for the Promotion of Science and spent one year as a Visiting Researcher with the University College London, U.K. He is currently the Director of the Social-ICT Innovation Laboratory, NICT (formerly CRL) and engaging the research for the community-based wireless IoT network system. He is a Senior Member of the Institute of Electrical, Information and Communication Engineers (IEICE), Japan. He was a recipient of the IEICE Young Researchers Award in 2000, the IEICE Electronics Society Award in 2007, the Young Scientists Prize in the Commendation for Science and Technology from the Minister of Education, Culture, Sports, Science and Technology in 2008, and the Meritorious Award on Radio by the Association of Radio Industries and Businesses in 2010.

...