FAST-DTN: Farther-Aim-Shorter-Try Disruption Tolerant Network for Building Monitoring Applications

Manussanun Buranachokphaisan*, Hideya Ochiai†, Hiroshi Esaki*
*Graduate School of Information Science and Technology, The University of Tokyo
Email: {kaew, hiroshi}@hongo.wide.ad.jp
†VLSI Design and Education Center, The University of Tokyo
Email: ochiai@vdec.u-tokyo.ac.jp

Abstract*—Wireless sensor network in building monitoring system (WSN-BMS) generally requires high reliability at scale. However, the performance of WSN-BMS is always deteriorated by the unidirectional links and dynamic nature of wireless links in the building. Potential-based Entropy Adaptive Routing (PEAR) protocol uses DTN-based approach to attain reliability and scalability over intermittently-connected wireless network, but this approach results high delivery latency such that it is not preferable in some applications of BMS. In this paper, we point out the problems leading to large delivery latency and propose FAST (Farther-Aim-Shorter-Try) forwarding scheme for PEAR in WSN-BMS. FAST modifies the current next-hop selection scheme to avoid the unidirectional links and combines DTN-based approach with the traditional routing scheme, i.e. link quality metric and retransmission, to improve the delivery latency. We implemented FAST to PEAR and evaluated its performance on WiFi-based UTMesh testbed with 16 node-floor scenario and 33 node-multistory scenario. The experiment results show that FAST outperformed PEAR regarding to delivery latency. It decreased 65.28% and 83.5% of median delivery latency compared to PEAR in floor and multistory scenario respectively.

Keywords*—WSN, DTN, BMS, Retransmission, Routing Metric, Unidirectional Link

I. INTRODUCTION

Wireless sensor network (WSN) has been proposed to use in building energy management system (BEMS). In BEMS, a number of sensor are deployed to monitor the environmental data and energy consumption which is used to analyze consumer behavior as well as manage electrical equipment in order to reduce energy consumption in the building. Since monitoring application generally requires the completeness of data, wireless sensor network in building monitoring system (WSN-BMS) must achieve high reliability at scale. However, WSN always suffers from dynamic nature of wireless links. To overcome this issue, the approach of delay/disruption tolerant network (DTN) which promises high delivery ratio even in intermittently-connected links is introduced to WSN-BMS.

DTN [1] describes the characteristics of the network which is high latency and intermittent connectivity, namely mobile ad-hoc networks and vehicular ad-hoc networks. Most DTN routing protocols were designed based on the assumption that the network contains the moving nodes, so node mobility or connectivity pattern is exploited to find the proper next-hop which generally the node that is close or moves toward sink or destination (We use sink and destination alternately in this paper). DTN relies on hop-by-hop reliability with store-and-forward mechanism. Nodes store the received messages in the buffers and periodically exchange information with their neighbors to confirm that the messages are delivered to the next-hop. ARQ (Automatic Repeat reQuest) with retransmission is not preferable as DTN expects large delivery latency.

WSN-BMS differs from DTN, so some approaches which show the good performance in DTN result the bad performance in WSN-BMS. WSN-BMS is static multi-hop wireless network. The routing metric of DTN is equivalent to hop metric in static WSN where nodes select the least hop path to deliver the messages. Hop metric is not a good metric for such network [2], [3] because nodes always choose long links with high loss rate. Without ARQ, the messages are delayed in the nodes' buffers when nodes fail to forward the messages. Moreover, bidirectional communication is necessary for hop-by-hop delivery. Howbeit, the unidirectional links often arise in static WSN [4], [5] and possibly interrupt the information exchange between a pair of nodes. These problems lead to large delivery latency of DTN-based WSN-BMS.

The works in [6] presented that even in static multi-hop wireless network, the connectivity was intermittent. The experiment with Potential-based Entropy Adaptive Routing (PEAR) protocol, the DTN-based routing protocol, on 50 node-scale testbed resulted 100% delivery ratio over 10 hops in the building scenario. This result showed that DTN routing protocol was able to achieve high reliability and scalability over the intermittently-connectivity. However, PEAR gave bad performance on delivery latency. The average and 99th percentile delivery latency was 238 and 700 seconds respectively. Although BMS is a delay-tolerant application [7], BMS needs near real-time data to visualize and feedback those data to consumers [8].

In this paper, we propose FAST (Farther-Aim-Shorter-Try) forwarding scheme for PEAR to reduce delivery latency in WSN-BMS. FAST combines DTN approach with the traditional WSN routing protocol by introducing retransmission scheme and link quality metric to PEAR. The sender challenges sending the messages to the next-hop selected by hop metric (Farther Aim). When the sender fails in forwarding messages, it has a chance to retransmit the messages once, but to the alternative next-hop with more reliable link which typically is shorter links (Shorter Try). To prevent the failure, FAST uses *Unidirectional link-Aware Next-hop Selection (ULANS)* scheme in the first challenge and *Delivery Predictability*, link quality metric for static DTN-based WSN, to find the alternative next-hop in retransmission.

We implemented our proposed scheme to demonstrate the effect of unidirectional links and the improvement of PEAR with FAST forwarding scheme. The experiment was carried out on UTMesh testbed [9] in the building monitoring scenario with two different deployment. The results show that FAST gave the best performance among all schemes concerning delivery latency.

The rest of this paper is structured as follows. Section II describes the concept of PEAR related to this work and explains the cause of high delivery latency in PEAR. FAST forwarding scheme is described in Section III. Section IV shows the experiment and evaluation of proposed scheme. We give a discussion in Section V. The related works are reviewed in Section VI. Finally, Section VII concludes this paper.

II. POTENTIAL-BASED ENTROPY ADAPTIVE ROUTING

PEAR acquires potential-based routing protocol (PBR) and hop-by-hop delivery with store-and-forward mechanism from DTN. In PBR, each node holds a *potential*, a positive scalar value representing the distance from each node to the sink in static scenario. Nodes located farther from the sink have larger potential. Nodes periodically update their potentials with neighbors by broadcasting 1-hop advertisement (ADV) and select the lowest potential-neighbor as the next-hop. Nodes exchange message information to check the status of each message before forwarding similar to hop-by-hop delivery. In this section, we review only forwarding scheme and hop-by-hop delivery of PEAR. More details can be found in [10].

A. Forwarding Scheme

Let N be a set of nodes in the network. Neighbor node of node $n \in N$ is denoted by $nbr(n)$. Potential of $n \in N$ for each destination $d \in N$ is defined by $V^d(n)$. The potential at the destination always ties to zero i.e. $V^d(d) = 0$.

PEAR's next-hop selection scheme is described as follows,

$$F^d_{max}(n) = \max_{k \in nbr(n)} \{V^d(n) - V^d(k)\} \quad (1)$$

$$NH^d(n) = \{k | k \in nbr(n) \wedge F^d_{max}(n) = V^d(n) - V^d(k)\} \quad (2)$$

where $F^d_{max}(n)$ is the maximum difference of node n's potential and its neighbors' potential and $NH^d(n)$ is the next-hop of node n for the destination d. First, nodes compare their potential with neighbors (Eq. 1). Then, nodes choose the neighbors that give the maximum potential difference to be their next-hops (Eq. 2). Since the potential indicates the distance from the destination, the next-hop is the neighbor located farthest toward the destination in the transmission range.

B. Hop-by-hop delivery

In store-and-forward mechanism, intermediate nodes copy and store the received messages in the buffer. Before forwarding the messages, nodes investigate if the next-hop already has the messages. We call this process as *Investigation*. A sender sends a request message containing message ID stored in its buffer to the next-hop. Then, the next-hop replies a response message consisting of the state of each message, which is already-received, not-received or delivered, to the sender. After that, sender sends only not-received messages to the next-hop. The investigation is illustrated in Fig. 1.

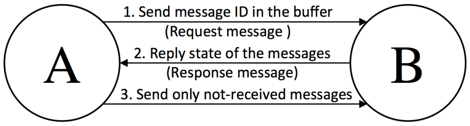


Fig. 1. The illustration of investigation in PEAR when A is a sender and B is a next-hop of A.

PEAR does not apply ARQ with retransmission in forwarding. Instead of that, nodes periodically investigate and send the messages in the buffers to make sure that all messages are delivered to the next hop. Nodes evict the messages in the buffer only when they receive delivered state from the response message or when time-to-live of the message reaches zero.

C. Problems

Even though PEAR achieves good performance in DTN, the different feature between DTN and static WSN causes large delivery latency when PEAR is implemented to WSN-BMS. Here, we point out three problems leading to delivery latency.

1) Unidirectional links: Notice that nodes send the messages only when they receive the response message from their next-hops. Bidirectional links are necessary in order to succeed investigation. Nonetheless, the current scheme considers only the potential in the next-hop selection. In reality, the unidirectional links possibly arises between two connected nodes when only one of two nodes can directly send the messages to the other. In this case, nodes cannot succeed investigation, consequently, the messages cannot be forwarded to the next-hop.

2) No ARQ: In traditional routing protocol, if the sender do not receive an acknowledgement (ACK) within a specific time, the sender resends the messages instantly until it receives ACK. On the other hand, PEAR relies on only the periodic investigation. When the sender does not receive the response message or fails to forward the data messages, the sender has to wait for the next investigation period.

3) Hop metric: The current routing scheme is similar to hop routing metric in static WSN. Choosing the next-hop based on hop count gives low latency in wired network or network with moving node, but this is not true for static WSN. Long links are likely to be disrupted and cause high message drop resulting similar to no ARQ problem.

III. FAST FORWARDING SCHEME

To solve the issues mentioned in last section, we proposed FAST (Farther-Aim-Shorter-Try) forwarding scheme for PEAR to improve delivery latency in static WSN. We first describe the overview of FAST, then we explain each mechanism in detail.

FAST integrates DTN-based approach and the traditional WSN routing scheme to decrease delivery latency in DTN-based routing protocol in WSN-BMS. The concept of FAST is to move the responsibility of forwarding messages to any neighbors within one investigation period. FAST still conducts the investigation before forwarding the messages to assure the delivery, but it applies retransmission to handle the loss during the investigation. FAST modifies the next-hop selection to avoid the unidirectional links and prepares more reliable next-hop for retransmission.

Ideally, the route with lower number of hops gives lower delivery latency. Thus, FAST attempts to minimize the number of hops as many as possible with hop metric. The sender challenge sending the request message to the *primary next-hop* which is the farthest neighbor with bidirectional links in the transmission range. When the sender fails in investigation i.e. the response message is not received, the sender makes a reinvestigation once. To prevent repeatedly failure, the sender resends the request message to the *alternative next-hop* chosen by link quality metric. Thanks to high quality links, the sender has higher probability of successful investigation, as well as, forwarding the messages to another node closer to the sink. FAST gives only one chance for the reinvestigation, thus the sender waits for next investigation period if it still fails in the reinvestigation.

A. Link Quality Estimation & Information Feedback

Link quality metric helps FAST providing the alternative path for reinvestigation. FAST seeks the path that nodes have high probability to succeed the investigation. In [11], the estimator measures the quality of each link by calculating the ratio of the number of received ADVs to the number of expected ADVs to find the delivery probability. FAST also estimates link by observing the received ADV. However, PEAR conducts the investigation before sending the data messages, therefore the estimator does not calculate the probability of delivery success directly. The estimator just predicts if nodes can succeed the investigation.

In this paper, we establish link quality and path quality metric called *Forward Predictability (P_F)* and *Delivery Predictability (P_D)* respectively, where $P_F, P_D \in [0,1]$. Forward predictability is the bidirectional link quality predicting if nodes can succeed and, as a result, have a chance to send the messages on each link. Delivery predictability is the multiplication of forward predictability forecasting the probability that nodes are able to succeed the investigation along the path.

FAST simply estimates link quality by calculating Advertise Reception Ratio (ARR). ARR_{nk} is the ratio of the number of ADVs received at n to the number of ADVs sent by k in a period of time, where $n, k \in N$ and $k \in nbr(n)$. Nodes calculate ARR periodically to track time-varying link quality. Then, forward predictability between node n and neighbor k and delivery predictability of node n for destination d are computed as follows,

Forward Predictability:

$$P_F(n,k) = ARR_{nk} \times ARR_{kn} \tag{3}$$

Delivery Predictability:

$$P_D^d(n) = \max_{k \in nbr(n)} \{P_D^d(k) \times P_F(n,k)\} \tag{4}$$

where $P_D^d(d) = 1$.

To calculate P_F and P_D, nodes share their link quality information with every neighbor by piggybacking those information with ADV. Each node estimates ARR of every link locally and embeds its P_D and the neighbor sequence consisting all neighbors' ID and ARR in ADV. After nodes receive ADV, they search for their ID in the neighbor sequence and update P_F with received ARR and local estimated ARR (Eq. 3). Then, nodes compute P_D and select the maximum multiple as its P_D (Eq. 4).

B. Unidirectional Link Avoidance

FAST detects the unidirectional links from the neighbor sequence in ADV. Nodes learn that the link connecting with any neighbor is unidirectional if they cannot find their ID in the neighbor sequence. FAST adds *Link Status (L_{nk})* entry in the neighbor table to classify the link. Link status is either available('A') or unavailable ('U'). It is set to 'A' when the link is bidirectional, while it is assigned to 'U' if the link is unidirectional.

C. Next-hop Selection Scheme

FAST uses two next-hop selection scheme for primary next-hop and alternative next-hop. The primary next-hop is chosen based on hop metric similar to PEAR. As we explained, PEAR's next-hop selection scheme chooses only the lowest potential node to be the next-hop without being aware of the existence of the unidirectional links. In order to avoid this problem, FAST modifies current next-hop selection and calls it *Unidirectional Link-Aware Next-hop Selection (ULANS)*. ULANS is defined as follows,

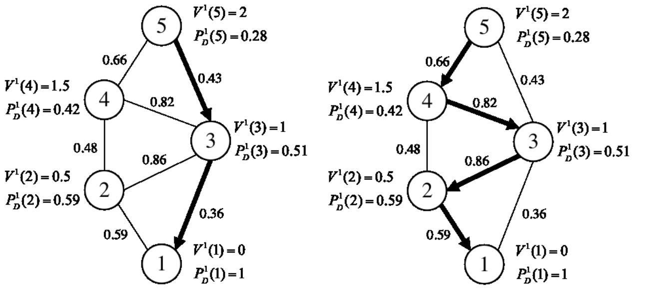
$$\begin{aligned} F^d(n) &= \max_{k \in nbr(n) \wedge L_{nk} = `A'} \{V^d(n) - V^d(k)\} && (5) \\ NH_{ULANS}^d(n) &= \{k | k \in nbr(n) \wedge \\ &\quad F^d(n) = V^d(n) - V^d(k)\} && (6) \end{aligned}$$

ULANS adds one more condition to address the unidirectional links. When nodes compare the potential with their neighbors, nodes consider only neighbors that bidirectional links exist. As a result, ULANS forces nodes to choose only the lowest potential neighbor with bidirectional link as the next-hop.

The alternative next-hop is selected by considering the quality of links along the path from source to sink. FAST uses delivery predictability to decide the alternative next-hop as shown in Eq. 7.

$$NH_{DP}^d(n) = \{k | k \in nbr(n) \wedge P_D^d(n) = P_D^d(k) \times P_F(n,k)\} \tag{7}$$

With this scheme, nodes select the neighbor that gives the highest delivery predictability among all neighbors for the reinvestigation. Fig. 2 illustrates the example of paths established from primary next-hop and alternative next-hop in FAST.



(a) Primary path (b) Alternative path

Fig. 2. The example of primary path and alternative path from node 5 to node 1.

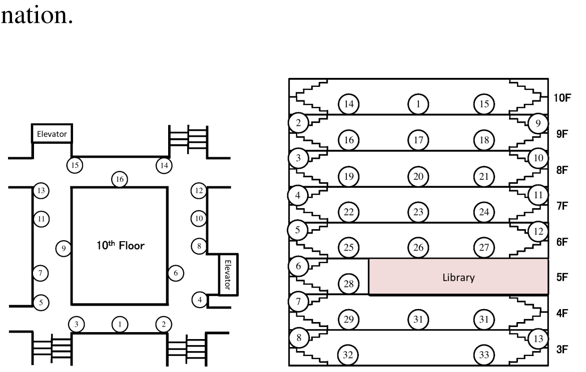
IV. EXPERIMENTAL EVALUATION

A. Experiment Setup & Scenario

The experiment was carried out on UTMesh testbed to evaluate our proposed scheme. The UTMesh nodes were operating with Linux embedded computer, Armadillo-200, with WiFi (IEEE802.11) module working in ad-hoc mode. All nodes were powered by batteries. The source nodes generated and sent messages to the sink every 30 seconds. ADV broadcast, system update and next-hop selection period were set to 5 seconds. Nodes retransmitted the messages in their buffers every 10 seconds.

Apart from the original PEAR, three more selection schemes were implemented to PEAR in order to study the effect of unidirectional links on DTN-based routing protocol and evaluate FAST by comparing FAST to PEAR (hop based) routing scheme and link quality based routing scheme. In the experiment result, PEAR-ULANS denotes the experiment with only ULANS. PEAR-DP represents the experiment with delivery predictability based next-hop selection scheme in Eq. 7. Finally, PEAR-FAST is the experiment with our proposed FAST.

UTMesh nodes were deployed on floor scenario and multistory scenario in Eng. Bldg. 2, The University of Tokyo. The configuration and detail of each deployment are shown in Fig. 3 and Table. I, respectively. In floor scenario, all nodes except node 1 were the source nodes, but in multistory scenario, only nodes deployed on the corridor were the source nodes. Nodes on the stairs just relayed the messages from the sources to the destination.



(a) (b)

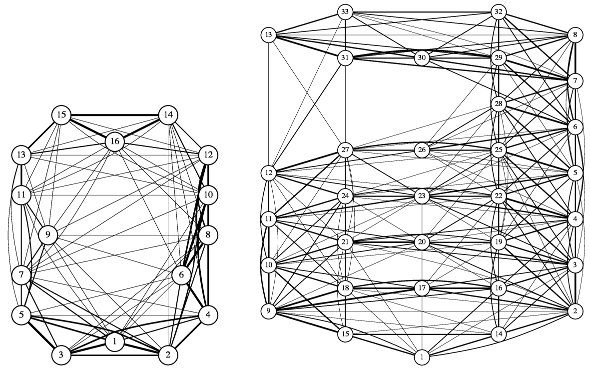
Fig. 3. The deployment configuration of (a) floor scenario and (b) multistory scenario.

TABLE I. THE DETAIL OF FLOOR AND MULTISTORY SCENARIO.

Detail	Floor	Multistory
Location (Floor)	10th	3rd-10th
Number of nodes	16	33
Source Node ID.	2-16	14-33
Destination Node ID.	1	1
Experiment time/scheme	1 hr	1.5 hrs

B. Features of the experimented networks

Fig. 4 shows the topology and connectivity of both floor and multistory scenario. The thickness of line implies the value of forward predictability of each link. The thinker line means higher forward predictability. Since our testbed was operated with WiFi, some nodes were connected across many floors in multistory scenario. Considering the links connecting nodes on the same floor, the shorter links mostly had higher forward predictability. However, the links connecting nodes deployed on the different floor mainly had low forward predictability. Even though those links were shorter than the links on the same floor, the signal strength was attenuated by the roof leading to low forward predictability.



(a) Floor scenario (b) Multistory scenario

Fig. 4. Topology and connectivity of experimented networks.

C. Performance Metrics

We studied and evaluated the proposed scheme by observing delivery ratio, delivery latency, hop count, copy count and buffer size. The definition of each metric is given as follows,

1) Delivery Ratio: the ratio of the number of messages received at the destination to the number of messages sent by the source.

2) Delivery Latency: the amount of time each message travels from source to destination.

3) Hop Count: the number of hops that each message travels from the source to the destination.

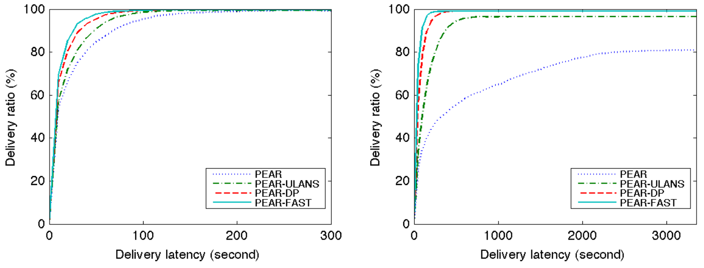
4) Copy Count: the number of copies of each message in the network during the delivery.

5) Buffer Size: the number of entries occupied in the buffer of each node.

The delivery latency is the main performance metric interested in this paper. The routing scheme should give low

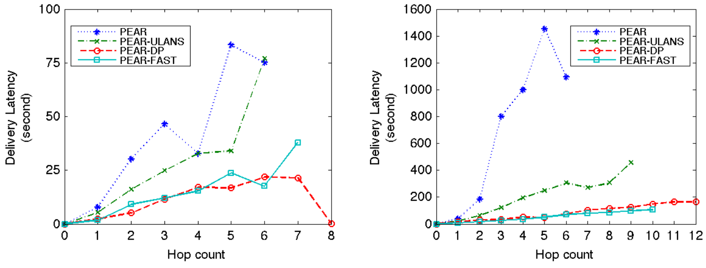
TABLE II. OVERALL PERFORMANCE.

Scenario	Scheme	Delivery Ratio (%)	Delivery Latency (sec) Median	Delivery Latency (sec) 99%	Avg. Hop Count	Avg. Copy Count	Avg. Buffer Size
Floor	PEAR	99.33	10.34	160	1.64	1.97	2.25
	PEAR-ULANS	99.44	9.4	100	1.98	2.61	2.49
	PEAR-DP	99.83	3.63	80	3.13	3.72	3.18
	PEAR-FAST	99.61	3.59	70	2.25	3.4	2.14
Multistory	PEAR	80.92	158.96	2460	2.28	2.95	50.68
	PEAR-ULANS	96.48	93.34	580	3.04	4.44	15.39
	PEAR-DP	98.97	40.12	320	4.75	6.62	12.71
	PEAR-FAST	98.84	26.62	60	3.74	6.38	6.4



(a) Floor scenario (b) Multistory scenario

Fig. 5. Delivery latency vs. Delivery ratio



(a) Floor scenario (b) Multistory scenario

Fig. 6. Hop count vs. Delivery latency

delivery latency, while preserving high delivery ratio. Hop count and copy count implies the delivery pattern of the routing scheme. Buffer size reflects on the congestion and resource constraint of nodes.

Overall performance of each metric is represented by the average value, except delivery latency. We use median and 99th percentile for delivery latency in order to avoid the outlier.

D. Evaluation Results

The overall performance is shown in Table. II. We analyzed the effect of unidirectional links and FAST performance as follows,

1) The effect of unidirectional links: Comparing the performance of PEAR and PEAR-ULANS, the unidirectional links did not influence much on delivery ratio and median delivery latency in floor scenario (Fig. 5a). On the other hand, the unidirectional links significantly reduced PEAR performance in multistory scenario (Fig. 5b).

We examined unidirectional link selection ratio (UDLSR) to prove that the unidirectional links were the cause of bad performance on delivery ratio and delivery latency in PEAR. UDLSR is the ratio of the number of times that a node selects the neighbor with unidirectional link as the next-hop to the number of times next-hop selection process is executed. Fig. 7 illustrates the distribution of UDLSR of PEAR and PEAR-ULANS. We can see that some nodes frequently chose neighbors with unidirectional links (40-60% of the experiment time) in PEAR, while all nodes avoided unidirectional links and gave low UDLSR in PEAR-ULANS.

PEAR-ULANS decreased 41.28% in the median and 76.43% in 99th percentile delivery latency from PEAR, but PEAR-ULANS increased hop count and copy count. The increment of hop count indicated that the unidirectional links were usually long links. Although the hop count was increased, when we compared the hop count with the delivery latency as shown in Fig. 6, the messages were delivered faster with the same hop count. The buffer size was also increased in floor scenario. With short range links, ULANS created and sent more copied messages in the network, so intermediate nodes received and stored more messages in their buffers. However, the buffer size was greatly decreased in multistory scenario. The reason was that PEAR deletes the messages in the buffer when the state *delivered* is informed. Seeing that the delivery latency was greatly decreased in multistory scenario, the messages were quickly deleted from the buffer.

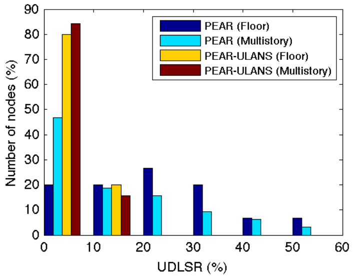


Fig. 7. The distribution of unidirectional link selection ratio (UDLSR).

2) FAST Performance: Even though ULANS already filtered out the unidirectional links, the rest of long links tended to be low quality links. PEAR-DP avoided those links and gave the better performance on delivery ratio and delivery latency. Since high quality links were particularly short links, PEAR-DP resulted more hop count and copy count compared to PEAR and PEAR-ULANS, but lower delivery latency with same number of hops (Fig. 6). PEAR-DP occupied more buffer entries in floor scenario, but it released buffer faster in multistory scenario similar to PEAR-ULANS.

PEAR-FAST showed the best performance among all schemes regarding to the delivery latency, especially in multistory scenario. PEAR-FAST reduced 65.28% and 83.5% in median delivery latency in comparison with PEAR on floor and multistory scenario, respectively. In addition, 99% of the messages were gathered within one minute, while other schemes took more than 5 minutes in multistory scenario.

Certainly, PEAR-FAST gave more hop count than PEAR, but lower than PEAR-DP. The reason for this was the messages were forwarded across some nodes when nodes succeeded in the investigation with primary next-hop. However, FAST lessened few copy count since FAST created multi-path delivery. The messages were possibly forwarded to both primary and alternative next-hop at the different transmission time. For example, when the sender failed in investigation with primary next-hop, but succeeded in reinvestigation, the sender forwarded the messages to the alternative next-hop at the first investigation time. At the next investigation time, the sender finished the investigation with primary next-hop, so the sender sent the messages to the primary next-hop. In this case, the messages delivered to both primary and alternative next-hop. Such situation could happen at every intermediate node, thus FAST still had large copy count comparing to PEAR and PEAR-ULANS.

PEAR-FAST resulted the lowest average buffer size in both scenarios. The effect of delivery latency on the buffer size was more dominant than the copy count even in floor scenario.

V. DISCUSSION

The combination of DTN approach and traditional routing scheme in FAST improved PEAR's performance significantly on delivery latency and delivery ratio. The improvement of delivery latency also benefited to the buffer size. However, there were a trade off between the redundancy and delivery latency. While PEAR minimized hop count, other schemes increased the number of hops as well as copied messages. This redundancy causes congestion in the network and also affects to energy efficiency of sensor node which is the important issue of WSN. Thus, we aim to reduce these redundancy in future works.

Our experiment also confirmed that hop metric is not a proper metric for static WSN regarding to delivery latency. PEAR-DP decreased delivery latency approximately 60% in both scenarios. Nevertheless, only link quality metric is not enough. Without the retransmission, the messages are likely to lost any time owing to the dynamic of wireless links. Therefore, the retransmission or retinvestigation in DTN-based approach is necessary for delivery latency improvement.

The performance of routing scheme depends on the deployment and network scale. We can see that floor and multistory scenario gave different results in some performance metrics. The unidirectional links did not influence much in floor scenario, thus all schemes achieve more than 99% delivery ratio. In addition, even though PEAR-ULANS and PEAR-DP delivered messages faster than PEAR, the congestion at intermediate nodes was higher in floor scenario. However, FAST achieved good performance in both scenarios.

VI. RELATED WORKS

In this section, we review the related works on unidirectional links, link quality metric and retransmission.

Many techniques of detecting and handling the unidirectional links were proposed. However, some were developed based on Ad-hoc On Demand Distance Vector (AODV) routing [12]–[14] and some exploited the unidirectional links in delivery [15], [16]. Seeing that PBR is different from AODV, in addition, hop-by-hop delivery requires only bidirectional links, those techniques cannot apply to PEAR.

Among existing link quality metric [11], [17]–[19], ETX (expected transmission count) is the most widely used in WSN routing protocol. ETX estimates the number of retranmission required from source to destination. This metric does not meet our requirement since we already fix the number of retransmission. Our estimator works similar to the one in [11], but ours estimates link quality with more precision. The metric *Delivery Predictability* has already proposed in [20]. However, it was defined in the context of mobility. Our work describes this metric in static condition.

Retransmission is the well-known mechanism to guarantee reliability [21]. All retransmission schemes resend the messages to the same next-hop. With hop metric, the messages are likely lost on the long links, so the sender has to retransmit the messages many times before the messages reach the next-hop. Consequently, retransmission could aggravate the congestion in the network [22]. The objective of applying retransmission in this work is different from those researches. FAST uses hop-by-hop delivery to promise 100% delivery ratio and retransmission to improve delivery latency. Thus, FAST limits the number of retransmission and provides alternative next-hop to avoid repeatedly message lost in retransmission.

VII. CONCLUSION

In this paper, we address the issues of DTN-based routing protocol in WSN-BMS. Some DTN approaches are not suitable for WSN-BMS due to the different features of DTN and WSN-BMS. Moreover, unawareness of unidirectional links also leads to high delivery latency. We proposed FAST forwarding scheme for PEAR to solve these issues. FAST integrates retransmission and link quality metric from WSN routing scheme with hop-by-hop delivery in DTN. FAST relieves the effect of unidirectional links by Unidirectional Link-Aware Next-hop Selection (ULANS) scheme and enhances investigation success with delivery predictability in retransmission. Our experimental evaluation with 16 node-scale floor scenario and 33 node-scale multistory scenario proved that FAST can considerably decrease delivery latency, while achieving high reliability and scalability in comparison with PEAR.

REFERENCES

[1] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*. ACM, 2003, pp. 27–34.

[2] R. Draves, J. Padhye, and B. Zill, "Comparison of routing metrics for static multi-hop wireless networks," in *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4. ACM, 2004, pp. 133–144.

[3] D. S. De Couto, D. Aguayo, B. A. Chambers, and R. Morris, "Performance of multihop wireless networks: Shortest path is not enough," *ACM SIGCOMM Computer Communication Review*, vol. 33, no. 1, pp. 83–88, 2003.

[4] R. Prakash, "A routing algorithm for wireless ad hoc networks with unidirectional links," *Wireless Networks*, vol. 7, no. 6, pp. 617–625, 2001.

[5] J. G. Jetcheva and D. B. Johnson, "Routing characteristics of ad hoc networks with unidirectional links," *Ad Hoc Netw.*, vol. 4, no. 3, pp. 303–325, May 2006.

[6] H. Ochiai, M. Nakayama, and H. Esaki, "Hop-by-hop reliable, parallel message propagation for intermittently-connected mesh networks," in *Proceedings of the 2011 IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks*, ser. WOWMOM '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 1–9.

[7] Y.-H. Jeon, "Qos requirements for the smart grid communications system," *International Journal of Computer Science and Network Security*, vol. 11, no. 3, pp. 86–94, 2011.

[8] J. E. Petersen, V. Shunturov, K. Janda, G. Platt, and K. Weinberger, "Dormitory residents reduce electricity consumption when exposed to real-time visual feedback and incentives," *International Journal of Sustainability in Higher Education*, vol. 8, no. 1, pp. 16–33, 2007.

[9] H. Ochiai, K. Matsuo, S. Matsuura, and H. Esaki, "A case study of utmesh: Design and impact of real world experiments with wi-fi and bluetooth devices," in *Applications and the Internet (SAINT), 2011 IEEE/IPSJ 11th International Symposium on*, July 2011, pp. 433–438.

[10] H. Ochiai and H. Esaki, "Mobility entropy and message routing in community-structured delay tolerant networks," in *Proceedings of the 4th Asian Conference on Internet Engineering*, ser. AINTEC '08. New York, NY, USA: ACM, 2008, pp. 93–102.

[11] M. D. Yarvis, W. S. Conner, L. Krishnamurthy, J. Chhabra, B. Elliott, and A. Mainwaring, "Real-world experiences with an interactive ad hoc sensor network," in *Parallel Processing Workshops, 2002. Proceedings. International Conference on*. IEEE, 2002, pp. 143–151.

[12] J.-B. Lee, Y.-B. Ko, and S.-J. Lee, "Euda: Detecting and avoiding unidirectional links in ad hoc networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 8, no. 4, pp. 63–67, Oct. 2004.

[13] V. Ramasubramanian and D. Mosse, "Bra: A bidirectional routing abstraction for asymmetric mobile ad hoc networks," *Networking, IEEE/ACM Transactions on*, vol. 16, no. 1, pp. 116–129, Feb 2008.

[14] M. Zuhairi, H. Zafar, and D. Harle, "On-demand routing with unidirectional link using path loss estimation technique," in *Wireless Telecommunications Symposium (WTS), 2012*, April 2012, pp. 1–7.

[15] L. Sang, A. Arora, and H. Zhang, "On exploiting asymmetric wireless links via one-way estimation," in *Proceedings of the 8th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, ser. MobiHoc '07. New York, NY, USA: ACM, 2007, pp. 11–21.

[16] B. Chen, S. Hao, M. Zhang, M. C. Chan, and A. Ananda, "Deal: Discover and exploit asymmetric links in dense wireless sensor networks," in *Sensor, Mesh and Ad Hoc Communications and Networks, 2009. SECON '09. 6th Annual IEEE Communications Society Conference on*, June 2009, pp. 1–9.

[17] D. S. De Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," *Wireless Networks*, vol. 11, no. 4, pp. 419–434, 2005.

[18] A. Adya, P. Bahl, J. Padhye, A. Wolman, and L. Zhou, "A multi-radio unification protocol for ieee 802.11 wireless networks," in *Broadband Networks, 2004. BroadNets 2004. Proceedings. First International Conference on*. IEEE, 2004, pp. 344–354.

[19] R. Fonseca, O. Gnawali, K. Jamieson, and P. Levis, "Four-bit wireless link estimation." in *HotNets*, 2007.

[20] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE mobile computing and communications review*, vol. 7, no. 3, pp. 19–20, 2003.

[21] M. A. Mahmood and W. K. Seah, *Reliability in wireless sensor networks: Survey and challenges ahead*. School of Engineering and Computer Science, Victoria University of Wellington, 2012.

[22] C. Wang, K. Sohraby, B. Li, M. Daneshmand, and Y. Hu, "A survey of transport protocols for wireless sensor networks," *Ieee Network*, vol. 20, no. 3, pp. 34–40, 2006.