Hindawi
Mobile Information Systems
Volume 2019, Article ID 5173858, 12 pages
https://doi.org/10.1155/2019/5173858

Research Article

An Improved Opportunistic Routing Protocol for Intermittently Connected Delay Tolerant Wireless Sensor Networks

Dong Yeong Seo[1] and Yun Won Chung[2]

[1]*Department of Information and Telecommunication Engineering, Graduate School, Soongsil University, 369 Sangdo-ro Dongjak-gu, 06978 Seoul, Republic of Korea*
[2]*School of Electronic Engineering, Soongsil University, 369 Sangdo-ro Dongjak-gu, 06978 Seoul, Republic of Korea*

Correspondence should be addressed to Yun Won Chung; ywchung@ssu.ac.kr

Received 7 July 2018; Revised 7 April 2019; Accepted 2 May 2019; Published 26 June 2019

Academic Editor: Filippo Gandino

Copyright © 2019 Dong Yeong Seo and Yun Won Chung. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In intermittently connected delay tolerant wireless sensor networks, sensor data generated at sensor nodes should be delivered to a sink node using opportunistic contacts between intermittently connected nodes. Since there is no stable end-to-end routing path from a source node to a sink node in intermittently connected network, an opportunistic routing protocol to deliver sensor data efficiently is needed. In this paper, an improved opportunistic routing protocol is proposed, where both current delivery predictability and maximum delivery predictability are used together to decide whether sensor data should be forwarded or not to a contact node. The proposed protocol can reduce buffer overflow and thus increase the delivery ratio, which is one of the most important performance measures in delay tolerant wireless sensor networks. The performance of the proposed routing protocol is compared with that of PRoPHET protocol and FREAK protocol, by varying buffer sizes and the number of nodes, in terms of delivery ratio, overhead ratio, and delivery latency. Performance analysis results show that the proposed protocol has better delivery ratio, overhead ratio, and delivery latency than the PRoPHET protocol and FREAK protocol have, in most considered parameter values, with appropriate selection of message dissemination thresholds.

1. Introduction

In intermittently connected delay tolerant wireless sensor networks, conventional wireless sensor network protocols cannot be used directly for sensor data delivery from a sensor node to a sink node since there is no stable end-to-end routing path between them. Instead, sensor data generated at sensor nodes should be delivered to a sink node using opportunistic contacts between nodes. In intermittently connected delay tolerant wireless sensor networks, both the characteristics of delay tolerant networks and wireless sensor networks exist. Delay tolerant networks have intermittent connectivity, and thus, routing is carried out using opportunistic contact [1]. Wireless sensor networks have battery-operated nodes and thus deal with efficient battery power management by using low processing power [2]. However, considering these two characteristics of delay tolerant networks and wireless sensor networks together for efficient message delivery is not a simple problem, and extensive future work is still needed [1].

Application of intermittently connected delay tolerant wireless sensor networks includes message delivery in disaster environment, wildlife monitoring, etc. In disaster environment sensing, data cannot be delivered via communication infrastructure or stable routing path between nodes since infrastructure is destroyed and stable routing path does not exist. In wildlife monitoring, also, sensing data for animals should be delivered via opportunistic contacts between them. In this paper, we consider a disaster environment as an example scenario, where communication infrastructure is destroyed, so that sensor data should be delivered through intermediate nodes such as pedestrian, car, and tram via opportunistic contact. Then, we focus on the problem of how to efficiently deliver sensor data to a sink node in intermittently connected network environment. That is, we will consider the routing protocol itself by neglecting medium access control (MAC) issues, such as power-efficient duty cycled operation for limited battery energy. Figure 1 shows a considered intermittently connected delay tolerant wireless sensor networks environment, where sensor data are generated at mobile nodes and delivered to a sink node, where end-to-end routing path from a node to a sink node is not guaranteed.

A lot of work has been carried out on the routing protocols for intermittently connected delay tolerant networks [3–8], of which the Epidemic [6], Spray and Wait [7], and probabilistic routing protocol using history of encounters and transitivity (PRoPHET) [8] protocols are representative examples. In the Epidemic protocol, routing is performed based on simple flooding. That is, when two nodes contact each other, if the other node does not have a message already, the message is forwarded to it. In the Epidemic protocol, messages are rapidly delivered to the destination nodes. However, a high message overhead is the major disadvantage. To reduce the high message overhead in the Epidemic protocol, the number of total message copies in a network for any generated message is limited by a predefined number in the Spray and Wait protocol, which consists of a Spray phase and a Wait phase. In the Spray phase, message copies are distributed until some nodes have only one message copy. In the Wait phase, a message is not forwarded to another node, other than the destination node of the message. Although the Spray and Wait protocol can solve the message overhead problem of the Epidemic protocol, message copies are forwarded blindly, and an intelligent selection of forwarding nodes is still needed. In the PRoPHET protocol, which was standardized in Internet Research Task Force (IRTF) as RFC 6693 [8], each node calculates the delivery predictability of other nodes based on the contact history between them. Then, the messages are forwarded to a node with a higher delivery predictability to the destination node of the considered message. By doing this, forwarding nodes can be intelligently selected.

A lot of work has been done to improve the performance of the PRoPHET protocol [9–15]. In [9], a message is forwarded, only if the delivery predictability of the other node is higher than the predefined minimum threshold value and the sum of delivery predictabilities of the node which forwarded the message already is lower than a predefined maximum threshold. In [10], an ant colony optimization algorithm was used to improve the performance of the PRoPHET protocol. In [11], sociality of nodes is classified based on moving patterns of nodes and message is forwarded based on the sociality of nodes. In [12], delivery predictability is adjusted based on the number of forwarded message copies in order to solve the performance degradation problem that is caused by an excessive number of copies of any single message and thus manage the buffer of nodes efficiently. In [13], depending on the density of nodes, either the PRoPHET or the Epidemic protocol is selected. If the number of neighbor nodes is higher than a threshold, the PRoPHET protocol is used for routing. Otherwise, the Epidemic protocol is used. In [14], both contact duration and contact frequency between nodes are considered to calculate delivery predictability. In [15], more messages are distributed to nodes with higher delivery predictability in the Spray phase of the Spray and Wait protocol.

Studies on delay tolerant wireless sensor networks have been carried out recently [16–21]. In [16], sensor networks are deployed for wildlife habitat monitoring, and sensor data are gathered by data mule such as automobile and helicopter. Disruption support capabilities of delay tolerant networks are used for habitat monitoring. In [17], delay tolerant networks protocol is used to solve the data delivery challenge in the intermittently connected vehicular sensor networks, where position-based forwarding strategy is assumed for vehicular sensor networks. Vehicle driving direction is used to determine packet forwarding. In [18], the concept of credibility of a node is defined based on node selfishness, and Markov process is used to predict node movement. The proposed protocol reduces transmission delay and communication overhead, while ensuring message delivery rate. In [19], frequency routing, encounters, and keenness (FREAK) protocol is proposed, where average frequency of contact with base station is used to determine message forwarding, based on the assumption that future can be predicted based on past history, which is widely assumed in delay tolerant networks. In [20], delay tolerant wireless sensor networks are used to monitor river pollution, where several sensor clusters with floating transmitters are used to forward messages to a gateway located at the riverbank opportunistically. In [21], an energy-aware routing protocol in intermittently connected delay tolerant wireless sensor networks was proposed, where messages are forwarded based on the node's remaining battery, delivery predictability, and type of nodes.

In these studies [16–21], most proposed routing protocols for delay tolerant wireless sensor networks use currently developed basic delay tolerant networks protocols, and wireless sensor networks are considered as an application area for applying the delay tolerant networks protocol using opportunistic contacts. Therefore, there is still room for performance enhancement for delay tolerant wireless sensor networks. In this paper, we consider the PRoPHET protocol as the base protocol. The PRoPHET protocol with the GRTR strategy [8], which is a default routing strategy in PRoPHET protocol, is used for comparing protocol, where messages are forwarded to nodes with higher delivery predictability. GRTR is not a specific acronym but a just mnemonic. Also, FREAK is used for comparing the protocols. To improve the performance of the conventional PRoPHET protocol, we define the maximum delivery predictability for any node as the maximum value of delivery predictability for any node calculated until now, as was proposed in our preliminary work [22] originally. Our proposed work extends our preliminary work in [22] and differs from it significantly in the following ways:

(i) Sensor environment is newly considered by placing a sink node in a network and considering only communication from mobile nodes to a sink node.

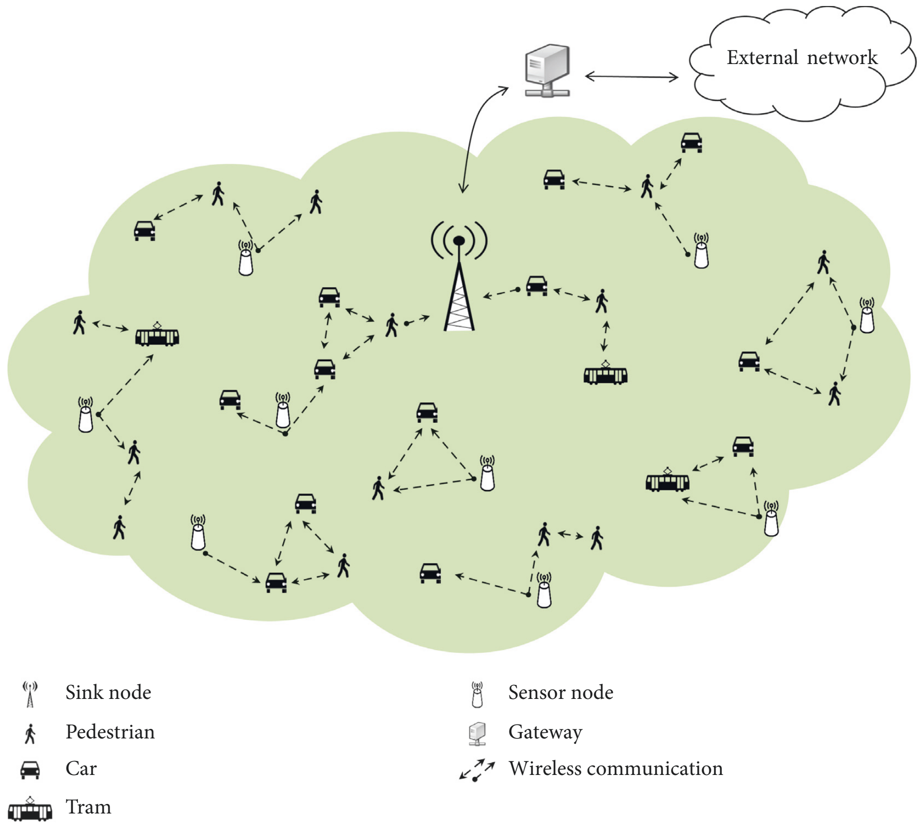


Figure 1: A considered network environment.

(ii) The criteria of forwarding are generalized by considering four cases based on the relationship between current and maximum delivery predictabilities of the sending node and predictabilities of the receiving node. We note that forwarding is possible in our preliminary work only if both the current and the maximum delivery predictabilities of the receiving nodes are larger than those of the sending node. However, in the proposed work, message forwarding is possible in three different cases.

(iii) The performance of the proposed protocol is analyzed in detail based on the simulation using ONE simulator.

The remainder of this paper is organized as follows: In Section 2, a detailed algorithm of the proposed protocol is proposed and the proposed protocol is illustrated using examples. In Section 3, the performance of the proposed protocol is analyzed extensively using simulation. Finally, Section 4 summarizes this work and presents future work.

2. Proposed Protocol

In this section, a detailed algorithm of the proposed protocol is described. Then, an example of message delivery in the proposed protocol is described. In this paper, the PRoPHET protocol is considered as the basic routing protocol. In PRoPHET, delivery predictability is defined between any two nodes, and it increases whenever the two nodes contact each other. It decreases as time elapses after the last contact. The detailed formula for the calculation of delivery predictability is defined in IRTF RFC 6693 [8] and are summarized here. The delivery predictability of node A to node B, which is denoted as $P(\mathrm{A}, \mathrm{B})$, estimates a likelihood of delivering a message to node B, and it has a value between 0 and 1. If node A and node B contact each other, the delivery predictability between them increases, as defined in equation (1) [8]:

$$P(\mathrm{A}, \mathrm{B}) = P(\mathrm{A}, \mathrm{B})_{\mathrm{old}} + \left(1 - \Delta - P(\mathrm{A}, \mathrm{B})_{\mathrm{old}}\right) \times P_{\mathrm{encounter}}, \quad (1)$$

where $P_{\mathrm{encounter}}$ and Δ are a scaling factor and an upper bound of $P(\mathrm{A}, \mathrm{B})$, respectively. The delivery predictability between nodes A and B decreases exponentially after the contact as time goes on, as defined in the following equation [8]:

$$P(\mathrm{A}, \mathrm{B}) = P(\mathrm{A}, \mathrm{B})_{\mathrm{old}} \times \gamma^{K}, \quad (2)$$

where γ and K are aging constant and the number of passed time units after last contact, respectively. Finally, there is a transitive property. That is, if node A and node B contact frequently and if node B and node C contact frequently, it is highly probable that node A and node C contact frequently, as defined in the following equation [8]:

$$P(\mathrm{A}, \mathrm{C}) = \mathrm{MAX}\left(P(\mathrm{A}, \mathrm{C})_{\mathrm{old}}, P(\mathrm{A}, \mathrm{B}) * P(\mathrm{B}, \mathrm{C})_{\mathrm{recv}} \times \beta\right), \quad (3)$$

where β is a scaling constant.

In this paper, we use the history of delivery predictability for any destination node as well as the current delivery predictability. We define the maximum delivery predictability (mP) for any destination node, which is the highest value of delivery predictabilities recorded for the destination node, as was proposed in our preliminary work [22] originally as follows:

$$mP(\mathrm{A},\mathrm{B}) = \mathrm{MAX}[P(\mathrm{A},\mathrm{B})t], \quad 0 < t < \text{current time}. \tag{4}$$

In the proposed protocol, delivery predictabilities for all other nodes are calculated continuously, as in the PRoPHET protocol, and the maximum value of the delivery predictability for each node is updated. This maximum value is defined as maximum delivery predictability, in contrast to the current delivery predictability defined in PRoPHET [8]. By using both the current value and the maximum value of delivery predictability together, we propose an efficient routing protocol in intermittently connected delay tolerant wireless sensor networks. The rationale behind the proposed protocol is that history information of delivery predictability can help to find a better node to forward a message.

Figure 2 shows the flow chart of the proposed protocol. When node A and node B contact each other, they exchange summary vectors, which include forwarding the information of the messages that they have and the delivery predictabilities for the destination nodes of the messages. Then, based on the summary vector, node A lists the messages to forward, which do not exist in node B. Then, node A chooses a message in the list and compares the delivery predictability for the destination node of the considered message of node A with that of node B. If the current delivery predictability of node B is higher than that of node A, then maximum delivery predictability is additionally compared. If the maximum delivery predictability for the destination node of node B is higher than that of node A, the message is forwarded to node B. Otherwise, the difference between current delivery predictabilities of node B and node A is compared with the sum of the difference between the maximum delivery predictabilities of nodes A and B and a predefined threshold value T_{m2} as follows:

$$\Delta P_{\mathrm{BA}} > \Delta mP_{\mathrm{AB}} + T_{\mathrm{m2}}, \tag{5}$$

where the definition of notations is given in Figure 2. If the difference between the current delivery predictabilities of node B and node A is higher than the sum of the difference between maximum delivery predictabilities of nodes A and B and a predefined threshold value, the message is forwarded to node B; otherwise, it is not forwarded. If the current delivery predictability of node B is not higher than that of node A, then maximum delivery predictability is also compared. If the maximum delivery predictability for the destination node of node B is not higher than that of node A, the message is not forwarded to node B. Otherwise, the difference between maximum delivery predictabilities of node B and node A is compared with the sum of the difference between the current delivery predictabilities of nodes A and B and a predefined threshold value T_{m1} as follows:

$$\Delta mP_{\mathrm{BA}} > \Delta P_{\mathrm{AB}} + T_{\mathrm{m1}}, \tag{6}$$

where the definition of notations is given in Figure 2. If the difference between the maximum delivery predictabilities of node B and node A is higher than the sum of the difference between the current delivery predictabilities of nodes A and B and a predefined value, the message is forwarded to node B; otherwise, the message is not forwarded. This procedure is repeated for all the listed messages.

The novelty of the proposed protocol is to use both current delivery predictability and maximum delivery predictability together for efficient message forwarding, while only current delivery predictability is used in original PRoPHET protocol. Although current delivery predictability is a good indicator to decide a better forwarder, as shown in the performance analysis of extended PRoPHET protocols [9–15], we additionally consider maximum delivery predictability to decide a forwarder more selectively. This is because message forwarding based on only current delivery predictability may result in more message forwarding than needed, and this may decrease the delivery ratio due to buffer overflow of mobile nodes, which do not have sufficient buffer memory. In order to reduce buffer overflow, we restrict message forwarding efficiently by considering both current delivery predictability and maximum delivery predictability together, as shown in Figure 2 already. Also, the threshold values of T_{m1} and T_{m2} are defined to control the dissemination of messages appropriately. By doing these, the number of forwarded messages is controlled more selectively and better performance is achieved.

Figures 3 and 4 show examples of message forwarding in PRoPHET with GRTR strategy and the proposed protocol, respectively, in the same scenario. We assume that node A has messages m1 to m7 with corresponding delivery predictability values to the destination nodes for the messages. Also, node B has delivery predictability values for the destination nodes for the messages stored in node A. In Figure 3, node A forwards messages 1, 3, 4, and 6 to node B, according to GRTR strategy of PRoPHET protocol [8], since the delivery predictabilities of node B for those messages are higher than those of node A.

In Figure 4, however, different situations happen depending on the predefined message dissemination threshold values of T_{m1} and T_{m2}. We note that these threshold values are assumed to be between −1 and 1 since they are related with predictability comparison. In Figure 4, if both of them are equal to 1, only messages 4 and 6 are forwarded to node B, based on the relationship of current and maximum delivery predictabilities. If the values of T_{m1} and T_{m2} are equal to −1, messages 1, 2, 3, 4, 5, and 6 are forwarded to node B. From the results in Figure 4, it is shown that message forwarding can be managed dynamically by controlling the values of T_{m1} and T_{m2}. If these values are small, more message disseminations are achieved, as can be inferred from Figure 2. Otherwise, less message disseminations are allowed. In Section 3, we analyze the impact of those values on the performance of the proposed protocol in detail.

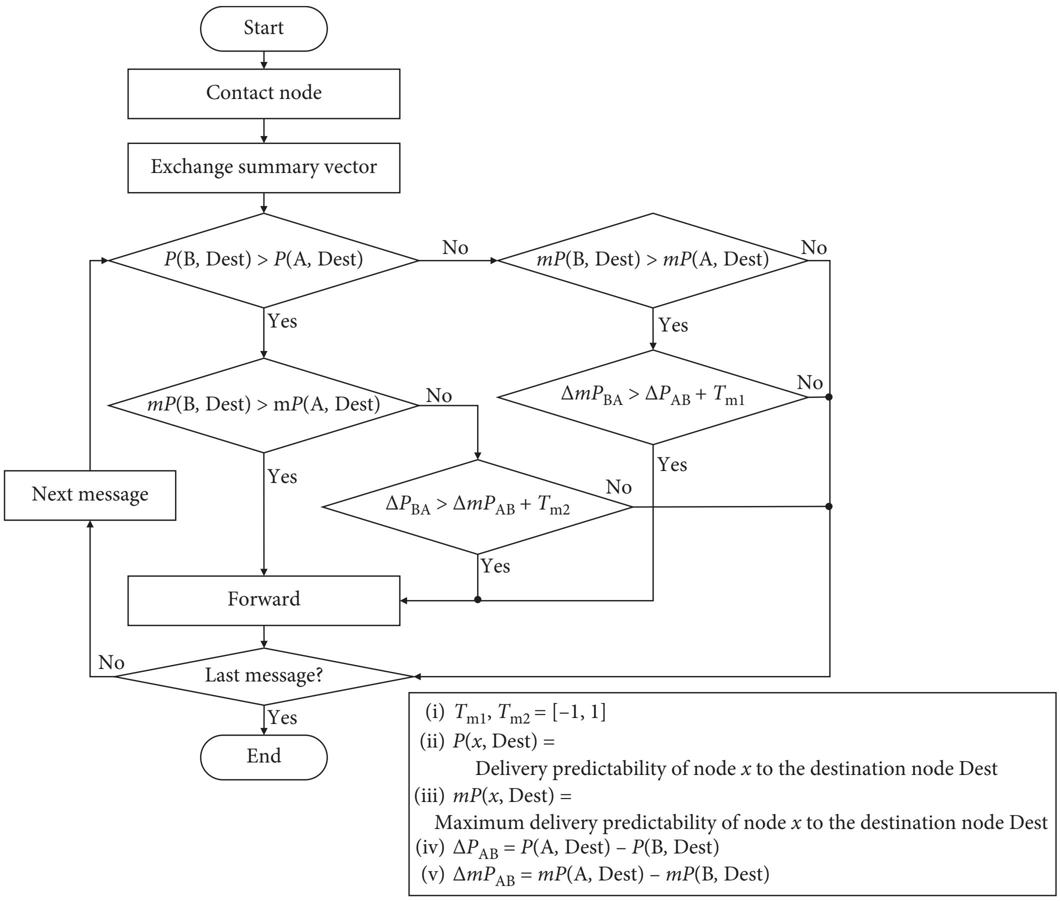
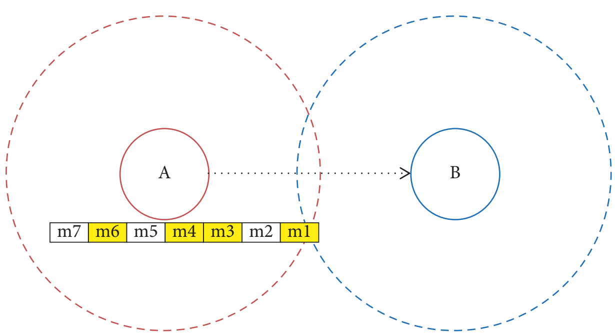


Figure 2: Flow chart of the proposed protocol.



<Message list of node A>

Message ID	P(A, Dest)
m1	0.75
m2	0.65
m3	0.7
m4	0.55
m5	0.9
m6	0.85
m7	0.8

<Delivery probability info of node B>

Node ID	P(B, Dest)
C	0.8
D	0.5
E	0.75
F	0.9
G	0.8
H	0.95
I	0.7

Figure 3: Message delivery in the PRoPHET protocol.

3. Performance Analysis

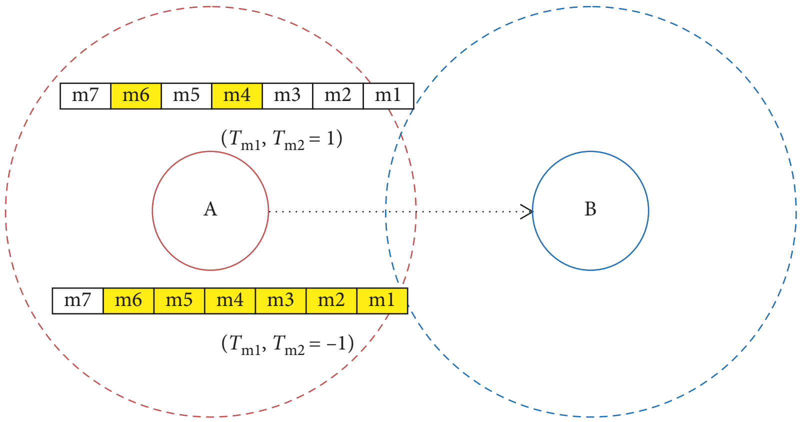
In this section, the performance of the proposed protocol is compared with that of the PRoPHET protocol and FREAK protocol [19] using simulation in the considered intermittently connected delay tolerant wireless sensor networks, with an objective to show the effectiveness of the proposed protocol. Simulation was carried out by using opportunistic network simulator (ONE) developed by Helsinki University [23, 24], for different buffer sizes, in terms of delivery ratio, overhead ratio, and delivery latency, which are defined as follows:

$$\text{delivery ratio} = \frac{N_{\mathrm{dm}}}{N_{\mathrm{cm}}}, \tag{7}$$

$$\text{overhead ratio} = \frac{N_{\mathrm{rm}} - N_{\mathrm{dm}}}{N_{\mathrm{dm}}}, \tag{8}$$

$$\text{delivery latency} = \frac{\mathrm{SD}_{\mathrm{dm}}}{N_{\mathrm{dm}}}, \tag{9}$$

where N_{dm}, N_{cm}, N_{rm}, and $\mathrm{SD}_{\mathrm{dm}}$ represent the number of successfully delivered messages, the number of created



<Message list of node A>

Message ID	P(A, Dest)	mP(A, Dest)
m1	0.75	0.9
m2	0.65	0.75
m3	0.7	0.9
m4	0.55	0.75
m5	0.9	0.9
m6	0.85	0.9
m7	0.8	0.85

<Delivery probability info of node B>

Node ID	P(B, Dest)	mP(B, Dest)
C	0.8	0.85
D	0.5	0.85
E	0.75	0.8
F	0.9	0.9
G	0.8	0.95
H	0.95	0.95
I	0.7	0.75

<T_{m1}, T_{m2} = 1>

Message ID	P(B, Dest) > P(A, Dest)	mP(B, Dest) > mP(A, Dest)	$\Delta P_{BA} > \Delta mP_{AB} + T_{m2}$	$\Delta mP_{BA} > P_{AB} + T_{m1}$
m1	O	X	X	-
m2	X	O	-	X
m3	O	X	X	-
m4	O	O	-	-
m5	X	O	-	X
m6	O	O	-	-
m7	X	X	-	-

<T_{m1}, T_{m2} = −1>

Message ID	P(B, Dest) > P(A, Dest)	mP(B, Dest) > mP(A, Dest)	$\Delta P_{BA} > \Delta mP_{AB} + T_{m2}$	$\Delta mP_{BA} > P_{AB} + T_{m1}$
m1	O	X	O	-
m2	X	O	-	O
m3	O	X	O	-
m4	O	O	-	-
m5	X	O	-	O
m6	O	O	-	-
m7	X	X	-	-

Figure 4: Message delivery in the proposed protocol.

messages, the number of relayed messages, and the sum of the delays of the all delivered messages. Total 100 nodes are distributed in a considered area, and one node acts as a sink node and other nodes can generate sensor data and forward data using opportunistic contact. In order to analyze the effect of traffic load, three different message generation intervals are considered. Also, different values of buffer size of mobile sensor nodes are considered. Detailed values of the simulation parameters assumed in this paper are listed in Table 1.

In the numerical examples, we carried out simulations for different values of T_{m1} and T_{m2}, i.e., −1, 0, and 1, in order to show the effect of different values of T_{m1} and T_{m2} on the delivery ratio, overhead ratio, and delivery latency. These threshold values are defined to control the message dissemination speed and can have the value between −1 and 1, where the values of −1 and 1 for the values of T_{m1} and T_{m2} are two extremes and 0 is the medium value of the two extremes. The value of 1 for the threshold values restricts message dissemination and the value of −1 for the threshold values promotes message dissemination. The value of 0 for the threshold values is neutral to message dissemination. Since these three values of −1, 0, and 1 reflect important characteristics of threshold values for message dissemination speed, we only consider these three values for numerical examples. Also, we used three message intervals, i.e., 12 s, 60 s, and 300 s, in order to show the effect of traffic load on the performance of the proposed protocol.

Figure 5 shows the delivery ratio for different buffer sizes when the traffic load is low (message interval = 300 s). The proposed protocol with $T_{m1} = T_{m2} = -1$ has the highest delivery ratio for most buffer sizes, because if traffic load is low, the buffer of nodes will accommodate the messages sufficiently, and thus more dissemination of messages ($T_{m1} = T_{m2} = -1$) will be more favorable. On the other hand, less dissemination of messages ($T_{m1} = T_{m2} = 1$) will be less favorable. For very small buffer size (buffer size = 10 Mbytes), since the buffer size is not sufficient, less dissemination of messages ($T_{m1} = T_{m2} = -1$) performs well. The FREAK protocol has the smallest delivery ratio for most of the considered buffer sizes. This is because since message forwarding is not possible in the early phase of simulation, messages are delivered by direct delivery, since both contact nodes have not contacted sink node yet.

Figure 6 shows the overhead ratio for different buffer sizes when the traffic load is low (message interval = 300 s). The proposed protocol with $T_{m1} = T_{m2} = 1$ and $T_{m1} = T_{m2} = 0$ has less overhead ratio than PRoPHET protocol, and the proposed protocol with $T_{m1} = T_{m2} = -1$ has the highest overhead ratio. Since the proposed protocol with $T_{m1} = T_{m2} = -1$ forwards more messages, the message overhead is more. The FREAK protocol has the smallest overhead.

Figure 7 shows delivery latency for different buffer sizes when the traffic load is low (message interval = 300 s). The FREAK protocol has the largest delivery latency, and the proposed protocols with both $T_{m1} = T_{m2} = -1$ and $T_{m1} = T_{m2} = 0$ have a shorter delivery latencies than the PRoPHET protocol and FREAK protocol. This is because restriction of message dissemination in the considered environment increases message delivery latency.

Figure 8 shows the delivery ratio for different buffer sizes when the traffic load is medium (message interval = 60 s). The proposed protocol with $T_{m1} = T_{m2} = 0$ has the highest delivery ratio for most buffer sizes. This is because if traffic load is medium, an appropriate dissemination of messages is better than too many ($T_{m1} = T_{m2} = -1$) or too few

Table 1: Parameter values.

Parameter	Value
Area size (m^2)	4,500 × 3,400
Router	PRoPHET router
$P_{encounter}$	0.75
γ	0.98
β	0.25
Time unit for K (s)	30
Movement model	Pedestrians, car: shortest path map based movement tram: map route movement
Speed (m/s)	Pedestrian: U[0.5, 1.5], car: U[2.7, 13.9], tram: U[7,10]
Number of nodes	Total = 100 (default value, pedestrians = 62, car = 31, tram = 6, sink node = 1)
Simulation time (s)	43,200
Transmission range (m)	10
Packet transmission speed (kbps)	250
Buffer size (Mbytes)	10, 20, 30 (default value), 40, 50, 60, 70, 80, 90, 100,
Message interval (s)	12 (high traffic load), 60 (medium traffic load), 300 (low traffic load)
Message size (bytes)	U[500 k~1 M]

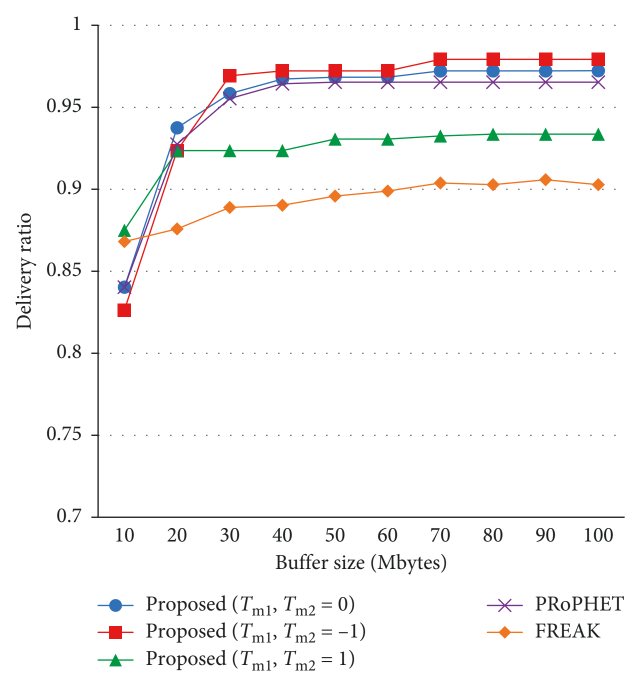


Figure 5: Delivery ratio for different buffer sizes (traffic interval = 300 s).

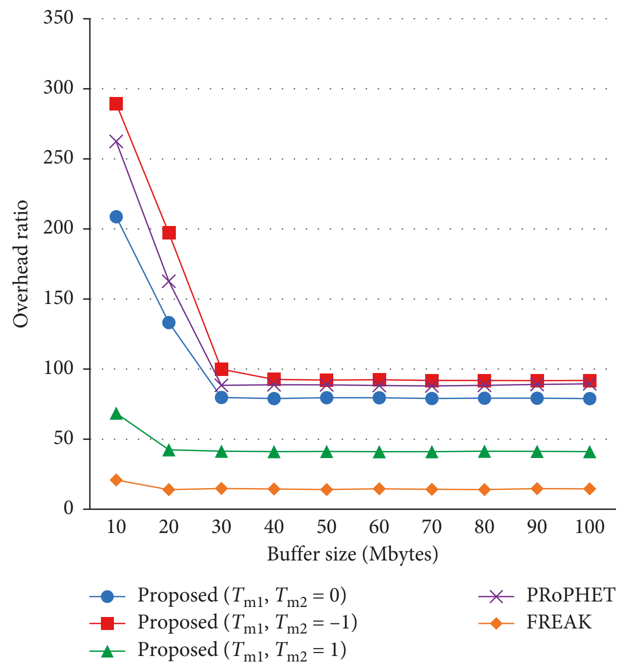


Figure 6: Overhead ratio for different buffer sizes (traffic interval = 300 s).

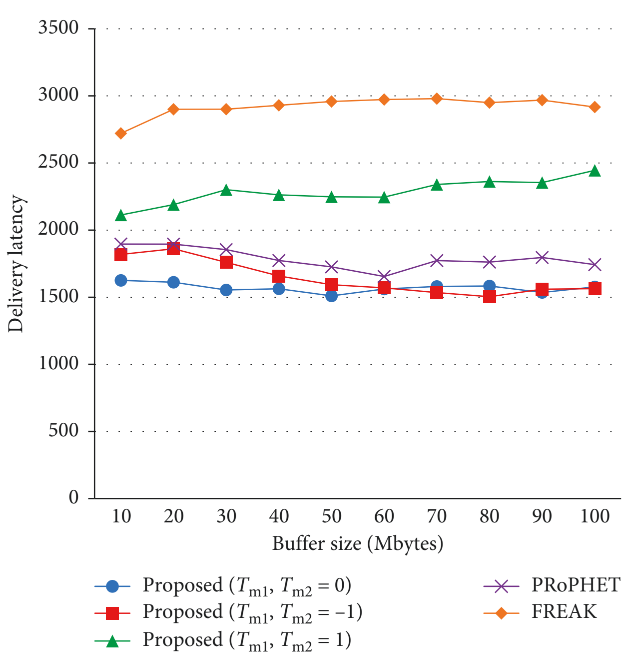


Figure 7: Delivery latency for different buffer sizes (traffic interval = 300 s).

($T_{m1} = T_{m2} = 1$) message dissemination. For small buffer sizes (buffer size = 10 Mbytes); however, the proposed protocol with $T_{m1} = T_{m2} = 1$ has the highest delivery ratio, because if the buffer size is insufficient, less dissemination of messages results in less message drop caused by buffer overflow and thus achieves a better delivery ratio.

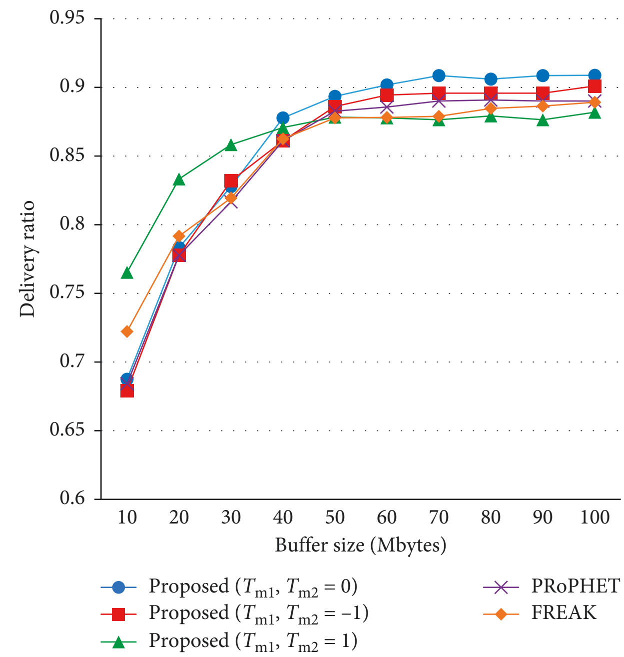


Figure 8: Delivery ratio for different buffer sizes (traffic interval = 60 s).

Figure 9 shows the overhead ratio for different buffer sizes when the traffic load is medium (message interval = 60 s). The proposed protocol with $T_{m1} = T_{m2} = 1$ has the lowest overhead ratio in small buffer sizes and has similar overhead ratio with the FREAK protocol for large buffer sizes. The proposed protocol with $T_{m1} = T_{m2} = -1$ and PRoPHET have the highest overhead ratio. This is because the proposed protocol with $T_{m1} = T_{m2} = 1$ forwards messages selectively, so the message overhead is smaller. On the other hand, since the proposed protocol with $T_{m1} = T_{m2} = -1$ forwards more messages, the message overhead is larger.

Figure 10 shows the delivery latency for different buffer sizes when the traffic load is medium (message interval = 60 s). The proposed protocol with $T_{m1} = T_{m2} = 1$ has the largest delivery latency since less dissemination results in longer delivery latency. The proposed protocol with $T_{m1} = T_{m2} = -1$ has a shorter delivery latency than PRoPHET protocol and FREAK protocol have.

Figure 11 shows the delivery ratio for different buffer sizes when the traffic load is high (message interval = 12 s). The proposed protocol with $T_{m1} = T_{m2} = 1$ has the highest delivery ratio for most buffer sizes. This is because if traffic load is high, more selective dissemination of message is better, which results in less message drop due to buffer overflow. For very large buffer sizes (buffer size = 100 Mbytes), however, the proposed protocol with $T_{m1} = T_{m2} = 1$ has a little bit smaller delivery ratio than that with both $T_{m1} = T_{m2} = -1$ and $T_{m1} = T_{m2} = 0$, because if the buffer size is sufficiently large, the effect of less message dissemination is more dominant than that of a large message drop due to buffer overflow. In most of the considered buffer sizes, PRoPHET protocols and FREAK protocols do not have better delivery ratio than the proposed protocol.

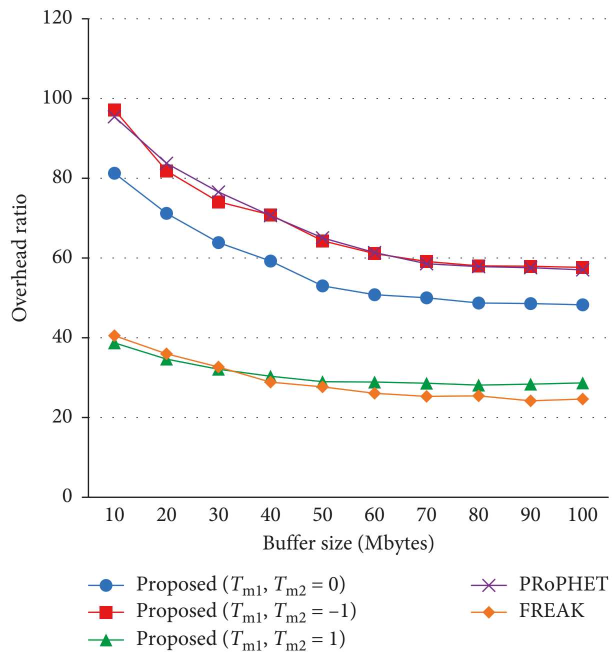


Figure 9: Overhead ratio for different buffer sizes (traffic interval = 60 s).

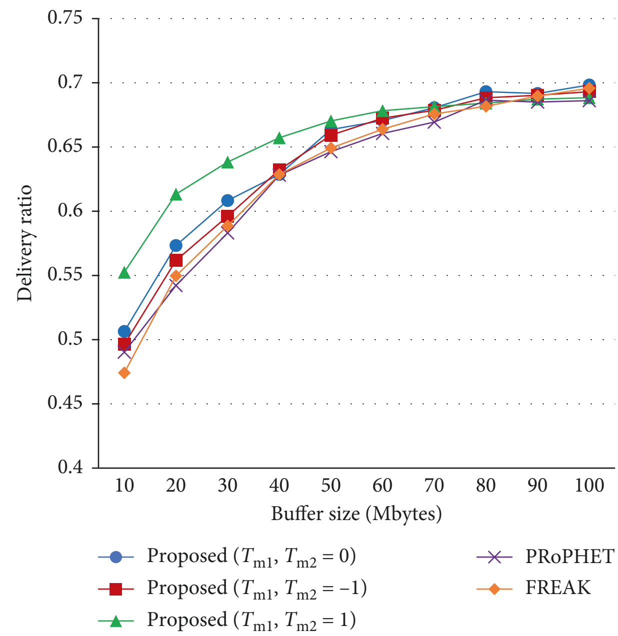


Figure 11: Delivery ratio for different buffer sizes (traffic interval = 12 s).

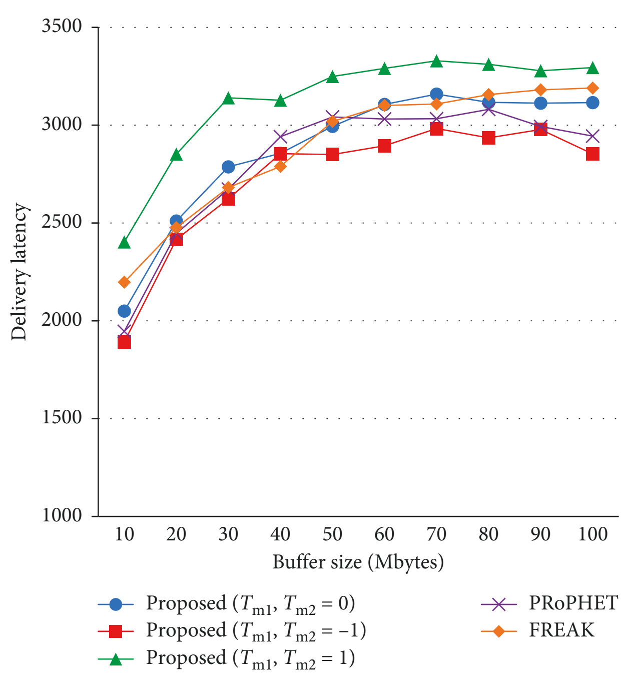


Figure 10: Delivery latency for different buffer sizes (traffic interval = 60 s).

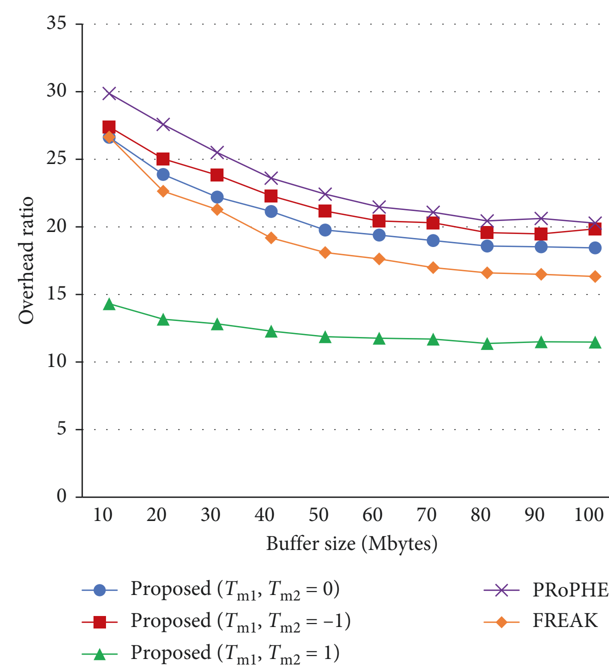


Figure 12: Overhead ratio for different buffer sizes (traffic interval = 12 s).

Figure 12 shows the overhead ratio for different buffer sizes when the traffic load is high (message interval = 12 s). The proposed protocol with $T_{m1} = T_{m2} = 1$ has the lowest overhead ratio, and PRoPHET has the highest overhead ratio, because the proposed protocol with $T_{m1} = T_{m2} = 1$ forwards messages selectively, so the message overhead is less. The FREAK protocol has the second smallest overhead ratio.

Figure 13 shows the delivery latency for different buffer sizes when the traffic load is high (message interval = 12 s). The proposed protocol with $T_{m1} = T_{m2} = 1$ has the largest delivery latency since less dissemination results in longer delivery latency. The proposed protocols with both $T_{m1} = T_{m2} = -1$ and $T_{m1} = T_{m2} = 0$ have shorter delivery latencies than PRoPHET for most buffer sizes.

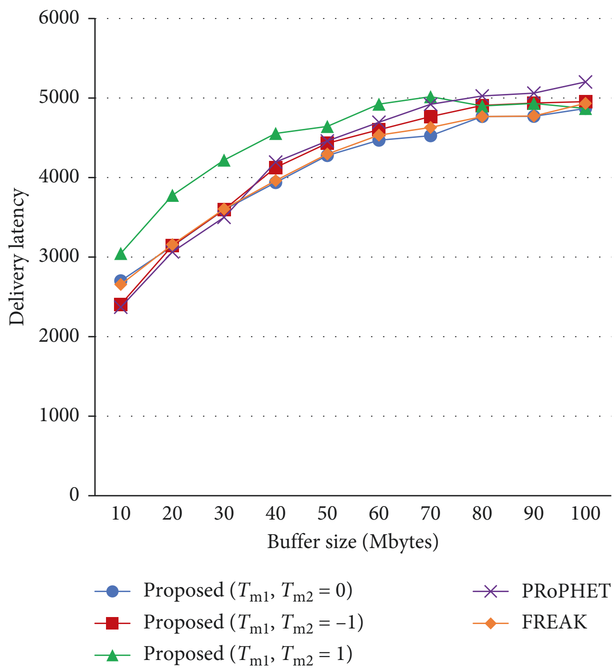


FIGURE 13: Delivery latency for different buffer sizes (traffic interval = 12 s).

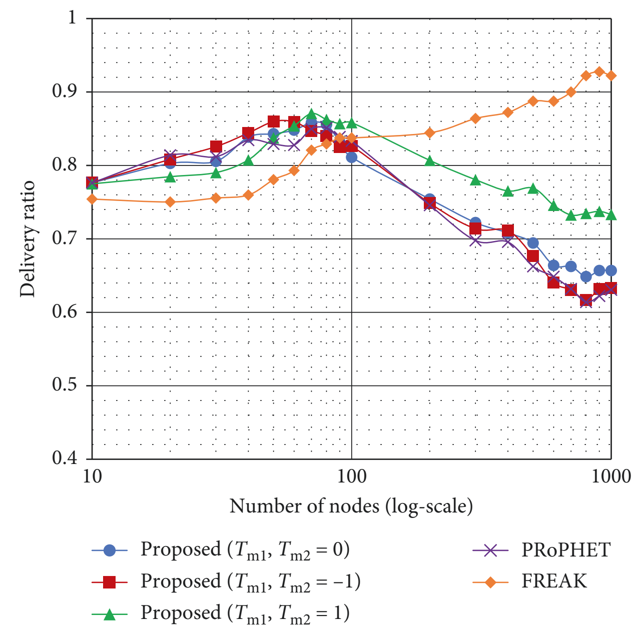


FIGURE 14: Delivery ratio for different number of nodes (traffic interval = 60 s).

Figure 14 shows the delivery ratio for different number of nodes when the traffic load is medium (message interval = 60 s). The proposed protocol with $T_{m1} = T_{m2} = -1$ has the largest delivery ratio when the number of nodes is smaller than or equal to 100. On the other hand, the FREAK protocol has the largest delivery ratio when the number of nodes is larger than 100. This is because if the number of nodes is not large, the proposed protocol with $T_{m1} = T_{m2} = -1$ promotes message dissemination and more and more messages are delivered successfully. On the other hand, if the number of nodes is large, the proposed protocol and PRoPHET protocol result in buffer overflow due to more message dissemination, and thus FREAK protocol has better delivery ratio. We note that a scenario of less number of nodes is more feasible in real delay tolerant wireless sensor networks due to the inherent characteristics of delay tolerant networks, and thus, the proposed protocol is more suited for delay tolerant wireless sensor networks.

Figure 15 shows the overhead ratio for different number of nodes when the traffic load is medium (message interval = 60 s). When the number of node is not large, the overhead ratio of all the protocols is small and the overhead ratio increases significantly as the number of nodes increases due to more message dissemination. The proposed protocol with $T_{m1} = T_{m2} = 1$ has the second smallest overhead ratio, when the number of nodes is smaller than or equal to 100. On the other hand, the FREAK protocol has the smallest overhead ratio, when the number of nodes is larger than 100.

Figure 16 shows the delivery latency for different number of nodes when the traffic load is medium (message interval = 60 s). Delivery latency decreases as the number of nodes increases. Also, the proposed protocol with $T_{m1} = T_{m2} = -1$ has the smallest delivery latency in most considered number

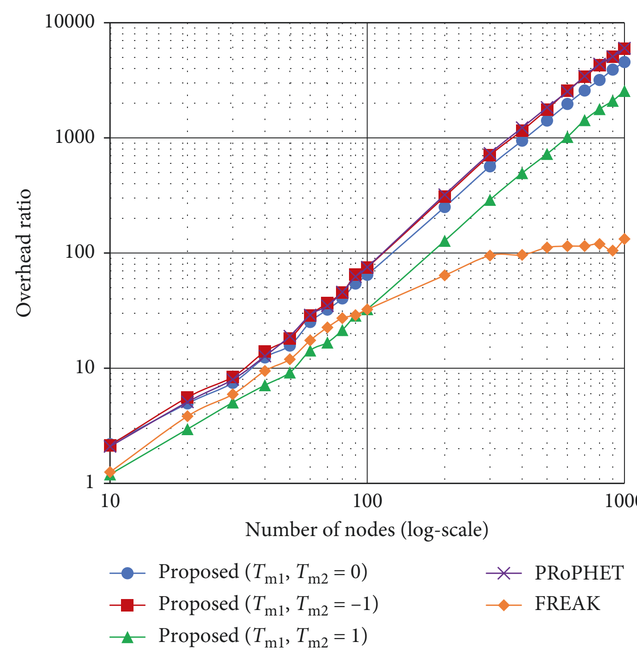


FIGURE 15: Overhead ratio for different number of nodes (traffic interval = 60 s).

of nodes. For large number of nodes, the FREAK protocol has the largest delivery latency.

4. Conclusion and Future Work

In this paper, an efficient routing protocol was proposed, where both current delivery predictability and maximum delivery predictability are used to decide whether sensor data should be forwarded or not. Then, the performance of the

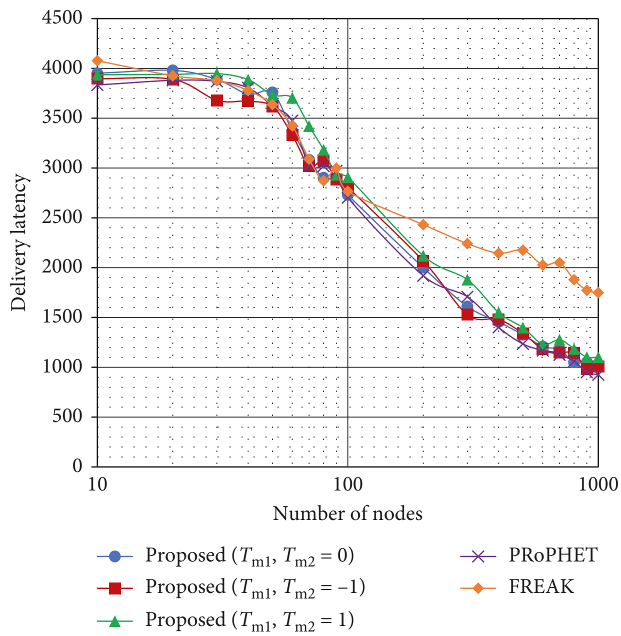


Figure 16: Delivery latency for different number of nodes (traffic interval = 60 s).

proposed protocol was compared with that of PRoPHET protocol and FREAK protocol, for different buffer sizes and different number of nodes, in terms of delivery ratio, overhead ratio, and delivery latency. Performance analysis results show that the proposed protocols have a better delivery ratio, overhead ratio, and delivery latency than the PRoPHET protocol and FREAK protocol have, for varying buffer sizes, given appropriate selection of message dissemination thresholds, where an exception is that the FREAK protocol has the smallest overhead ratio when the traffic load is the smallest. For varying the number of nodes, the proposed protocols have a better delivery ratio than the PRoPHET protocol and FREAK protocol have, when the number of nodes is not large. Also, the proposed protocol has the smallest delivery latency.

In our future work, MAC issues such as duty cycled operation of nodes will be considered additionally in order to extend the lifetime of the node efficiently, which is one of the important requirements of wireless sensor networks. Then, an efficient strategy of selecting message dissemination thresholds and duty ratio will be proposed and analyzed based on cross-layer work between MAC and routing protocols.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (NRF-2016R1D1A1B03930299).

References

[1] Y. Li and R. Bartos, "A survey of protocols for intermittently connected delay-tolerant wireless sensor networks," *Journal of Network and Computer Applications*, vol. 41, pp. 411–423, 2014.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks*, vol. 38, no. 4, pp. 393–422, 2002.

[3] S. Burleigh, A. Hooke, L. Torgerson et al., "Delay-tolerant networking: an approach to interplanetary internet," *IEEE Communications Magazine*, vol. 41, no. 6, pp. 128–136, 2003.

[4] Delay Tolerant Networking Research Group, http://www.dtnrg.org.

[5] Z. Zhang, "Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges," *IEEE Communications Survey and Tutorial*, vol. 8, no. 1, pp. 24–37, 2006.

[6] X. Zhang, G. Neglia, J. Kurose, and D. Towsley, "Performance modeling of epidemic routing," *Elsevier Computer Networks*, vol. 51, no. 10, pp. 2867–2891, 2007.

[7] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *Proceedings of ACM SIGCOMM Workshop on Delay-Tolerant Networking*, pp. 252–259, Philadelphia, PA, USA, August 2005.

[8] A. Lindgren, A. Doria, E. Davies, and S. Grasic, "Probabilistic routing protocol for intermittently connected networks," in *IRTF RFC 6693*, 2012.

[9] A. El Ouadrhiri, M. El Kamili, M. Raiss El Fenni, and L. Omari, "Learning controlled forwarding strategy improving probabilistic routing in DTNs," in *Proceedings of IEEE Wireless Communications and Networking Conference(WCNC)*, Istanbul, Turkey, April 2014.

[10] M. Ababou, R. Elkouch, M. Bellafkih, and N. Ababou, "AntProPHET: a new routing protocol for delay tolerant networks," in *Proceedings of 14th Mediterranean Microwave Symposium (MMS)*, Marrakech, Morocco, December 2014.

[11] K. H. LEE, T. M. Park, and C. K. Kim, "Social-community based DTN routing," *Journal of Korean Institute of Information Scientists and Engineers*, vol. 38, no. 5, pp. 366–373, 2011.

[12] Y. Liu, J. Wang, S. Zhang, and H. Zhou, "A buffer management scheme based on message transmission status in delay tolerant network," in *Proceedings of Global Telecommunications Conference (GLOBECOM 2011)*, Kathmandu, Nepal, December 2011.

[13] M. J. Kim and Y. W. Chung, "An improved message delivery scheme based on node density in delay tolerant network," *Journal of Korean Institute of Information Technology*, vol. 12, no. 10, pp. 69–74, 2014.

[14] H. J. Lee, J. C. Nam, W. K. Seo, J. I. Choi, and Y. J. Cho, "An efficient DTN routing protocol with considering contact duration," *Journal of Korean Institute of Communication Sciences*, pp. 921-922, 2015.

[15] J. Xue, X. Fan, Y. Cao, J. Fang, and J. Li, "Spray and wait routing based on average delivery probability in delay tolerant

network," in *Proceedings of International Conference on Network Security, Wireless Communications and Trusted Computing (NSWCTC 2009)*, Wuhan, China, April 2009.

[16] A. Tovar, T. Friesen, K. Ferens, and B. McLeod, "A DTN wireless sensor network for wildlife habitat monitoring," in *Proceedings of CCECE 2010*, Calgary, Canada, May 2010.

[17] F. Li, L. Zhao, X. Fan, and Y. Wang, "Hybrid position-based and DTN forwarding for vehicular sensor networks," *International Journal of Distributed Sensor Networks*, vol. 8, no. 4, article 186146, pp. 1–10, 2012.

[18] S. Zou, W. Wang, and W. Wangm, "A routing algorithm on delay-tolerant of wireless sensor network based on the node selfishness," *EURASIP Journal on Wireless Communications and Networking*, vol. 2013, no. 1, p. 212, 2013.

[19] P. Raveneau, E. Chaput, R. Dhaou, and A. Beylot, "FREAK DTN: frequency routing, encounters and keenness for DTN," in *Proceedings of 2014 IFIP Wireless Days (WD)*, Rio de Janeiro, Brazil, November 2014.

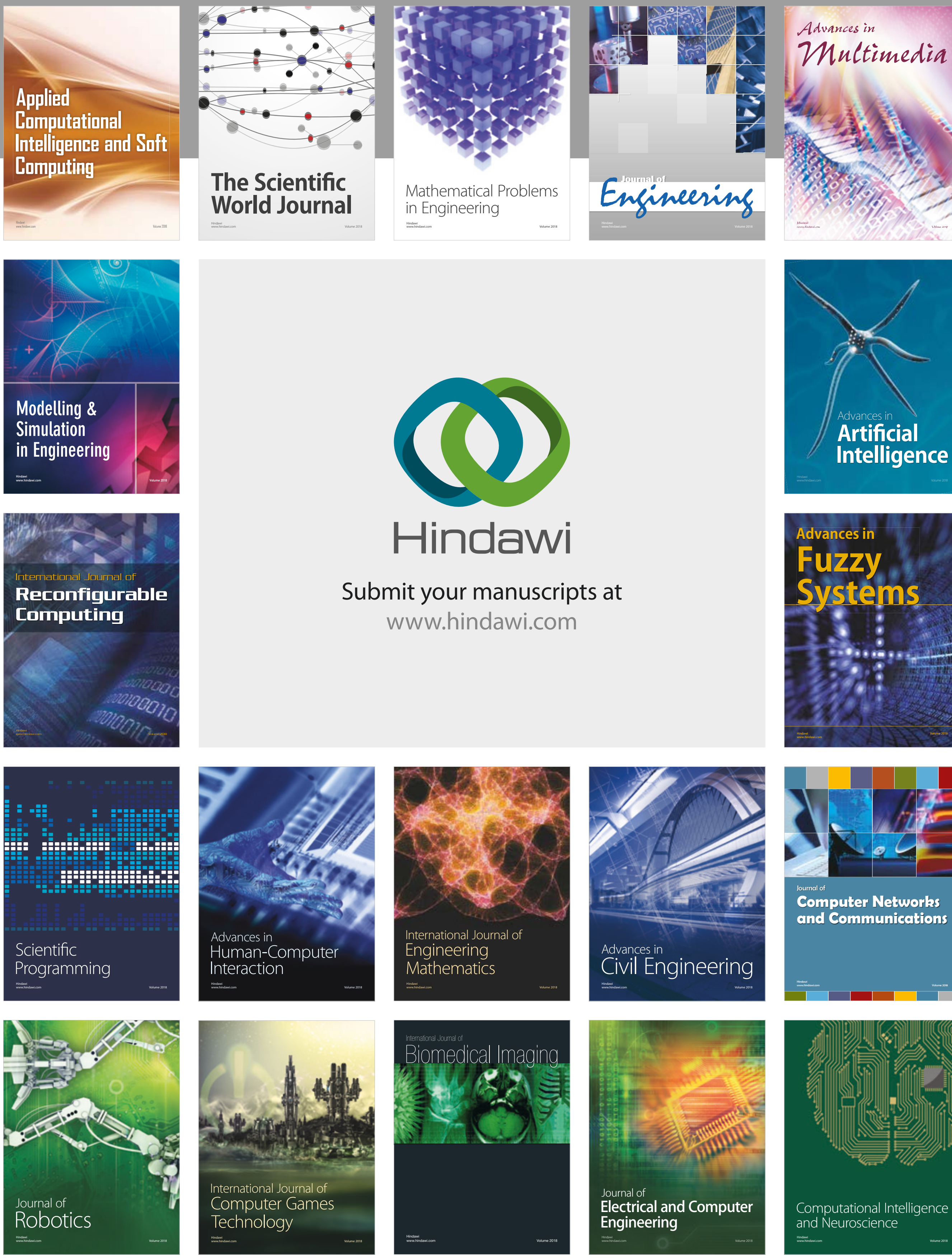
[20] C. Velásquez-Villada and Y. Donoso, "Delay/disruption tolerant network-based message forwarding for a river pollution monitoring wireless sensor network application," *Sensors*, vol. 16, no. 436, pp. 1–25, 2016.

[21] M.W. Kang and W.Y. Chung, "A novel energy-aware routing protocol in intermittently connected delay-tolerant wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 13, no. 7, pp. 1–10, 2017.

[22] D. Y. Seo and Y. W. Chung, "A novel routing protocol using current and maximum delivery predictabilities in delay tolerant networks," *Proceedings of Applied Science and Engineering for Better Human Life*, vol. 12, pp. 29–33, 2017.

[23] The Opportunistic Network Environment Simulator, http://www.netlab.tkk.fi/tutkimus/dtn/theone/.

[24] A. Keranen, J. Ott, and T. Karkkainen, "The ONE simulator for DTN protocol evaluation," in *Proceedings of the Second International Conference on Simulation Tools and Techniques*, Rome, Italy, March 2009.


Applied Computational Intelligence and Soft Computing
The Scientific World Journal
Mathematical Problems in Engineering
Journal of Engineering
Advances in Multimedia
Modelling & Simulation in Engineering
Hindawi
Submit your manuscripts at
www.hindawi.com
Advances in Artificial Intelligence
International Journal of Reconfigurable Computing
Advances in Fuzzy Systems
Scientific Programming
Advances in Human-Computer Interaction
International Journal of Engineering Mathematics
Advances in Civil Engineering
Journal of Computer Networks and Communications
Journal of Robotics
International Journal of Computer Games Technology
International Journal of Biomedical Imaging
Journal of Electrical and Computer Engineering
Computational Intelligence and Neuroscience