



## Extension of ReInForM Protocol for $(m,k)$ -firm Real-time Streams in Wireless Sensor Networks

Bijun Li and Ki-II Kim\*, *Member, KIICE*

Department of Informatics, Engineering Research Institute, Gyeongsang National University, Jinju 660-701, Korea

### Abstract

For real-time wireless sensor network applications, it is essential to provide different levels of quality of service (QoS) such as reliability, low latency, and fault-tolerant traffic control. To meet these requirements, an  $(m,k)$ -firm based real-time routing protocol has been proposed in our prior work, including a novel local transmission status indicator called local DBP (L\_DBP). In this paper, a fault recovery scheme for  $(m,k)$ -firm real-time streams is proposed to improve the performance of our prior work, by contributing a delay-aware forwarding candidates selection algorithm for providing restricted redundancy of packets on multipath with bounded delay in case of transmission failure. Each node can utilize the evaluated stream DBP (G\_DBP) and L\_DBP values as well as the deadline information of packets to dynamically define the forwarding candidate set. Simulation results show that for real-time service, it is possible to achieve both reliability and timeliness in the fault recovery process, which consequently avoids dynamic failure and guarantees meeting the end-to-end QoS requirement.

**Index Terms:** Reliability, Real-time, Wireless sensor networks,  $(m,k)$ -firm deadline stream

### I. INTRODUCTION

A wireless sensor network (WSN) is comprised of small devices, known as sensor nodes, with computation, communication, and sensing capabilities as well as various resource constraints [1]. In recent years, the application of the complementary metal-oxide semiconductor (CMOS) camera and small microphones in WSNs allows them to gather not only data information but also background multimedia information [2]. However, given the limited resources of nodes, such as memory, energy consumption, CPU performance, and unstable factors of wireless communication, it is much more difficult for WSNs to meet the requirements of such real-time applications than traditional sensing applications, since it must handle the special quality of service (QoS) requirements

of real-time multimedia applications as well, such like high reliability and low latency of packet transmission [3]. Therefore, the fault tolerance characteristic is highly necessary for maintaining the robustness of the network, either through hardware or software. In this paper, we focus on the fault tolerance scheme in real-time routing protocols.

Most existing fault tolerance schemes for WSNs implement redundancy to recover packet loss and increase the probability of data delivery. This concept is established based on multiple copies of the same packet that travels through a multipath to the sink. In [1], a multipath-based reliable information forwarding protocol called ReInForM was used to deliver the data at desired levels of reliability to recover failures caused by channel errors. It controls the number of paths required for the desired reliability using only local knowledge of channel

Received 05 July 2012, Revised 08 August 2012, Accepted 16 August 2012

\* Corresponding Author E-mail: kikim@gnu.ac.kr

**Open Access** <http://dx.doi.org/10.6109/jicce.2012.10.3.231>

print ISSN:2234-8255 online ISSN:2234-8883

© This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Copyright © The Korea Institute of Information and Communication Engineering

error rates and does not require any maintenance of the multipath. However, the forwarding node selection mechanism of ReInForM considers only the required reliability so that it cannot be applied to meet the timeliness requirement of real-time services. In [4], a dynamic jumping real-time fault-tolerant routing protocol (DMRF) was proposed to handle the potential faults of the network such as failure, congestion, and void regions. Each node could use the remaining transmission time of the data packets and the state of the forwarding candidate node set to determine the next hop. It is designed to guarantee the performance of real-time services, although only soft real-time can be satisfied due to its hop-by-hop transmission mode. For some specific applications such as multimedia transmission in WSNs, which requires a firm real-time guarantee, it is not enough to meet the requirements. Moreover, without an end-to-end QoS guarantee, the intermediate nodes cannot adjust the delivery to the real-time performance effectively. SPEED [5] introduced a real-time communication routing protocol which can provide desired delivery speed across the sensor networks through a combination of feedback control and non-deterministic geographic forwarding. However, it also supports only soft real-time transmission since it does not take into account the packet deadline of a real-time stream, which may consequently lead to end-to-end dynamic failure of transmissions.

In our prior work [6], a firm real-time routing protocol was presented to guarantee end-to-end QoS of real-time services, which becomes an issue for [5], using a scheduling policy called distance-based priority (DBP) [7]. This scheduling policy is used to evaluate the QoS performance of real-time streams and to better service them according to their  $(m,k)$ -firm requirements. A real-time message stream is considered to have an  $(m,k)$ -firm guarantee requirement that at least  $m$  out any  $k$  consecutive messages from the stream must meet their deadlines to ensure adequate QoS. This concept was improved to be a local transmission status indicator called local DBP (L\_DBP), which makes the intermediate nodes aware of the local stream transmission status, and also distinguishes fault categories as congestion and link failure, so that different fault recovery schemes could be implemented according to it. The simulation result showed that for real-time services such as multimedia transmission, this protocol performs much better than soft real-time routing protocols for end-to-end QoS guarantee. Therefore, in this paper, we introduce a novel failure recovery scheme for  $(m,k)$ -firm streams to advance our prior work. To the best of our knowledge, no existing work has implemented failure recovery specifically for  $(m,k)$ -firm based real-time streams. [8] shows that the most popular

approach for fault tolerance and recovery is to use a multipath between the source node and sink in two manners: disjoint multipath and braided multipath. The former one constructs a number of alternate paths which are node/link disjoint, with the primary path and secondary paths while the latter one constructs alternate paths in a braid partially overlaying the primary path. Considering the resource constraints of WSNs and the fact that braided paths are less expensive than disjoint paths in terms of latency and overhead, we use a braided multipath mechanism in this paper as the principal framework. Based on it, each node in network can execute a delay-aware forwarding candidate selection algorithm to choose a number of nodes for establishing a multipath and implement a redundancy-based link failure recovery scheme.

The rest of this paper is organized as follows. The proposed scheme is described in Section II. Simulation results are shown in Section III. Section IV concludes the proposed scheme.

## II. PROPOSED SCHEME

The key components of the proposed failure recovery scheme are discussed in this section. Taking advantage of the  $(m,k)$ -firm model, it is available for our scheme to guarantee the end-to-end QoS of firm real-time services in terms of both reliability and timeliness. This scheme includes two subsystems, denoted as the sink-source system and local system, to implement end-to-end/intermediate performance evaluation and failure recovery. Both systems use stream DBP ( $G\_DBP$ ) and  $L\_DBP$  values as well as the deadline information of packets in their executions of the principal forwarding candidate selection algorithm.

The first metric for end-to-end QoS performance evaluation is  $G\_DBP$ , which shows the quality of one real-time stream. It is measured at the sink, using the equation in [6].

$$G\_DBP_{s(x)} = k_{s(x)} - l_{s(x)}(m_{s(x)},s) + 1 \quad (1)$$

where  $G\_DBP_{s(x)}$  is the measured DBP value of stream  $x$  at the sink,  $k_{s(x)}$  comes from the required  $(m,k)$ -firm of stream  $x$ ,  $l_{s(x)}(m_{s(x)},s)$  denotes the position (from the right) of the  $m$ -th deadline meeting in the current state  $s$  of stream  $x$ .

The sink periodically sends back the value of  $G\_DBP_{s(x)}$  to the corresponding source nodes, to inform them with the real-time end-to-end QoS of the streams they generated. After a source node receives the feedback of  $G\_DBP_{s(x)}$ , it adds the value to the header of the packets it generates and then forwards them to the

intermediate nodes. Therefore, all nodes on the transmission path would be notified with the end-to-end QoS of the current stream they are transmitting, and based on it, also considering  $L\_DBP$ , these nodes could make decisions of whether redundancy of packets on multipath is required. Also, the  $G\_DBP_{s(x)}$  value helps the corresponding source node to deal with the link failure that occurred between itself and the next hop. Details are discussed in Section II-A.

For local transmission status evaluation on the intermediate nodes, we continue using the value of  $L\_DBP$ , which was proposed in our prior work [5]. Different from  $G\_DBP$ , the calculation of  $L\_DBP$  is as follows:

$$L\_DBP_{s(x)_i} = k_{s(x)} - m_{s(x)} - c_{s(x)_j} - f_{s(x)_j} \quad (2)$$

where  $L\_DBP_{s(x)_j}$  stands for the distance to failure on node  $i$ ,  $k$  and  $m$  are set as the value of the required  $(m,k)$ -firm;  $c_{s(x)_j}$  and  $f_{s(x)_j}$  denote the congestion and link failure levels of downstream node  $j$ , respectively.

By Eq. (2) each node on the transmission path could obtain the information of the real-time status of all traffic passed through it, and also distinguish the causes of network faults as congestion or link failure at the same time. Based on the feedback from the downstream node, the results of  $L\_DBP$  can be categorized as follows [5]. 1) While an upstream node receives periodic beacons from the downstream node during a predefined time interval. 2) If it receives ACK, both  $c_{s(x)_j}$  and  $f$  keep the same. 3) If not,  $f_{s(x)_j}$  keeps the same while  $c_{s(x)_j} + 1$ , which indicates congestion occurring. 4) While upstream node does not receive periodic beacons from the downstream node during the time interval, then  $f_{s(x)_j} + 1$ , which indicates link failure happening.

In this paper, using the value of  $L\_DBP$ , intermediate nodes could calculate the required number of forwarding nodes and then make decisions for the multipath. Details are discussed in the following part.

The principal contribution of this paper is the algorithm used for nodes to choose the optimal forwarding nodes for redundancy on a multipath. Compared with previous fault tolerant schemes or real-time routing protocols, the proposed scheme makes it possible to establish multiple transmission paths with a bounded latency during transmissions. It is guaranteed that all selected nodes for forwarding multiple copies of packets can relay the packets in a timely manner. Due to the features of real-time services, packets loss would lead to not only end-to-end dynamic failures but also the timeout of a certain number of packets. The potential high latency which is involved in the use of a multipath may severely influence the quality of packets received by the sink. Thus, we present a new delay-aware algorithm for dynamically choosing the optimal forwarding nodes,

which can guarantee both required reliability and bounded delay, to improve the performance of prior work, and make it more adaptable for real-time services than others. The algorithm can be described in Table 1 as follows.

This algorithm shows how upstream node  $i$  makes the decision about which downstream node can be chosen as a candidate node. Firstly, if both  $G\_DBP_{s(x)}$  and  $L\_DBP_{s(x)_i}$  values are smaller than 0, which indicates a negative stream status and negative local transmission status as well, and also that the link failure level  $f_{s(x)_i}$  is not equal to 0, then the link failure recovery scheme would be touched off to determine a proper set of candidate nodes from its neighbors, for multipath establishment. The maximum allowable delay of the current stream is calculated to be within this time period so that packets arrived at sink could be considered useful. Thus, for an upstream node  $i$ , to find a proper forwarding candidate among all its downstream nodes in the radio range, is to choose the one that can keep a positive status of a stream that has an even more strict deadline requirement than the current one, and then add this node into the candidate nodes set. We assume that the needed information can be collected by periodic beaconing. That is, all nodes in that set are supposed to be able to guarantee a bounded delay of packets.

This algorithm makes it possible for nodes to establish a multipath in the link failure recovery for real-time services. However, not all nodes in that set are needed in the case of densely employed network such that there may be many candidates available from which to choose. Consequently, a calculation for the forwarding path selection should be done according to the actual situation. Two subsystems in Section II are introduced here for sink-source node and intermediate nodes to make decisions for choosing the optimal number of alternative paths needed for redundancy, from their candidate node set, respectively.

**Table 1.** Algorithm: candidate node selection

$G\_DBP_{s(x)}$ : stream DBP value of stream $x$
$L\_DBP_{s(x)_i}$ : local DBP value of stream $x$ on upstream node $i$
$L\_DBP_{s(x)_j}$ : local DBP value of stream $x$ on downstream node $j$
$f_{s(x)_i}$ : link failure level of stream $x$ on node $i$
$Deadline_{s(x)}$ : deadline of packet of stream $x$
$Delay\_trans\_i,j$ : delay of transmission between $i,j$
$T\_maxdelay_{s(x)}$ : maximum allowable delay of stream $x$
$Num\_dnode\_i$ : number of downstream nodes of node $i$
$D_{s(x)_j}$ : deadline of stream $x$ on node $j$
$Num\_strm\_j$ : number of streams on node $j$
$S\_candi\_i$ : candidate nodes set of node $i$

**PSEUDO-CODE RUNS AT NODE I IN EACH ROUND**

```

1: if  $G\_DBP_{s(x)} \leq 0$  //stream can't meet occurred
2:   if  $L\_DBP_{s(x)_i} \leq 0$  //local fault occurred
3:     if  $f_{s(x)_i} \neq 0$  //link failure occurred
4:        $T\_maxdelay_{s(x)} = Deadline_{s(x)} - Delay\_trans\_i,j$ 
5:       for j from 1 to Num_dnode_i //for all downstreams
6:         for x from 1 to Num_strm_j //for all streams
7:           if  $D_{s(x)_j} < T\_maxdelay_{s(x)} \&\& L\_DBP_{s(x)_j} > 0$ 
              //good status of reliability and timeliness
8:             j is in  $S\_candi\_i$ 
9:           end if
10:        end if
11:     end if
12: end if

```

**A. Sink-source Node System**

For the source node, the most useful information is the  $G\_DBP$  value of the stream it generated, as feedback from the sink. Therefore, the adapted number of alternative paths could be calculated using the following equation:

$$P_{-fwd-i} = \min\{|G\_DBP_{s(x)} - 1|, S_{-candi-i}\} \quad (3)$$

where  $P_{-fwd-i}$  is the optimal number of forwarding paths for multipath establishment.

This equation can be also used when the source node receives backpressure from its downstream node, which indicates the failures of some links on the primary path and the intermediate nodes have no candidate to choose, so that it is necessary to start using multipath at the source node.

**B. Intermediate System**

The local system includes all intermediate nodes and the links between them. Since the  $L\_DBP$  value is the most useful information for the intermediate nodes to evaluate the transmission status, it is used in Eq. (4) for the selection of the optimal number of alternative paths. The equation is as follows:

$$P_{-fwd-i} = \min\{|L\_DBP_{s(x)_i} - 1|, S_{-candi-i}\} \quad (4)$$

Similar to Eq. (3), the number of forwarding paths is calculated adaptively with respect to the candidate node set and the actual situation.

In case of severe channel errors, or a sparsely

employed network, it is possible that once an intermediate node detects link failure on a primary path, it finds no candidates for a multipath itself, so it sends backpressure to its upstream node. Therefore, the backpressure may finally reach the source node, and Eq. (3) would be executed for recovery as mentioned in Section II-A.

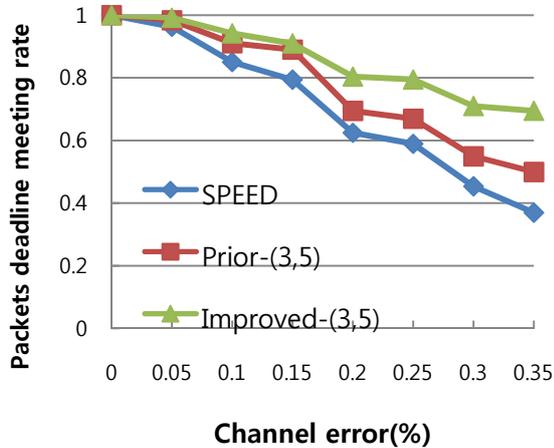
**III. PERFORMANCE EVALUATION**

Performance of the proposed scheme is proven by simulation. We chose NS-2 as the simulator and a uniform topology that includes 100 nodes in an area of  $200 \times 200$  m. The propagation model is set to be two-ray ground, and the protocols for the physical and medium access control (MAC) layer are set to be wireless-Phy and 802.11, respectively. The radio range is set to be 30 m for nodes to transmit 1,000 bytes packets on the bandwidth of 2 Mb/s.

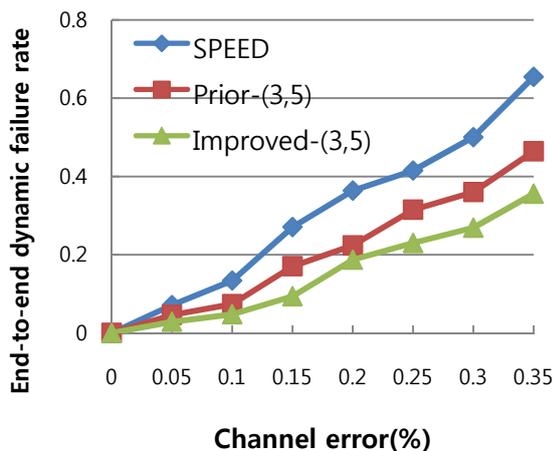
The proposed failure recovery scheme is used to improve the performance of our prior work, so the comparison was carried out with SPEED, our prior algorithm, and improved one. In order that the performance of failure recovery can be shown clearly and distinctly, we did not involve much cross-traffic so that most nodes in network would not suffer from heavy traffic load and its introduced packet loss. Channel error, as the principal cause of link failure in WSNs, is normally distributed across the network and increasing at a proportional speed from 0% to 35%. Simulations show the influence of various channel error rates on both the rate of meeting the packet deadline and the end-to-end dynamic failure rate.

In Fig. 1, as a critical metric for real-time services, the packet deadline meeting rate is measured for all three algorithms. The reason for missing the deadline is considered to be the introduced high latency of failure recovery methods. When link failure has occurred, each algorithm starts up a recovery scheme to handle the problem. For SPEED, backpressure-based feedback is used to inform the upstream node to switch the next hop when the downstream finds no next hop is available; however, there is no method to guarantee the switched node could not choose a path that would not increase the end-to-end delay of transmissions. Fig. 1 shows an obvious decline of the deadline meeting rate of SPEED after the mean channel error rate reached 10%. In the case of our prior work, with the same  $(m,k)$ -firm requirement and the lack of multipath latency evaluation, it also introduced high latency to the packets transmitted on alternative paths during link failure recovery, from the moment the channel error is increased to 15%. Thanks to the proposed candidate node selection algorithm, the improved work performs much more steadily than the

other two. Missing a deadline increased in a slow and stable manner, even at a channel error rate of 30% and 35%.



**Fig. 1.** Packets deadline missing incurred by SPEED, our prior algorithm with (3,5)-firm, and our improved algorithm with (3,5)-firm.



**Fig. 2.** End-to-end dynamic failure incurred by SPEED, our prior algorithm with (3,5)-firm, and our improved algorithm with (3,5)-firm.

The second simulation (Fig. 2) shows the impact of an increasing channel error to the end-to-end dynamic failure, which includes the verification of both reliability and timeliness, and is considered to be the most important QoS metric. Packet loss caused by link failure would lead to stream dynamic failures as well as packet deadline missing. For SPEED, when packet loss occurs due to link failure, no recovery scheme such as retransmission or redundancy is used to handle it, and the feedback process may not be timely. Our prior work has the same problem as mentioned in the last simulation,

that it cannot guarantee the multipath generated for redundancy could meet the timeliness requirement of the stream. Both algorithms lead to a rapidly increasing rate of dynamic failures from the channel error rate of 10%. The curve of the improved algorithm is much smoother than others and indicates a better capability of fault tolerance.

## IV. CONCLUSIONS

In this paper, a new link failure recovery scheme is proposed for  $(m,k)$ -firm based real-time services in WSNs. It utilizes the values of  $G\_DBP$  and  $L\_DBP$ , as well as deadline information of packets, to implement a restricted redundancy of packets on a multipath. The contribution of this scheme is the delay-aware algorithm it used for forwarding candidate node selection before multipath establishment. This algorithm is specific for firm real-time services such that it can evaluate the delay of the path when it makes a decision for selecting forwarding candidates. It could be guaranteed that the alternative paths for redundancy would meet the timeliness requirement and also raise the reliability of transmissions. Simulations show that the proposed scheme brings a remarkable performance improvement to our prior work.

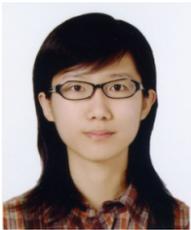
## ACKNOWLEDGMENTS

This research was supported by the Basic Science Research Program through the National Research Foundation (NRF) of Korea funded by the Ministry of Education, Science and Technology (2012-0001761) and the Ministry of Knowledge Economy (MKE), under the Information Technology Research Center (ITRC) support program supervised by the National IT Industry Promotion Agency (NIPA-2012-H0301-12-3003).

## REFERENCES

- [1] B. Deb, S. Bhatnagar, and B. Nath, "ReInForM: reliable information forwarding using multiple paths in sensor networks," *Proceedings of the 28th Annual IEEE International Conference on Local Computer Networks*, Konigswinter, Germany, pp. 406-415, 2003.
- [2] A. Sharif, V. Potdar, and E. Chang, "Wireless multimedia sensor network technology: a survey," *Proceedings of the 7th IEEE International Conference on Industrial Informatics*, Cardiff, Wales, pp. 606-613, 2009.
- [3] J. F. Mingorance-Puga, G. Macia-Fernandez, A. Grilo, and N. M. C. Tiglao, "Efficient multimedia transmission in wireless sensor network," *Proceedings of the 6th EURO-NF Conference on Next Generation Internet*, Paris, France, pp. 1-8, 2010.

- [4] G. Wu, C. Lin, F. Xia, L. Yao, H. Zhang, and B. Liu, "Dynamical jumping real-time fault-tolerant routing protocol for wireless sensor networks," *Sensors*, vol. 10, no. 3, pp. 2416-2437, 2010.
- [5] T. He, J. A. Stankovic, C. Lu, and T. Abdelzaher, "SPEED: a stateless protocol for real-time communication in sensor networks," *Proceedings of the 23rd International Conference on Distributed Computing Systems*, Providence, RI, pp. 46-55, 2003.
- [6] B. Li and K. I. Kim, "A novel routing protocol for (m, k)-firm-based real-time streams in wireless sensor networks," *Proceedings of the IEEE Wireless Communications and Networking Conference*, Shanghai, China, pp. 1715-1719, 2012.
- [7] M. Hamdaoui and P. Ramanathan, "A dynamic priority assignment technique for streams with (m, k)-firm deadlines," *IEEE Transactions on Computers*, vol. 44, no. 12, pp. 1143-1451, 1995.
- [8] H. Alwan and A. Agarwal, "A survey on fault tolerant routing techniques in wireless sensor networks," *Proceedings of the 3rd International Conference on Sensor Technologies and Applications*, Athens/Glyfada, Greece, pp. 366-371, 2009.



**Bijun Li**

is currently working toward a Ph.D at Gyeongsang National University, Jinju, Korea. Her research interests include wireless sensor networks, and routing protocols



**Ki-II Kim**

received the M.S. and Ph.D. degrees in computer science from ChungNam National University, Daejeon, Korea in 2002 and 2005, respectively. He is currently with the Department of Informatics at Gyeongsang National University. His research interests include routing for MANET, QoS in wireless networks, and multicast and sensor networks.