A Packet-Interleaving Scheme for Reliability Resilience under Burst Errors in Wireless Sensor Networks

Tsang-Ling Sheu and Yen-Hsi Kuo
Department of Electrical Engineering
National Sun Yat-Sen University
Kaohsiung, Taiwan
sheu@ee.nsysu.edu.tw ysk@atm.ee.nsysu.edu.tw

Abstract — **This paper presents a packet interleaving scheme (PIS) for increasing packet reliability under burst errors in wireless sensor networks (WSN). The proposed PIS, using Reed-Solomon (RS) codes, classifies data into two types: high-reliability-required (HRR) data and non-HRR data. An HRR packet is encoded with a short RS symbol, while a non-HRR packet with a long RS symbol. When an HRR and a non-HRR packet arrive at a sensor, they are interleaved on a symbol by symbol basis. Thus, the effect of burst errors (BE) is dispersed and consequently the uncorrectable HRR packets can be reduced. For the purpose of evaluation, two models, the uniform bit-error model (UBEM) and the on-off bit-error model (OBEM), are built to analyze the packet uncorrectable probability. In the evaluation, we first change the lengths of BE, then we vary the shift positions in a BE period, and finally we increase the number of correctable symbols to observe the superiority of the proposed PIS in reducing packet uncorrectable probability.**

Keywords: WSN, RS Code, burst errors, interleaving, packet uncorrectable probability

I. INTRODUCTION

Along with the increasing requirements for quality of living and home security, sensors have been widely deployed inside or outside a building to collect environmental information, such as temperature, humidity, image, motion picture, etc. To effectively deliver the collected data back to a control center for further analysis, a wireless sensor network (WSN) [1-3] is usually built. However, packet transmission over a WSN may encounter intermittent errors due to weak signals or interferences. The erroneous packets, if comprising of text or numbers, such as temperature or humidity, would require packet re-transmission, which increases network load. Thus, the motivation of this paper is to increase transmission reliability over a WSN, which has recently attracted many researchers' attention.

Basically, previous researches on transmission reliability over a WSN can be divided into two major categories: reliable routing and information coding. In the first category, to increase the transmission reliability after data are collected by a sensor node, relay nodes (RNs) are employed. For examples, H. Chebbo, et al. [4] modified IEEE 802.15.4 MAC frames. The authors added one bit in the frame control field, with which whether it is necessary to build a tree by RNs or just build a simple star, can be determined. Moreover, R. Sampangi, et al. [5] utilized RNs to divide sensors into several cluster networks. Since the distance from a sensor to its cluster head is reduced, the quality of data transmission is greatly improved. To protect the routing path, S. Kim, et al. [6] utilized both coding and retransmission schemes once the established path fails. However, in these schemes, it is inevitable that end-to-end packet delay will increase accordingly due to multiple-hop forwarding.

Thus, in the second category, instead of developing reliable routing, the authors switch their interests to information coding. For examples, E. Byrne, et al. [7] designed a coding scheme which can increase the probability of successful decoding based on graph theory. Y. Hamada, et al. [8] proposed a scheme to reduce packet error rate by using Luby Transform (LT) codes [9]. Their proposed scheme has achieved small complexity of O(n); yet too many packets require retransmission when bit error rate is high. Thus, K. Ishibashi, et al. [10] proposed an embedded forward error control (FEC) technique which utilizes RS (Reed Solomon) code to reduce packet error rate. Similarly, M. Busse, et al. [11] can recover lost chunks by using Fountain code and Raptor code. To increase data reliability and processing speed, K. Yu, et al. [12] designed a new FEC which protects header and payload, respectively. Similarly, M. Srouji, et al. [13] proposed a reliable data transfer scheme which can adjust the lengths of redundancy code based on the successful receiving rate at the downstream node.

Unlike the previous research work, in this paper we propose a packet interleaving scheme (PIS) to reduce the impact of burst errors (BE) on high-reliability-required (HRR) data in WSNs. Although Reed-Solomon (RS) codes may correct bit errors under certain constraints, it may not be economically worthy in dealing with burst errors when the number of consecutive bit errors exceeds a threshold. Hence, to increase packet correctable probability in a WSN, the proposed PIS first classifies the collected data into two different types: HRR data and non-HRR data. An HRR packet is encoded with a short RS symbol, while a non-HRR packet with a long RS symbol. When an HRR and a non-HRR packet arrive at a sensor, they are interleaved on a symbol-by-symbol basis. The noticeable benefit from the packet interleaving is that the burst errors are dispersed and the uncorrectable probability of HRR packets is significantly reduced.

The remainder of this paper is organized as follows. In Section 2, the proposed PIS and its operations are described. In Section 3, an analytical model is built using two bit-error models, the uniform bit-error model (UBEM) and the on-off bit-error model (OBEM). In Section 4, numerical results are presented and discussed. Finally, conclusions are drawn in Section 5.

II. PACKET INTERLEAVING SCHEME

A. *WSN with Multi-hop Tree Structure*

In a wireless sensor network (WSN), a coordinator is the sink which gathers all the data collected from other distant sensor nodes. To facilitate data gathering from all the sensor nodes, it is very constructive that the coordinator and the sensor nodes will collaborate to build a multi-hop tree structure (MTS), as shown in Figure 1. In an MTS-based WSN, data collected by a sensor node will be forwarded hop-by-hop to the coordinator. Thus, by fully utilizing a branch node of the MTS, in this paper, we propose a packet interleaving scheme (PIS) based on RS codes to reduce the impact of burst errors on packet uncorrectable probability.

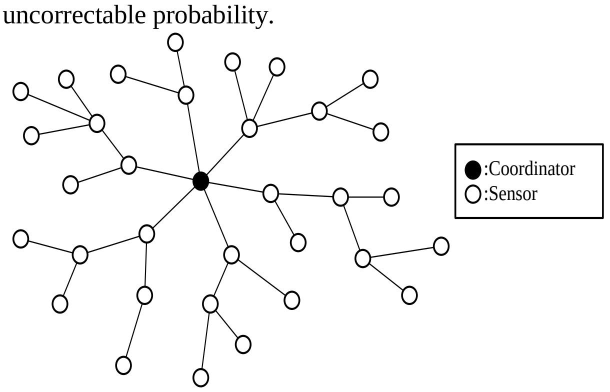


Figure 1. WSN with multi-hop tree structure

B. *Packet Interleaving*

In the proposed PIS, packets collected by a sensor node are classified into two different types: high-reliability required (HRR) packet and non-HRR packet. An HRR packet is defined as a packet which requires for retransmission, if uncorrectable burst errors exist. Payload in an HRR packet consists of numerical data, such as temperature, moisture, luminance, etc. This type of packet has relatively shorter data length (usually, a couple of bytes) and each uncorrectable HRR packet requires for retransmission. Hence, it is better to employ a shorter-length symbol (in this paper, we use m = 4) to encode an HRR packet with shorter data length. On the other hand, payload in a non-HRR packet consists of non-numerical data, such as video, audio, etc. This type of packet has relatively longer data length (usually, in the order of kilo bytes) and each uncorrectable non-HRR packet may not require retransmission. Thus, it is better to employ longer-length symbol (we use *m* = 8) to encode a non-HRR packet with longer data length.

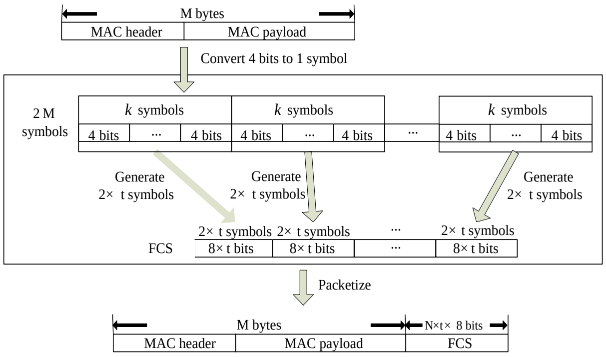


Figure 2. An HRR encoded with *m* = 4

As it is illustrated in Figure 2, an HRR packet is encoded with a shorter-length RS symbol (i.e., m = 4). First, an M-byte MAC-layer header and payload is converted to $\frac{(M\times 8)}{4}=2M$ symbols on the basis of 4 bits per symbol. Thus, we have the length of a codeword is n, where $n=2^m-1=2^4-1=15$, the number of symbols for user data in a codeword is k, where $k=n-2\times t=15-2\times t$, and the number of symbols for redundancy code in a codeword is $2\times t$. Let $N=\left\lceil \frac{2M}{k}\right\rceil$, which denotes the number of codeword required for encoding the M-byte packet (header plus payload). The total redundancy code (or FCS) is therefore equal to $N\times 2\times t$ symbols, or $N\times 2\times t\times 4\,(=N\times t\times 8)$ bits.

Similarly, as it is illustrated in Figure 3, a non-HRR packet is encoded with a longer-length symbol (m = 8). An M-byte MAC-layer header and payload is converted to $\frac{(M\times 8)}{8}=M$ symbols on the basis of 8 bits per symbol. A code-word length, $n=2^m-1=2^8-1=255$ bytes, is much greater than the maximum length of a packet (127 bytes in WSN). Thus, a code-word is sufficiently enough to encode a MAC-layer packet. Thus, the length of redundancy code (or FCS) is equal to $2\times t$ bytes.

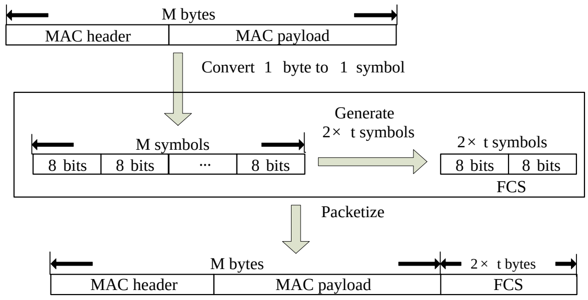


Figure 3. A non-HRR encoded with *m* = 8

When both an HRR and a non-HRR packet are received by a branch node in a tree-structured WSN, these two packets are interleaved on a symbol-by-symbol basis, as shown in Figure 4. The interleaved packet is then forwarded to an upper stream node, which performs decoding and correction process. However, as shown in Figure 5, an interleaved packet may not be correctable, if it encounters burst errors where the number of total errors is greater than t. In the proposed PIS, by separating the interleaved single packet back to their original two packets, each individual packet may become correctable. This is because the number of errors in each separated packet is highly possible to be smaller than t.

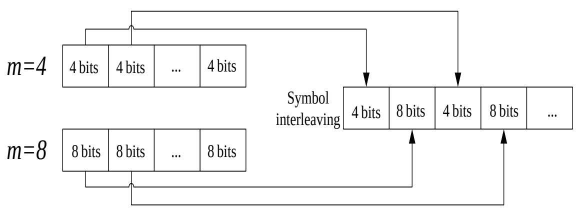


Figure 4. Packet interleaving on a symbol-by-symbol basis

 ISBN: 978-1-61208-450-3

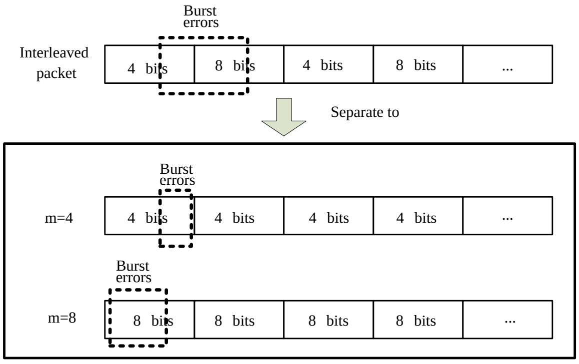


Figure 5. Burst errors dispersed on two symbols

III. ANALYTICAL MODEL

Two analytical models are built for comprehensive numerical simulations. The first one is referred to as uniform bit error model (UBEM), while the second one is referred to as on-off bit error model (OBEM). The first model assumes the errors occur evenly on the coded packets, while the second model assumes the errors may occur continuously in a burst length.

A. UBEM

Let P_{UB_be} and P_{UB_se} denote the probability of bit errors and the probability of symbol errors, respectively. Since any bit errors occur in a symbol may result in a symbol error and each symbol has m bits, we can derive P_{UB_se} directly from P_{UB_be}, as shown in Eq. (1).

$$P_{UB\_se} = 1-(1-P_{UB\_be})^m \tag{1}$$

Next, let us define two more parameters, N_S and N_C. The first parameter denotes the number of symbols in a codeword and the second parameter denotes the number of codeword in a packet. Hence, the uncorrectable probability of a codeword (P_{UB_cuc}), can be derived as shown in Eq. (2).

$$P_{UB\_cuc} = 1-\sum_{i=0}^{t}\left(\binom{N_s}{i}\times\left(P_{UB\_se}\right)^i\times\left(1-P_{UB\_se}\right)^{N_s-i}\right) \tag{2}$$

In Eq. (2), we know in a codeword if the number of symbol errors is smaller than t, then the codeword is correctable. Thus, the uncorrectable probability of a codeword can be summed up from $i=0$ to t, since there are $\binom{N_s}{i}$ different types of errors. Next, let us define P_{UB_puc} as the packet uncorrectable probability. Since there are N_C codeword in a packet, we can derive P_{UB_puc} as shown in Eq. (3).

$$P_{UB\_puc} = 1-\left(1-P_{UB\_cuc}\right)^{N_C} \tag{3}$$

B. OBEM

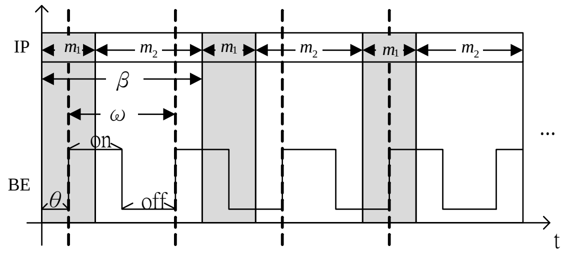


Figure 6. On-off bit error model

The on-off bit error model (OBEM) is illustrated in Figure 6. All the parameters used in the analysis are defined in Table I. A burst error (BE) period is defined as two consecutive bit error intervals where high bit errors appear first and then followed by low bit errors. Notice that θ_0 is defined as the length of right-shift position for an initial BE period; $\theta_0 = 0$ implies that no gap exists between the beginning of an interleaved packet and the beginning of the first BE period.

TABLE I. PARAMETERS USED IN OBEM

m_1	Number of bits in a symbol of HRR packet
m_2	Number of bits in a symbol of non-HRR packet
β	Packet interleaved length in bits ($m_1 + m_2$)
on	Length of high bit errors (in bits)
σ	Error probability of high bit errors
off	Length of low bit errors (in bits)
ρ	Error probability of low bit errors
ω	Burst length (*on* + *off*) (in bits)
θ_0	Length of initial right-shift position (in bits)

First, we define P_{OB_se} as the symbol-error probability in OBEM. An interleaved packet (IP) and a burst error (BE) may have different lengths; here we assume the former has a length of β bits ($\beta = m_1 + m_2$) and the later has a length of ω bits ($\omega = on + off$). Since every symbol in an IP may encounter different positions of bit errors, we have to analyze the bit error positions of a symbol before we can compute the symbol-error probability. To compute the error probability of the α^{th} symbol, we define (i) m_{1s} = the distance between the first bit of m_1 and the first bit of an IP, and (ii) m_{1e} = the distance between the last bit of m_1 and the first bit of an IP. Similarly, we define m_{2s} and m_{2e} for m_2. Thus, m_{1s}, m_{1e}, m_{2s}, and m_{2e} can be computed as shown in Eq. (4), (5), (6), and (7), respectively.

$$m_{1s} = \alpha \times \beta \tag{4}$$

$$m_{1e} = \alpha \times \beta + m_1 - 1 \tag{5}$$

$$m_{2s} = \alpha \times \beta + m_1 \tag{6}$$

$$m_{2e} = (\alpha + 1) \times \beta - 1 \tag{7}$$

For simplicity, the four parameters, m_{1s}, m_{1e}, m_{2s}, and m_{2e} are generalized to m_{ij}, where $i = 1, 2$ and

$j = s, e$. Let θ denote the length of right-shift position between an IP and a BE at the α^{th} symbol. Let $\Omega_{m_{ij}}$ denote the length of right-shift position for m_{1s}, m_{1e}, m_{2s}, and m_{2e} at the α^{th} symbol. Thus, we can compute θ and $\Omega_{m_{ij}}$ as shown in Eq. (8) and (9), respectively.

$$\theta = \left(\theta_0 + \left(\left\lceil \frac{header\ length}{\omega} \right\rceil \times \omega - header\ length\right)\right) mod\ \omega \quad (8)$$

$$\Omega_{m_{ij}} = \left(\left\lfloor \frac{m_{ij}+1}{\omega} \right\rfloor - 1\right) \times \omega + \theta \quad (9)$$

After we found the right-shift position between an IP and a BE, we can categorize the symbol errors into four cases. Case 1 shows whether or not a symbol may occupy one BE or two BE periods, their start bit and stop bit of a symbol all appear at the high-bit-error interval. Case 2 shows only the start bit of a symbol appears at the high-bit-error interval. Case 3 shows whether or not a symbol may occupy one BE or two BE periods, neither the start bit nor the stop bit appear at the high-bit-error interval. Finally, Case 4 shows only the stop bit of a symbol appears at the high-bit-error interval. Actually, the four different cases of symbol errors can be constrained by eight inequalities with four parameters, m_{ij}, $\Omega_{m_{ij}}$, on, and ω, as shown in Table II.

TABLE II. FOUR CASES OF SYMBOL ERRORS

Case	Conditions
1	$\Omega_{m_{is}} \le m_{is} < \Omega_{m_{is}} + on$ and $\Omega_{m_{ie}} \le m_{ie} < \Omega_{m_{ie}} + on$
2	$\Omega_{m_{is}} \le m_{is} < \Omega_{m_{is}} + on$ and $\Omega_{m_{ie}} + on \le m_{ie} < \Omega_{m_{ie}} + \omega$
3	$\Omega_{m_{is}} + on \le m_{is} < \Omega_{m_{is}} + \omega$ and $\Omega_{m_{ie}} + on \le m_{ie} < \Omega_{m_{ie}} + \omega$
4	$\Omega_{m_{is}} + on \le m_{is} < \Omega_{m_{is}} + \omega$ and $\Omega_{m_{ie}} \le m_{ie} < \Omega_{m_{ie}} + on$

Once we identify the four cases of symbol errors, we can compute the symbol-error probability. Since one codeword consists of N_s symbols, we define $P_{OB\_se}(\alpha), \alpha = 0,1,..., N_s - 1$, as the symbol-error probability of the α^{th} symbol. Let σ denote the error probability of high bit errors and ρ denote the error probability of low bit errors. Let N_σ represent the number of bits with errors in a symbol and N_ρ represent the number of bits without errors in a symbol. $P_{OB\_se}(\alpha)$ is computed as shown in Eq. (10).

$$P_{OB\_se}(\alpha) = 1 - (1-\sigma)^{N_\sigma} \times (1-\rho)^{N_\rho} \quad (10)$$

Now, we can compute N_σ and N_ρ for case 1 as shown in Eq. (11) and (12), case 2 as shown in Eq. (13) and (14), case 3 as shown in Eq. (15) and (16), and case 4 as shown in Eq. (17) and (18), respectively.

$$N_\rho = \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - 1\right) - \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times Off$$
$$= \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor\right) \times Off \quad (11)$$

$$N_\sigma = m_i - N_\rho = m_i - \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor\right) \times Off \quad (12)$$

$$N_\sigma = \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - 1\right) - \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times On + \Omega_{m_{is}} + On - m_{is}$$
$$= \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor\right) \times On + \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times \omega + \theta + On - m_{is}$$
$$= \left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor \times On + \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times Off + \theta - m_{is} \quad (13)$$

$$N_\rho = m_i - N_\sigma$$
$$= m_i - \left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor \times On - \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times Off - \theta + m_{is} \quad (14)$$

$$N_\sigma = \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - 1\right) - \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times On$$
$$= \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor\right) \times On \quad (15)$$

$$N_\rho = m_i - N_\sigma = m_i - \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor\right) \times On \quad (16)$$

$$N_\rho = \left(\left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - 1\right) - \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) - 1\right) \times Off + \Omega_{m_{is}} + \omega - m_{is}$$
$$= \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times Off + \left(\left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor - 1\right) \times \omega + \theta + \omega - m_{is}$$
$$= \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor \times On + \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - 1\right) \times Off + \theta - m_{is} \quad (17)$$

$$N_\sigma = m_i - N_\rho = m_i - \left\lfloor \frac{m_{is}+1}{\omega} \right\rfloor \times On - \left(\left\lfloor \frac{m_{ie}+1}{\omega} \right\rfloor - 1\right) \times Off - \theta + m_{is} \quad (18)$$

By substituting N_σ and N_ρ back to Eq. (10), we can derive $P_{OB\_se}(\alpha)$ for the four different cases of symbol errors. After we compute the symbol-error probability for the four different cases, our next step is to derive the uncorrectable probability of a codeword and the uncorrectable probability of a packet. We know there are N_C codeword in a packet and the uncorrectable probabilities of the N_C codeword are all different. Let $P_{OB\_cuc}(l), l = 0,1,..., N_c - 1$, denote the uncorrectable probability of the l^{th} codeword and let P_{OB_puc}

denote the uncorrectable probability of a packet. By using the combination theory of probabilities, we can derive $P_{OB\_cuc}(l)$ and P_{OB_puc} as shown in Eq. (19) and (20).

$$P_{OB\_cuc}(l) = 1 - \sum_{i=0}^{t}\left(\sum_{j=0}^{\tau_i-1}\left(\prod_{k=0}^{\lambda-1}\left(\binom{N_k}{r_k^{ij}}\times\left(P_{OB\_se}^{k}\right)^{r_k^{ij}}\times\left(1-P_{OB\_se}^{k}\right)^{N_k-r_k^{ij}}\right)\right)\right) \tag{19}$$

$$P_{OB\_puc} = 1-\prod_{l=0}^{N_C-1}\left(1-P_{OB\_cuc}(l)\right) \tag{20}$$

IV. NUMERICAL RESULTS

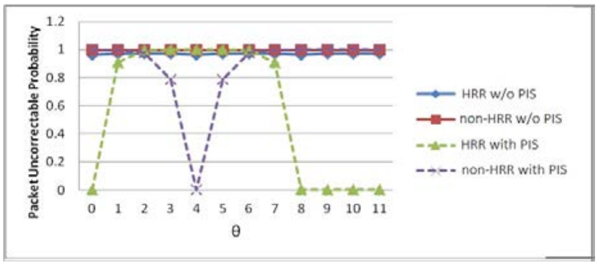
To study the influences of the four parameters, (i) the length of high bit errors (the on period), (ii) the length of low bit errors (the off period), (iii) the right-shift position (θ), and (iv) the number of correctable symbols (t), we perform numerical simulations. Table III shows the parameters and their setting used in the simulation.

TABLE III. PARAMETER SETTINGS

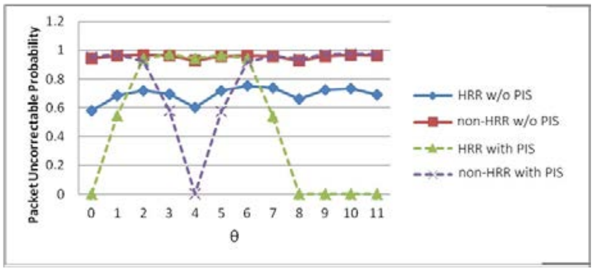
Parameter	Setting
m_1	4 bits
m_2	8 bits
on	4/6/8 bits
off	8/6/4 bits
σ	0.1
ρ	0.0001

A. Impact of Correctable Symbols

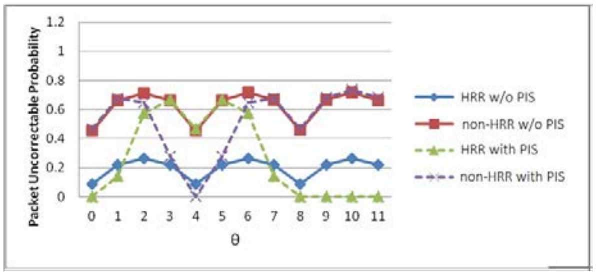
First, we are interested in studying the impact of increasing the number of correctable symbols when the length of high bit errors is shorter than the length of low bit errors; i.e., on = 4 bits and off = 8 bits. From Figure 7, we can observe that both the packet uncorrectable probabilities of HRR and non-HRR curves drop off very quickly, when the number of correctable symbols (t) is increased from 1 to 3. Additionally, we can observe that the curves of P_{OB_puc} with PIS (the two dashed lines) are much lower than the curves of P_{OB_puc} without using PIS (the two solid lines). The improvement of P_{OB_puc} by using PIS is more significant when t is small, which is quite beneficial for reducing packet overhead, since the number of redundancy bits in RS codes can be shorter. Another noticeable phenomenon is that although when θ is smaller than 7, HRR with PIS are completely inverted to non-HRR with PIS, the curves of HRR with PIS do drop to zero when θ is larger than 7.



(a) (on, off, t) = (4, 8, 1)



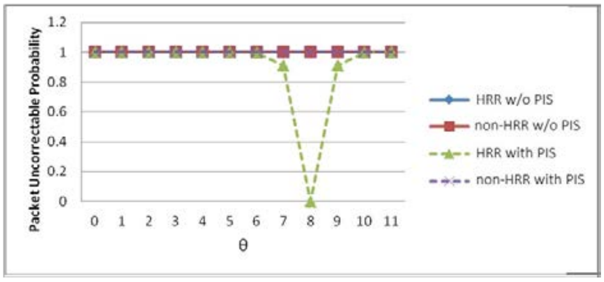
(b) (on, off, t) = (4, 8, 2)



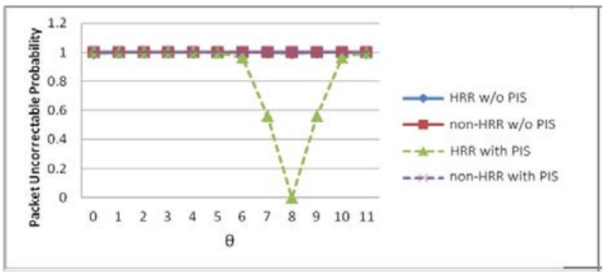
(c) (on, off, t) = (4, 8, 3)

Figure 7. Packet uncorrectable probability (on is smaller than off)

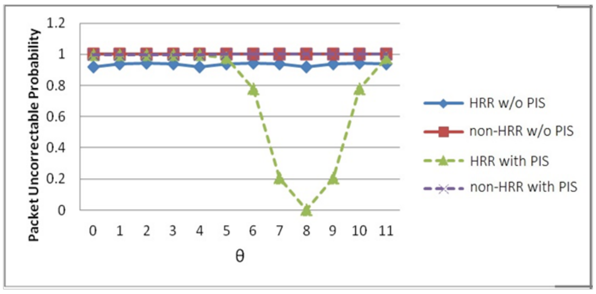
From Figures 8(a) to 8(c), we show the packet uncorrectable probabilities when the on period (8 bits) is longer than the off period (4 bits). It is interesting to notice that when t is larger than 3 and θ is smaller than 6, the packet uncorrectable probabilities of HRR with PIS are higher than those curves without PIS. Of course, when t is smaller than 3 and θ is larger than 6, the situations are completely inverted. Hence, from Figure 11, we have discovered that when the period of high bit errors exceeds the length of an HRR symbol (m = 4) and approaches to a non-HRR symbol (m = 8), it is better to encode packets with a small value of t; otherwise, there is no advantage achieved by using the proposed PIS.



(a) (on, off, t) = (8, 4, 1)



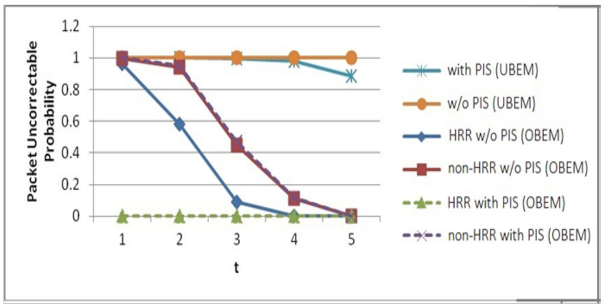
(b) (on, off, t) = (8, 4, 2)



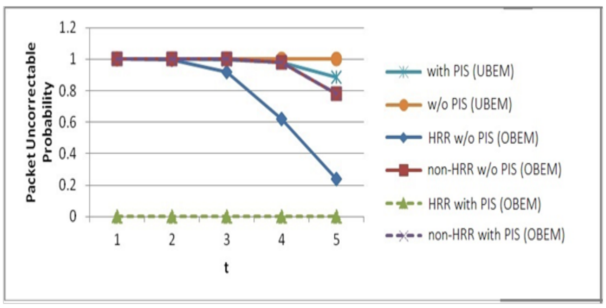
(c) (on, off, *t*) = (8, 4, 3)

Figure 8. Packet uncorrectable probability (on is larger than off)

Figures 9 shows the comparisons in packet uncorrectable probabilities between UBEM and OBEM. Notice that the curves of UBEM and the curves of OBEM vary along with the following two parameters: (i) when the number of correctable symbols increases from 1 to 5, the curves of UBEM dissever very quickly from those of OBEM; and (ii) when the right-shift position increases from zero to 8 bits, the gap between these two models becomes smaller. Since OBEM is more reactive to a real word than UBEM, it is rewarding to know that the proposed PIS can reduce packet uncorrectable probability in OBEM more significantly than that in UBEM. Another noticeable result is that no matter how we increase the period of high bit errors (the on period from 4 bits in 9(a) to 8 bits in 9(b)), HRR with PIS in OBEM always exhibits the lowest packet uncorrectable probability (near zero, in some cases). The relatively lower packet uncorrectable probability for HRR packets has demonstrated that the proposed PIS can successfully protect HRR packets from burst errors, while at the same time it does not sacrifice non-HRR packets from large uncorrectable bit errors.



(a) (on, off, θ) = (4, 8, 0)



(b) (on, off, θ) = (8, 4, 8)

Figure 9. Packet uncorrectable probability in UBEM vs in OBEM

V. CONCLUSIONS

In this paper, we have presented a packet interleaving scheme to reduce packet uncorrectable probability under burst errors in WSN. From the simulations, we have demonstrated that, no matter how we adjust the period of high bit errors, the proposed PIS behaves more resilient to burst errors in OBEM than in UBEM. Finally, by carefully adjusting the period of high bit errors and the right-shift positions, the PIS can reduce the uncorrectable probability of HRR packets to near zero.

REFERENCES

[1] F. Barac, K. Yu, M. Gidlund, J. Akerberg, and M. Bjrkman, “Towards Reliable and Lightweight Communication in Industrial Wireless Sensor Networks,” INDIN2012: IEEE 10th International Conference on Industrial Informatics, Beijing, China, Jul. 25-27, 2012.

[2] S. Marinkovic and E. Popovici, “Network Coding for Efficient Error Recovery in Wireless Sensor Networks for Medical Applications,” 2009 First Int’l Emerging Network Intelligence, Sliema, Malta, Oct. 11-16, 2009.

[3] B. Sklar, Digital Communications: Fundamentals and Applications, 2nd Ed. New Jersey, USA, 2001.

[4] H. Chebbo, S. Abedi, T. A. Lamahewa, D. B. Smith, D. Miniutti, and L. Hanlen, “Reliable Body Area Networks Using Relays: Restricted Tree Topology,” Int’l Conference on Networking and Communications, Maui, Hawaiian Island, USA, Jan. 30 - Feb. 2, 2010.

[5] R. V. Sampangi, S. R. Urs, and S. Sampalli, “A Novel Reliability Scheme Employing Multiple Sink Nodes for Wireless Body Area Networks,” 2011 IEEE Symposium on Wireless Technology and Applications (ISWTA), Langkawi, Malaysia, Sep. 25-28, 2011.

[6] S. Kim, R. Fonseca, and D. Culler, “Reliable Transfer on Wireless Sensor Network,” First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks, Santa Clara, California, Oct. 4-7, 2004.

[7] E. Byrne, A. Manada, S. Marinkovic, and E. Popovici, “A Graph Theoretical Approach for Network Coding in Wireless Body Area Networks,” 2011 IEEE International Symposium on Information Theory Proceedings (ISIT), Saint Petersburg, Russia, Jul. 31 - Aug. 5, 2011.

[8] Y. Hamada, K. Takizawa, and T. Ikegami, “Highly Reliable Wireless Body Area Network using Error Correcting Codes,” 2012 IEEE Radio and Wireless Symposium, Santa Clara, CA, USA, Jan. 15-18, 2012.

[9] M. Luby, “LT Codes,” The 43rd Annual IEEE Symposium on Foundations of Computer Science, Vancouver, British Columbia, Canada, Nov. 16-19 2002.

[10] K. Ishibashi, H. Ochiai, and R. Kohno, “Embedded Forward Error Control Technique for Low-Rate but Low Latency Communications,” IEEE Trans. on Wireless Comm., vol. 7, no. 5, pp. 1456-1460, May 2008.

[11] M. Busse, T. Haenselmann, and W. Effelsberg, “Energy-Efficient Data Dissemination for Wireless Sensor Networks,” Fifth IEEE Int’l Conference on Pervasive Computing and Communications Workshops, White Plains, New York, USA, Mar. 19-23, 2007.

[12] K. Yu, F. Barac, M. Gidlund, J. Akerberg, and M. Bjorkman, “A Flexible Error Correction Scheme for IEEE 802.15.4-based Industrial Wireless Sensor Networks,” 2012 IEEE Int’l Symposium on Industrial Electronics (ISIE), Hangzhou, China, May 28-31, 2012.

[13] M. S. Srouji, Z. Wang, and J. Henkel, “RDTS: A Reliable Erasure-Coding Based Data Transfer Scheme for Wireless Sensor Networks,” 2011 IEEE 17th International Conference on Parallel and Distributed Systems, Tainan, Taiwan, Dec. 7-9, 2011.

 ISBN: 978-1-61208-450-3