Performance Comparison of Packet Combining Based Error Recovery Schemes for Wireless Sensor Network

Sheng-Shih Wang, Pei-Chun Lee, and Kan-Yen in
Department of Information Management, Minghsin University of Science and Technology, Hsinchu, Taiwan
Email: sswang@must.edu.tw, pjlee@must.edu.tw, yy3282001@yahoo.com.tw

***Abstract*—Although existing researches have considered the automatic repeat request and forward error correction schemes to provide reliable communication, the former suffers frequent retransmissions and the latter incurs bandwidth overhead in wireless sensor networks. In order to reduce the number of retransmissions and support efficient correction, existing studies have proposed many packet combining based error recovery schemes for wireless sensor networks. This study investigates the recovery performance of these packet combining based error recovery schemes. This study provides the mathematical analysis and simulation results, using the bit error rate and the packet length as major parameters. The results present that different schemes show a tradeoff between the recovery speed and storage space overhead.**

***Index Terms*—packet combining, error recovery, automatic repeat request (ARQ)**

I. Introduction

Wireless sensor networks (WSNs) consist of large numbers of tiny and autonomous wireless devices, called sensor nodes [1]. The WSN is typically used in a variety of applications, such as environmental and habitat monitoring, object detection, and military surveillance [2]. A lot of WSN applications require the message transmission with a given reliability [3]. However, wireless links in WSNs are likely to be error-prone because of the effect of multipath fading, shadowing and interference, and so on. This causes that receivers gain a low packet reception ratio and transmitters have to retransmit the packets if reliable transmission should not be guaranteed. To tackle this challenge, previous studies have proposed many error recovery schemes, which are primarily based on the concept of error control.

Error detection and error correction are underlying strategies to error control. The automatic repeat request (ARQ) [4] and the forward error correction (FEC) [5] are typical error detection and error correction mechanisms, respectively. Unfortunately, the ARQ mechanism may encounter frequent retransmissions when the channel quality of networks is extremely awful, thereby degrading the system throughput. On the other hand, the FEC mechanism is more likely to cause additional recovery delay and potential bandwidth overhead.

The literature has argued that even if all the received copies of packets are erroneous, it may be possible to combine these copies to recover the correct packet [6]. Recent studies enlightened by [7] have proposed a lot of mechanisms to retrieve original packets for a variety of wireless networks [8]-[10]. The main idea behind these schemes is packet combining. The schemes consider all the corrupted receptions of a given packet and then conduct an exhaustive search to generate the candidate packets which may contain the original packet. Whenever the receiver obtains all the candidate packets, it then checks the cyclic redundancy check (CRC) residue of each candidate packet to find the correct one. The EARQ scheme [8] adopts this concept as an underlying strategy.

To achieve reliable data transmission in wireless sensor networks, a simple packet combining (SPaC) error recovery scheme [9] and a destination packet combining (DPC) [10] are proposed. The SPaC scheme introduces a merge procedure, hereafter called SPaC-Merge, which only buffers the latest two corrupted packets to generate the candidate packets. A lightweight and straightforward error recovery scheme, called the DPC, is proposed to exploit a majority voting strategy to derive the possible value (‘0’ or ‘1’) of each data bit. The DPC depends on these values of data bits to recover the original packet.

This paper evaluates the recovery performance of packet combining based error recovery schemes, considering the bit error rate (BER) and packet length as the primary parameters. The recovery performance of the schemes, including EARQ, SPaC-Merge, and DPC, are compared through the analysis and the simulation. The results show that the EARQ scheme has a better recovery speed than the SPaC-Merge and the DPC for a high bit error rate and a large packet length. This is because the EARQ buffers all the received corrupted packets to generate the candidate packets based on the buffered packets, thereby having a higher probability to retrieve the correct packet than both SPaC-Merge and DPC. However, the EARQ needs a considerable amount of storage space to buffer all the corrupted packets and candidate packets. The results also present that the three schemes achieve an approximate recovery speed in case of a good channel quality and a small packet length.

Manuscript received Dec 21, 2012; revised Feb 21, 2013.

doi: 10.12720/lnit.1.1.1-5

The rest of this paper is organized as follows. Section II introduces the packet combining based error recovery schemes investigated in this paper. Section III presents the mathematical analysis of recovery performance. Section IV shows the simulation results. Section V concludes this paper.

II. Preliminaries

To recover transmission errors, Chakraborty et al. extend the traditional ARQ mechanism and propose a packet combining-based scheme, called EARQ [8]. In the EARQ, receivers store all the corrupted packets and perform the XOR operation on each of any two of corrupted packets to derive multiple combination results, and depend on the reception status (i.e., '0' indicates correct or '1' indicates erroneous) of each bit of each XORed result to derive all the candidates of the original packet using an exhaustive search approach. Receivers then perform the CRC technique to test each candidate packet to determine which one is the original packet.

The EARQ intends to recover the correct packet from all the received corrupted packets. It significantly requires a considerable amount of buffer space to store all the corrupted packets, and thus it is more unlikely to be considered on storage-limited devices, such as sensor nodes. Motivated by reduction of retransmissions and buffer space, a lightweight error recovery scheme, called SPaC, is proposed for wireless sensor networks [9]. Specially, a 'Merge' procedure in the SPaC also performs packet combining to retrieve the original packet. Unlike in the EARQ, receivers in the SPaC store the latest two corrupted packet instead of all the corrupted packets. Once receiving a corrupted packet, receivers XOR this packet and the buffered packet, and derive the candidates of the original packets according to the reception status of bits in the XORed result. Then, the CRC test is performed to retrieve the original packet.

The DPC scheme is famous for its simplicity and efficiency due to its low computation overhead, and thus it is more suitable for time-critical applications [10]. The main idea behind the DPC is that receivers store the successively corrupted packets and use a majority voting strategy on these packets to guess the original packet. In this majority voting strategy, the value of each bit of the determined packet depends on the values of the corresponding bit in all the corrupted packets. For a bit, if the number of corrupted packets in which the corresponding data bits are '1' equals or exceeds half the number of corrupted packets, receivers regard this bit of the original packet as '1'. Otherwise, this bit of the original packet that receivers presume is '0.'

III. Analysis of Recovery Performance

To analyze the recovery performance of the EARQ, SPaC-Merge, and DPC, this study considers the following assumptions: (1) All packets are fixed in length; (2) The numbers of errors in corrupted packets are independent, identically distributed random variables with a binomial distribution; (3) Packet transmission uses the stop-and-wait automatic repeat request approach; (4) The transmission of ACK packets is always successful.

Let p_e denote the BER. Use k_r and $p(k_r)$, where $r \geq 2$, to respectively indicate the number of error bits in the *r*-th corrupted packet and the probability of having k_r error bits in the *r*-th corrupted packet, for $r \geq 2$. Assume that the packet length is *L* bits. Because the numbers of errors in corrupted packets are independent, we have

$$p(k_r) = \binom{L}{k_r} \cdot (p_e)^{k_r} \cdot (1-p_e)^{L-k_r}. \cdot \tag{1}$$

Given the packet length and bit error rate, the success probability of recovery of EARQ at the r-th round, termed $p_r^{\mathbf{E}}(s)$, is derived as

$$\begin{aligned} p_r^{\square}(s) = &\left[1 - \sum_{i=1}^{r-1} p_i^{\square}(s)\right] \\ &\cdot \left[(1-p_e)^L + (1-(1-p_e)^L) \cdot (1-\alpha_r^{\square})\right], \end{aligned} \tag{2}$$

where $\alpha_r^{\mathbf{E}}$ is the conditional probability that, provided P_r^{err} arrives, it has hidden errors with all corrupted packets [8]. Let $p_{i,j}^{\mathbf{E}}(h)$ be the probability that two corrupted packets, respectively having k_i and k_j error bits ($i < j$), have a hidden error of EARQ. It can be given as

$$p_{i,j}^{\square}(h) = 1 - \frac{\sum_{k_i=1}^{L} \sum_{k_j=k_i}^{L} \binom{L}{k_i} \cdot \binom{L-k_i}{k_j}}{\sum_{k_i=1}^{L} \sum_{k_j=k_i}^{L} \binom{L}{k_i} \cdot \binom{L}{k_j}}. \tag{3}$$

Recall that the occurrences of hidden errors in all corrupted packets are assumed to be independent events. Therefore, we derive $\alpha_r^{\mathbf{E}}$ as

$$\alpha_r^{\square} = \frac{\sum_{k_1=1}^{L} \sum_{k_2=1}^{L} \cdots \sum_{k_r=1}^{L} p(k_1) p(k_2) \cdots p(k_r) \cdot \prod_{1 \leq i, j \leq r} p_{i,j}^{\square}(h)}{\left(1-(1-p_e)^L\right)^r \cdot \prod_{l=2}^{r-1} \alpha_l^{\square}}. \tag{4}$$

Taking this $\alpha_r^{\mathbf{E}}$ in (2), we can obtain $p_r^{\mathbf{E}}(s)$.

Let the success probability of recovery of SPaC-Merge at the *r*-th round, termed $p_r^{\mathbf{S}}(s)$. It can be derived as

$$\begin{aligned} p_r^{\square}(s) = &\left[1 - \sum_{i=1}^{r-1} p_i^{\square}(s)\right] \\ &\cdot \left[(1-p_e)^L + (1-(1-p_e)^L) \cdot (1-\alpha_r^{\square})\right], \end{aligned} \tag{5}$$

where $\alpha_r^{\mathbf{S}}$ is the conditional probability that, provided P_r^{err} arrives, it has hidden errors with the *r*-th and *r*-th corrupted packets. Let $p_{i,j}^{\mathbf{S}}(h)$ indicate the probability that two corrupted packets, respectively having k_i and k_j error bits ($i < j$), have a hidden error of SPaC-Merge.

We have $p_{i,j}^{\mathbf{S}}(h) = p_{i,j}^{\mathbf{E}}(h)$. Recall that the occurrence of hidden errors in the latest two corrupted packet causes receivers to fail to recover the original packet. Thus, $\alpha_r^{\mathbf{S}}$ can be derived as

$$\alpha_r^{\mathbf{S}} = \frac{\sum_{k_{r-1}=1}^{L}\sum_{k_r=1}^{L} p(k_{r-1})p(k_r)\cdot p_{i,j}^{\mathbf{S}}(h)}{\left(1-(1-p_e)^L\right)^r \cdot \prod_{l=2}^{r-1}\alpha_l^{\mathbf{S}}}. \quad (6)$$

Taking this $\alpha_r^{\mathbf{S}}$ in (6), we can obtain $p_r^{\mathbf{S}}(s)$.

Given the packet length and the bit error rate, the success probability of recovery of DPC at the r-th round, denoted as $p_r^{\mathbf{D}}(s)$, is derived as

$$p_r^{\mathbf{S}}(s) = \left[1-\sum_{i=1}^{r-1} p_i^{\mathbf{S}}(s)\right] \cdot\left[(1-p_e)^L + (1-(1-p_e)^L)\cdot(1-\alpha_r^{\mathbf{S}})\right], \quad (7)$$

where $\alpha_r^{\mathbf{D}}$ is the conditional probability that, provided P_r^{err} arrives, it has hidden errors with at least half of all corrupted packets. Let $p_{i,j}^{\mathbf{D}}(h)$ be the probability that two corrupted packets, respectively having k_i and k_j error bits ($i < j$), have a hidden error of DPC. Like $p_{i,j}^{\mathbf{S}}(h)$ determination, we obtain $p_{i,j}^{\mathbf{D}}(h) = p_{i,j}^{\mathbf{E}}(h)$. Recall that the occurrences of hidden errors in all corrupted packets are assumed to be independent events. As a result, we derive $\alpha_r^{\mathbf{D}}$ as

$$\alpha_r^{\mathbf{S}} = \frac{\sum_{k_1=1}^{L}\sum_{k_2=1}^{L}\cdots\sum_{k_{\lceil r/2\rceil}=1}^{L} p(k_1)p(k_2)\cdots p(k_{\lceil \frac{r}{2}\rceil})\cdot \prod_{1\le i,j\le\lceil \frac{r}{2}\rceil} p_{i,j}^{\mathbf{S}}(h)}{\left(1-(1-p_e)^L\right)^r \cdot \prod_{l=2}^{r-1}\alpha_l^{\mathbf{S}}}. \quad (8)$$

Taking $\alpha_r^{\mathbf{D}}$ in (9), we can derive $p_r^{\mathbf{D}}(s)$.

Suppose that the original packet can be recovered at the R-th round. Let $\Gamma^{\mathbf{E}}$, $\Gamma^{\mathbf{S}}$, and $\Gamma^{\mathbf{D}}$ denote the expected values of R of EARQ, SPaC-Merge, and DPC, respectively. According to the above successful probabilities of recovery, we can obtain $\Gamma^{\mathbf{E}}$, $\Gamma^{\mathbf{S}}$, and $\Gamma^{\mathbf{D}}$ using (9), (10), and (11), respectively.

$$\Gamma^{\mathbf{E}} = \sum_{k=1}^{R} k\cdot p_k^{\mathbf{E}}(s). \quad (9)$$

$$\Gamma^{\mathbf{S}} = \sum_{k=1}^{R} k\cdot p_k^{\mathbf{S}}(s). \quad (10)$$

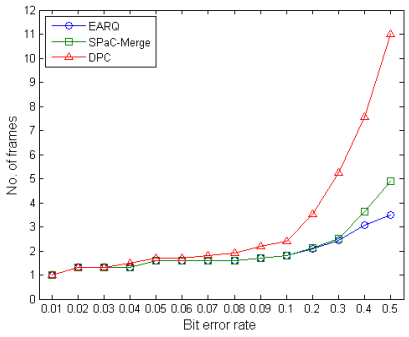
$$\Gamma^{\mathbf{D}} = \sum_{k=1}^{R} k\cdot p_k^{\mathbf{D}}(s). \quad (11)$$

IV. Simulation Results

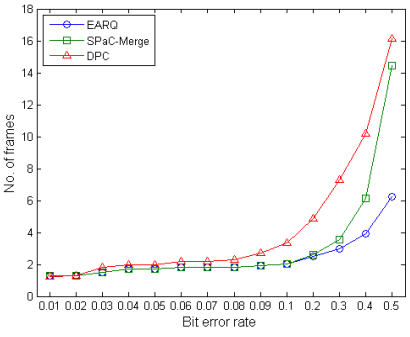
In this study, we use C++ to perform simulations, considering the binary symmetric channel model [11]. We use the BER to indicate the channel quality. The packet length includes 16-bit and 32-bit. The value of each data bit in the packet the transmitter sends is randomly determined. Simulation results were averaged over 20 runs. This study uses recovery speed and space overhead as simulation metrics to evaluate the recovery performance of different schemes.

A. Recovery Speed

In the simulation, the recovery speed indicates the total number of packets the receiver requires to obtain to recover the original packet. Fig. 1 shows the simulation results of recovery speed of different schemes. If increasing the packet length accompanies a high packet error probability, receivers require receiving more corrupted packets to recover the original packet. Although the packet error probability increases as the packet length increases, a good channel condition keeps the packet error probability low. Receivers can either easily receive the correct copy of the original packet or recover the original packet depending on a small number of corrupted packets. Therefore, the packet length causes a minor influence on recovery speed for three schemes. As the BER increases, the packet error probability becomes high. Receivers require more information of error bits carried in the received packets to recover the original packet.



(a)



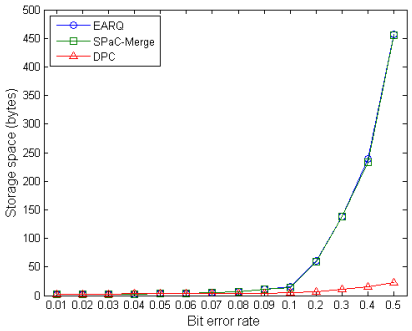
(b)

Figure 1. Simulation results of recovery speed. (a) L=16. (b) L=32

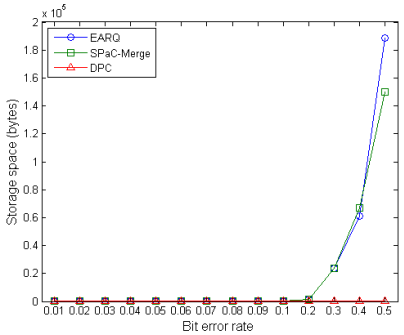
When the BER increases, the receiver is likely to require to receive more retransmitted packets (i.e., correct or erroneous copies of the original packet) to recover the original packet. Under a good channel condition, the numbers of required packets for three schemes slightly

increases. This is because the transmitted packet is more likely to be correct. These schemes approach an approximate recovery speed when the BERs do not exceed 0.1 for L =16 and 0.08 for L=32, as shown in Figs. 1(a) and 1(b), respectively.

Note that the probability of occurrence of hidden error increases with the increase of BER, thereby causing a speedy downgrade of recovery speed. This is significant in the DPC scheme, as the hidden error significantly dominates the recovery performance of DPC. Recall that the receiver in the EARQ possesses completed erroneous information of corrupted packets because it buffers all the corrupted copies of the original packet, and the SPaC-Merge scheme uses two corrupted packets only to recover the original packet. Compared with the EARQ, the SPaC-Merge has less information of error bits, thereby requiring a considerable number of corrupted packets to recover the original packet. Therefore, under a severe channel condition, the EARQ can outperform the SPaC-Merge and DPC in recovery speed.



(a)



(b)

Figure 2. Simulation results of space overhead. (a) L=16. (b) L=32

B. Space Overhead

This paper also investigates the space overhead of the EARQ, SPaC-Merge, and DPC schemes. This overhead is the total amount of storage space for buffering the received corrupted packets, XORed result, and all candidate packets. The space to buffer candidate packets is only required in the EARQ and SPaC-Merge. Specially, the space for storing candidate packets significantly dominates the overall storage space. Fig. 2 illustrates the overhead of storage space that the EARQ, SPaC-Merge, and DPC schemes require under different BERs when the packet lengths are 16 and 32 bits. In general, receivers require more corrupted packets to recover the original packet under a worse channel condition (i.e., lower BER). The results show that the DPC scheme outperforms the EARQ and SPaC-Merge schemes. We reason that the EARQ and SPaC-Merge schemes have to generate numbers of candidate packets using the XOR operation on the received corrupted packets, thereby rendering a considerable amount of space overhead compared with the DPC.

In general, the total amount of storage space for buffering the received corrupted packets, XORed result, and candidate packets increases with the increase of packet length. This can be obtained in Fig. 2. Recall that, to derive the candidate packets, the EARQ considers all the received corrupted packets while the SPaC-Merge scheme considers two corrupted packets only. In case of a low BER, the spaces for buffering received corrupted packets of EARQ and SPaC-Merge are approaching. However, if the BER increases, the EARQ requires additional space to store corrupted packets, thereby rendering a significant amount of space overhead, compared to the SPaC-Merge scheme.

V. Conclusion

In this paper, we have compared the performance of the EARQ, SPaC-Merge, and DPC, via analysis and simulations. As the analysis result shows, the hidden error and packet length significantly influence the recovery performance, such as the expected value of round number. This can be validated from our simulations. Simulation results also show that both recovery speed and space overhead degrade as the BER increases in three schemes. Moreover, the EARQ and SPaC-Merge schemes show a better recovery speed than the DPC scheme because they consider the received corrupted packets to generate candidate packets. On the other hand, the DPC scheme outperforms the EARQ and SPaC-Merge schemes in storage space overhead as it exploits a majority voting strategy on all the received corrupted packets and no candidate packets are derived.

Acknowledgment

This work is supported by the National Science Council of the Republic of China under Grant NSC101-2221-E-159-020.

References

[1] J. Ford, "Telecommunications with MEMS devices: An overview," *The 14th Annual Meeting of the IEEE Lasers and Electro-Optics Society*, vol. 2, no. 12, pp. 415-416, Nov. 2001.

[2] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," *IEEE Commun. Mag.*, vol. 40, no. 8, pp. 102-114, Aug. 2002.

[3] Y. Charfi, N. Wakamiya, and M. Murata, "Adaptive and reliable multi-path transmission in wireless sensor networks using forward error correction and feedback," in *Proc. IEEE Wireless Communications and Networking Conf.*, 2007, pp. 3684-3689.

[4] J. Chang and T. Yang, "End-to-end delay of an adaptive selective repeat ARQ protocol," *IEEE Trans. Commun.*, vol. 42, no. 11, pp. 2926-2928, Nov. 1994.

[5] M. Ghaderi, D. Towsley, and J. Kurose, "Reliability gain of network coding in lossy wireless networks," in *Proc. IEEE INFOCOM*, 2008, pp. 2171-2179.

[6] A. Miu, H. Balakrishnan, and C. E. Koksal, "Improving loss resilience with multi-radio diversity in wireless networks," in *Proc. ACM International Conf. Mobile Computing and Networking*, 2005, pp. 16-30.
[7] P. S. Sindhu, "Retransmission error control with memory," *IEEE/ACM Trans. Netw.*, vol. 25, no. 5, pp. 473-479, May 1977.
[8] S. S. Chakraborty, E. Yli-Juuti, and M. Liinaharja, "An ARQ scheme with packet combining," *IEEE Commun. Lett.*, vol. 2, no. 7, pp. 200-202, July 1998.
[9] H. Dubois-Ferrière, D. Estrin, and M. Vetterli, "Packet combining in sensor networks," in *Proc. 3rd International Conf. Embedded Networked Sensor Systems*, 2005, pp. 102-115.
[10] Z. Zhou, Z. Peng, J. H. Cui, and Z. Shi, "Efficient multipath communication for time-critical applications in underwater acoustic sensor networks," *IEEE/ACM Trans. Netw.*, vol. 19, no. 1, pp. 28-41, Feb. 2011.
[11] J. Hu, T. M. Duman, E. M. Kurtas, and M. F. Erde, "Bit-patterned media with written-in errors: Modeling, detection, and theoretical limits," *IEEE Trans. Magn.*, vol. 43, no. 8, pp. 3517-3524, Aug. 2007.

Kan-Yen Lin received the M.S. degree in Information Management from Minghsin University of Science and Technology, Taiwan, in 2011. From 2011, she worked as a software engineer at Aifu Company, Hsinchu, Taiwan. Her research interests include design of routing protocols for wireless sensor networks, and error recovery schemes for wireless networks.

Sheng-Shih Wang received the Ph.D. degree in computer science and information engineering from Tamkang University, Taiwan in 2006. In 2007, he joined the faculty of the Department of Information Management at Minghsin University of Science and Technology, Taiwan, and currently serves as an Assistant Professor. From 1999 to 2002, he worked as an R&D Engineer at Siemens Telecommunication Systems Limited, engaged in the design and development of software of switches. His current research interests include design of routing protocols and MAC protocols for wireless sensor networks, vehicular ad hoc networks, low power and lossy networks, and WiMAX systems. He is a member of the IEEE.

Pei-Chun Lee received the B.S., M.S., and Ph.D. degrees in Computer Science and Information Engineering from National Chiao Tung University, Taiwan, in 1998, 2000, and 2006, respectively. She joined the faculty of the Department of Information Management at Minghsin University of Science and Technology, Taiwan, and served as an Assistant Professor in August 2006. Dr. Lee's current research interests include design and analysis of wireless communications networks and network protocol design.