Packet Aggregation Based on Encryption-Then-Compression for Highly Efficient Multi-Hop Transmission

Ryota Yatsu*, Takanori Hara*, Koji Ishibashi*, Sota Tsuchiya†, and Hideki Endo†
*Advanced Wireless & Communication Research Center (AWCC),
The University of Electro-Communications
1-5-1 Chofugaoka, Chofu-shi, Tokyo 182-8585, Japan
Email: {yatsu, hara}@awcc.uec.ac.jp, koji@ieee.org
†Fundamental Technology Dept., Tokyo Gas Co., Ltd.
1-7-7 Suehiro-cho, Tsurumi-ku, Yokohama, Kanagawa 230-0045, Japan
Email: {stsuchiya, endou.hideki}@tokyo-gas.co.jp

***Abstract*—In this paper, we tackle a critical bottleneck problem appeared in multi-hop transmissions; only a single node can forward packets from the other nodes in the network to a common destination. Since every node encrypts data before transmission for privacy protection, typical compression techniques cannot be applied to the aggregated packet. We hence propose the packet aggregation based on *encryption-then-compression* (EtC) technique enabling the efficient compression of the encrypted data. Assuming *intermittent receiver-driven data transmission* (IRDT) as a multi-hop transmission protocol, numerical results show that our proposed method can achieve the lower decompression error probability than the conventional EtC and significantly reduce the energy consumption per a correctly received packet.**

I. Introduction

Internet-of-Things (IoT) connecting devices via wireless communications would enhance user experiences and reduce the daily maintenance cost of services. Smart-meters are one good example of IoT, which sends customer information, usage data, status logs of meters, and more to a service center. In practice, those meters would be installed in every customer's house, and some of them might be placed in unfavorable places for wireless communications such as a metal plumbing box or the space between houses. Increasing the transmission power for the stable connection is an option to overcome this problem, but it results in the even shorter life of the battery-driven smart-meter because of the exponential nature of wireless path-loss. Multi-hop transmissions hence are a judicious option doubtlessly, considering the scalability of services without increasing power consumption [1] [2]. Note that this is also true for general wireless sensor networks (WSNs) requiring low power consumption of sensor nodes.

Receiver-initiated medium access control (RI-MAC) protocol and its improvements such as *intermittent receiver-driven data transmission* (IRDT) [3] are known as energy-efficient MAC protocols with multi-hop capability, and smart-meters in Japan and IEEE 802.15.4e also employ variations of RI-MAC [4]. The performance of those protocols depends on the duty cycle and the density of nodes in the network [5]. Therefore, if only a single node can forward packets from the other nodes in the network to a common destination, the performance of the system critically relies on the node called *bottleneck node* while the energy consumption of the node increases remarkably, which is called a *bottleneck problem*.

In order to alleviate this burden, *packet aggregation* has been proposed so as to reduce the number of forwarding processes and the size of corresponding overheads by buffering all the packets to be forwarded and forwarding them at once [6]. However, this aggregation only reduces the overheads but the size of the resulting payload which increases in proportion to the number of the aggregated packets. As an alternative, a compression scheme exploiting the correlation of packet-headers has been proposed in [7]. However, the gain of this compression is generally limited since the header part is much shorter than the payload of the packet. Thus, the compression of the payload, namely data part, is an only remedy to reduce the burden to the bottleneck node.

As mentioned above, the payload of smart-meters includes the area information, customer information, daily activities, usage data, status log, and so on. These data do not change sharply in time and also do not change so much among neighboring meters. Hence the correlation among the packets hence can be exploited to reduce the size. However, for privacy protection, the packet must be encrypted and cannot be decrypted at the intermediate node namely relay node as obvious, so that the entropy of the encrypted binary sequence is maximized, and the sequence cannot be compressed as clear from information-theoretical results [8]. To cope with this fundamental challenge, *encryption-then-compression* (EtC) methods have been proposed and studied in [9] [10].

In the method of [9], any intermediate nodes can compress encrypted binary sequences using error-correcting codes without decryption. The destination node decompresses the received compressed-sequences based on the Slepian-Wolf theorem [11] that utilizes the correlation between the encrypted sequences and the encryption key. However, this method is not

applicable to the packets encrypted with block ciphers, which are frequently used in Transport Layer Security/Secure Sockets Layer (TLS/SSL). To address this problem, the EtC method for block ciphers operating in the cipher block chaining (CBC) mode has been proposed [10]. This method divides the encrypted sequences into subsequences whose length is equivalent to that of the encryption key and then compresses them using error-correcting codes. However, its performance degrades when the system employs block ciphers using a short-length encryption key such as Data Encryption Standard (DES) [12] and Advanced Encryption Standard (AES) [13]. This problem arises from the fact that the compression performance depends on the decoding performance of error-correcting codes of which performance depends on the length of the codewords.

In the light of the above, this paper proposes a new packet aggregation method in combination with EtC for multi-hop systems employing block ciphers with a short-length key. Our proposed method can effectively compress the aggregated packet via EtC and achieve the superior decompression performance. Numerical results confirm our proposed method achieves a lower decompressing error probability than the conventional method proposed in [10]. Assuming IRDT as an example, we further demonstrate that our proposed system can decrease packet collision probability and the average energy consumption of the entire network per packet as compared to the conventional system without the EtC.

Throughout the paper, we use the following notation. The transpose, the exclusive OR, and the vectorize operators are represented as $\cdot^{\mathrm{T}}$, $\oplus$, and $\mathrm{Vec}(\cdot)$ respectively. Moreover, $\mathbb{N}$ and $\mathbb{R}^{+}$ denote the set of natural numbers and the set of positive real numbers, respectively.

II. System Model

A. System Model

In this paper, we assume a network model composed of M source nodes (SNs), a relay node (RN), and a common destination node (DN). For the sake of simplicity, we suppose that all SNs and DN can communicate with only RN. Therefore, RN is the bottleneck node as described above.

Fig. 1 shows a block diagram of the system. The m-th SN, $\mathrm{SN}_m, m = 1, \ldots, M$ converts its own data into a binary information packet $\mathbf{X}^{(m)} = (X_1^{(m)}, \ldots, X_l^{(m)}, \ldots, X_L^{(m)})^{\mathrm{T}} \in \{0,1\}^{L\times 1}$ where its occurrence probability of the bit "1" is $p \in [0,1]$, and every bit follows the identically independent distribution, i.e., $\Pr(X_l^{(m)} = 1) = p$ and $\Pr(X_l^{(m)} = 0) = 1 - p$. Upon the generation of the packet, SN_m encrypts the packet $\mathbf{X}^{(m)}$ with a block cipher in CBC mode. Let $\mathbf{K}^{(m)} \in \{0,1\}^{K\times 1}$ be the unique encryption key of length K, and $\mathbf{y}_0^{(m)}, \ldots, \mathbf{y}_N^{(m)} \in \{0,1\}^{K\times 1}$ be the $(N+1)$ cipher blocks of which length is K where $N \triangleq K/L$, $N \in \mathbb{N}$, and $\mathbf{y}_0^{(m)}$ denotes the given initial cipher block for SN_m. The N information blocks, $\mathbf{x}_n^{(m)} \in \{0,1\}^{K\times 1}, n = 1, \ldots, N$, is given

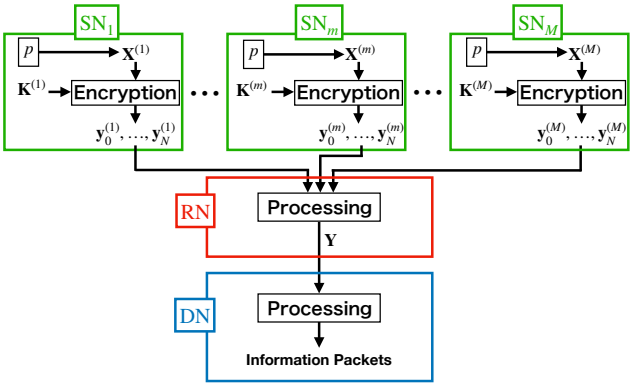

Fig. 1. Block diagram of the system.

by dividing $\mathbf{X}^{(m)}$ by the equal block length K. Namely,

$$\mathbf{x}_n^{(m)} = (x_{n,1}^{(m)}, \ldots, x_{n,K}^{(m)})^{\mathrm{T}} = \left(X_{(n-1)K+1}^{(m)}, \ldots, X_{nK}^{(m)}\right)^{\mathrm{T}}. \tag{1}$$

Let an arbitrary block cipher execution function using $\mathbf{K}^{(m)}$ denote $\mathcal{F}_{\mathbf{K}^{(m)}}[\cdot] : \{0,1\}^{K\times 1} \to \{0,1\}^{K\times 1}$. Then, in CBC mode, the input to the block cipher $\hat{\mathbf{x}}_n^{(m)} = (\hat{x}_{n,1}^{(m)}, \ldots, \hat{x}_{n,K}^{(m)}) \in \{0,1\}^{K\times 1}$ and output $\mathbf{y}_n^{(m)} = (y_{n,1}^{(m)}, \ldots, y_{n,K}^{(m)})$ can be expressed, respectively, as

$$\hat{x}_{n,k}^{(m)} = x_{n,k}^{(m)} \oplus y_{n-1,k}^{(m)}, \quad (k = 1, \ldots, K), \tag{2}$$

$$\mathbf{y}_n^{(m)} = \mathcal{F}_{\mathbf{K}^{(m)}}\left[\hat{\mathbf{x}}_n^{(m)}\right], \quad (n = 1, \ldots, N). \tag{3}$$

After the encryption, SN_m sends all cipher blocks $\mathbf{y}_0^{(m)}, \ldots, \mathbf{y}_N^{(m)}$ to RN.

After RN receives all the cipher blocks, it executes an arbitrary compression process that will be explained in detail in Section III so as to obtain the shortened version of the aggregated packet. The resulting sequence is $\mathbf{Y} \in \{0,1\}^{R'NK}$ where $R' \in \mathbb{R}^{+}$ and $R'NK \in \mathbb{N}$. Then, RN transmits $\mathbf{Y}$ to DN. Finally, once DN received $\mathbf{Y}$, it performs the decompression of $\mathbf{Y}$ and restores information. Throughout the paper, it is assumed that received packets are not distorted by fading and additive white Gaussian noise while the packet is dropped once the collision among packets occurred.

B. Intermittent Receiver-Driven Data Transmission

In this paper, the IRDT is assumed as a transmission protocol. Note that our proposed packet aggregation method is surely applicable to any multi-hop-based protocols.

Fig. 2 illustrates the transmission procedure of IRDT. Every node wakes up every intermittent interval denoted by T_{IDLE} and decides to be a transmitter (Tx) or a receiver (Rx) where a node with a new data packet (DATA) becomes Tx where the packet generation interval is defined by T_{G}.

When a node becomes Rx, it broadcasts *ready-to-receive* (RTR) packet to inform neighboring nodes that the node is ready for reception. Upon sending RTR, Rx carriers out short-period listening to recognize whether active Tx exists or not.

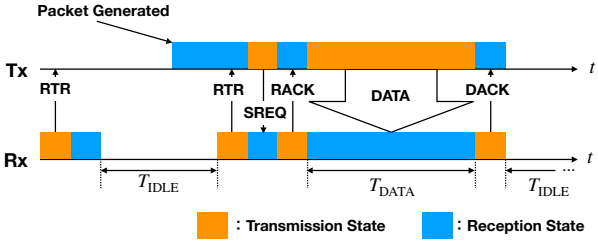

Fig. 2. Packet transmission based on IRDT [3].

If Rx does not receive a *send-request* (SREQ) packet from anyone during the period, it gets back to sleep.

Meanwhile a node becomes Tx, it carries out long period listening until it receives the RTR in order to find the *valid* Rx which should have the sufficient received power and be closer to the common destination. If it is valid, Tx immediately sends the SREQ back to Rx. Rx then returns *receive acknowledgement* (RACK), and the connection is established. Finally, Tx sends the DATA of which length is T_{DATA} to RX, and, if and only if the DATA packet is correctly received at Rx, Rx returns *acknowledgement for data* (DACK). If Tx cannot receive RACK or DACK, it discards the DATA packet and gets back to the sleep state. Note that let the current I_{T}, I_{R}, I_{S} correspond to transmission, reception, and sleep states, respectively, while the circuit voltage is V.

For simplicity, we assume that RN operates as Tx or Rx whereas SN and DN do as only Tx and as only Rx, respectively. Moreover, we assume that the ideal carrier sensing is performed by every node before any transmission to avoid collision, and SNs can recognize each other and RN's transmission perfectly while SNs cannot recognize DN's transmission and vice versa.

As explained in the previous subsection, the communication failure occurs only when the collision happened, and there exist three different cases of the communication failure: 1) SREQ collision, 2) RTR-SREQ collision, and 3) RTR-DATA collision [5]. SREQ collision is a case that SREQs are simultaneously transmitted by multiple nodes. RTR-SREQ collision and RTR-DATA collision occur among hidden nodes. For example, even with the perfect carrier sense, DN may transmit the RTR to RN while RN is receiving SREQ or DATA from SN. Considering the fact that the SREQ packet is even shorter than the DATA packet, RTR-SREQ collision is negligible, and we assume that the communication failure occurs due to SREQ collision or RTR-SREQ collision in the following. Note that, when RN as the bottleneck node transmits the DATA packet to DN, SN cannot receive any RTR, and this results in the SREQ collision at the next reception of RTR from RN. Therefore, decreasing the occurrence probability of the SREQ collision is the most important design factor to address the bottleneck problem in IRDT.

III. Proposed Packet Aggregation

Based on the system model described above, we here propose a new packet aggregation method in combination with EtC employing a block cipher in CBC mode with a short-length encryption key. In the rest of the paper, it is assumed that RN has the sufficient large memory to cache multiple packets for the packet aggregation. Furthermore, we also assume that DN perfectly knows the encryption key $\mathbf{K}^{(1)}, \ldots, \mathbf{K}^{(M)}$ and the occurrence probability p of the element "1" in information packet $\mathbf{X}^{(1)}, \ldots, \mathbf{X}^{(M)}$.

Fig. 3 illustrates the operation of the method. In our proposed method, RN continuously operates as Rx until it collects M_{agg} packets from SNs. After RN collects M_{agg} packets, it applies the EtC to the collected packets as the following steps.

First, RN generates $(N+1)$ sequences $\mathbf{Y}_0, \ldots, \mathbf{Y}_N \in \{0,1\}^{M_{\mathrm{agg}}K \times 1}$ from all the aggregated cipher blocks from $\mathrm{SN}_{m_1}, \ldots, \mathrm{SN}_{m_i}, \ldots, \mathrm{SN}_{m_{M_{\mathrm{agg}}}}$ where $m_i \in \{1, \ldots, M\}$ represents the index of the resulting transmission order of SNs to RN. Then, the sequence $\mathbf{Y}_n = (Y_{n,1}, \ldots, Y_{n,M_{\mathrm{agg}}K})^{\mathrm{T}} \in \{0,1\}^{M_{\mathrm{agg}}K \times 1}$, $n = 0, \ldots, N$ can be expressed as

$$\mathbf{Y}_n = \mathrm{Vec}(\mathbf{y}_n^{(m_1)}, \ldots, \mathbf{y}_n^{(m_{M_{\mathrm{agg}}})}). \tag{4}$$

With this expression, RN compresses $\mathbf{Y}_0, \ldots, \mathbf{Y}_{N-1}$ into the sequences $\mathbf{S}_0, \ldots, \mathbf{S}_{N-1} \in \{0,1\}^{J \times 1}$, which must be an invertible operation. We further note that the last sequence $\mathbf{Y}_N$ is not compressed to enable the decompression and decryption at DN.

Let $\mathbf{H} \in \{0,1\}^{J \times M_{\mathrm{agg}}K}$ denote the parity check matrix of the low-density parity-check (LDPC) code [14] with the coding rate $R \in (0,1]$, where $J \triangleq (1-R)M_{\mathrm{agg}}K$. Using the matrix $\mathbf{H}$, the compressed sequence $\mathbf{S}_n \in \{0,1\}^{J \times 1}$ is given by

$$\mathbf{S}_n = \mathbf{H}\mathbf{Y}_n. \tag{5}$$

After the compression, RN integrates $\mathbf{S}_0, \ldots, \mathbf{S}_{N-1}$ and $\mathbf{Y}_N$, into the sequence $\mathbf{Y} \in \{0,1\}^{1 \times R'NK}$ and transmits it to DN. According to (5), R' is given by

$$R' = \frac{1 + (1-R)(N-1)}{N} M_{\mathrm{agg}}. \tag{6}$$

Therefore, the required time of compressed data transmission from RN to DN is $R'T_{\mathrm{DATA}}$. Without loss of generality, this processing is performed with a negligibly short time, so that we ignore this processing time in the rest of the paper. We note that our proposed method can utilize the LDPC code with the M_{agg} times codeword length as compared to the conventional one [10], which does not employ packet aggregation. Since the longer $\mathbf{Y}_n$ induces the lower decompression error probability, the proposed method can improve compression performance.

After DN receives $\mathbf{Y}$, it tries to decompress and decrypt $\mathbf{S}_0, \ldots, \mathbf{S}_{N-1}$ and $\mathbf{Y}_N$ as follows. Firstly, DN divides $\mathbf{Y}_N$ into $\mathbf{y}_N^{(m_1)}, \ldots, \mathbf{y}_N^{(m_{M_{\mathrm{agg}}})}$, as follows

$$\mathbf{y}_N^{(m_i)} = (y_{N,1}^{(m_i)}, \ldots, y_{N,K}^{(m_i)})^{\mathrm{T}} = (Y_{N,(i-1)K+1}, \ldots, Y_{N,iK})^{\mathrm{T}}. \tag{7}$$

Then, DN can obtain the input to block encryption $\hat{\mathbf{x}}_N^{(m_i)}$ from $\mathbf{y}_N^{(m_i)}$:

$$\mathcal{F}^{-1}_{\mathbf{K}^{(m_i)}}[\mathbf{y}_N^{(m_i)}] = \mathcal{F}^{-1}_{\mathbf{K}^{(m_i)}}\left[\mathcal{F}_{\mathbf{K}^{(m_i)}}[\hat{\mathbf{x}}_N^{(m_i)}]\right] = \hat{\mathbf{x}}_N^{(m_i)}, \tag{8}$$

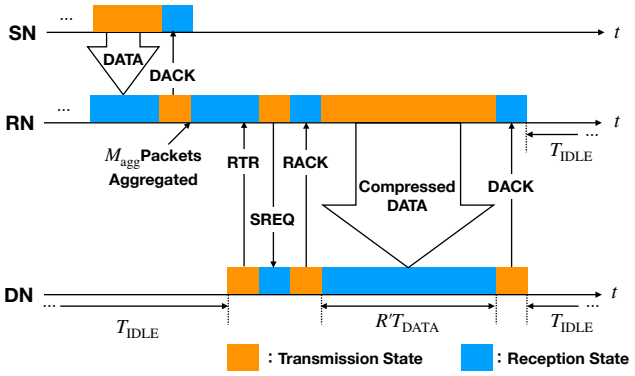

Fig. 3. Packet transmission from RN to DN in the proposed EtC system.

where $\mathcal{F}^{-1}_{\mathbf{K}^{(m)}}[\cdot] : \{0,1\}^{K\times 1} \to \{0,1\}^{K\times 1}$ denotes the decryption execution function for block cipher using $\mathbf{K}^{(m)}$. After that, to obtain the sequence $\mathbf{Y}_{N-1}$, DN decompresses the EtC sequence $\mathbf{S}_{N-1}$ based on the *belief propagation algorithm* (BPA) [15]. The BPA utilizes the decrypted sequence,

$$\hat{\mathbf{X}}_N = \text{Vec}(\hat{\mathbf{x}}_N^{(m_1)}, \ldots, \hat{\mathbf{x}}_N^{(m_{M_{\text{agg}}})}), \tag{9}$$

and p as side information. Since LDPC code $\mathbf{H}$ is utilized to compress $\mathbf{Y}_{N-1}$ to $\mathbf{S}_{N-1}$, the relationship (5) can be represented as a factor graph that consists of two kinds of nodes, i.e., variable nodes and function nodes. In our proposed scheme, each variable node corresponds to each bit of $\mathbf{Y}_{N-1}$, each function node represents a constraint due to (5), and these nodes are connected based on the arrangement of the non-zero components in $\mathbf{H}$. In the BPA, the messages for a maximum a posterior probability estimation are updated at the variable and function nodes, alternately and iteratively. According to (2), $\hat{\mathbf{X}}_N$ can be seen as a noisy version of $\mathbf{Y}_{N-1}$, and they correlate as $\Pr(Y_{N-1,i'} \neq \hat{X}_{N,i'}) = p$ and $\Pr(Y_{N-1,i'} = \hat{X}_{N,i'}) = 1-p$, where $i' = 1, \ldots, M_{\text{agg}}K$. Thus, the initial marginal distributions of the bits of $\mathbf{Y}_{N-1}$ are calculated at each variable node based on this correlation and $\hat{\mathbf{X}}_N$. After that, the messages from variable nodes to function nodes are updated based on the marginal distributions, and they are used to update the messages from function nodes to variable nodes. Finally, the marginal distributions are calculated again utilizing the updated messages from function nodes to variable nodes, and the bits of $\mathbf{Y}_{N-1}$ are estimated. Until estimated $\mathbf{Y}_{N-1}$ satisfies $\mathbf{S}_{N-1} = \mathbf{H}\mathbf{Y}_{N-1}$ or iterations have reached the limit set initially, message exchanging and estimation are continued. DN can obtain the remained sequences $\mathbf{Y}_1, \ldots, \mathbf{Y}_{N-2}$ from the EtC sequences $\mathbf{S}_1, \ldots, \mathbf{S}_{N-2}$, respectively, by the repetition of operations from (7) to (9) and BPA. Using (2), the n-th information block $\mathbf{x}_n^{(m_i)}$ is given by

$$x_{n,k}^{(m_i)} = \hat{x}_{n,k}^{(m_i)} \oplus y_{n-1,k}^{(m_i)} \quad (k = 1, \ldots, K). \tag{10}$$

From the above, all information packets $\mathbf{X}^{(m_1)}, \ldots, \mathbf{X}^{(m_{M_{\text{agg}}})}$ can be obtained.

IV. Numerical Results

In this section, we evaluate the fundamental compression performance of the proposed EtC method and the advantage of our proposed packet aggregation method in terms of packet collision probability and average energy consumption via computer simulations. Note that we evaluate all the performances with the network described in Section II-A without loss of generality.

A. Comparison of Compression Performance

We first confirm the compression performance of the proposed method via up to 1,000,000 computer simulation trials. All simulations in this subsection show the relationship between the decompression error probability and the entropy of the information packet, where $(M_{\text{agg}}, N, K, R) = (15, 3, 128, 0.5)$. In addition, AES [13] is used as the block cipher, and the initial cipher block $\mathbf{y}_0^{(m)}$ is assumed to be a binary sequence that "0" and "1" appear equiprobably. Furthermore, the maximum number of BPA iterations is 40. According to Slepian-Wolf theorem [11], the entropy of the information packet agrees with the theoretical limit of the lossless compression rate. The entropy $H(p)$ of the information packet with the given probability p is given by

$$H(p) = -p\log_2 p - (1-p)\log_2(1-p). \tag{11}$$

Fig. 4 shows the compression performance of the proposed method and the conventional one [10] where the packet aggregation is not applied, i.e. $M_{\text{agg}} = 1$ and the parity check matrix $\mathbf{H}$ is designed based on the Gallager construction method [14]. It can be seen that the proposed method achieves a lower decompression error probability than the conventional one for any entropy.

Fig. 5 shows the compression performance of the proposed method with several different constructions of $\mathbf{H}$. In this comparison, the Gallager construction method and the one based on the IEEE 802.11 standard [16] are considered. Since the size of $\mathbf{H}$ is 972×1944 in the IEEE802.11n standard, the sequence length of $\mathbf{Y}_n$ is adjusted by zero-padding (ZP). For example, when the original length of $\mathbf{Y}_n$ is 1920, the resulting sequence after ZP $\mathbf{Y}'_n$ is composed of 1920 bits and 24 zero bits, namely $\mathbf{Y}'_n = (Y_{n,1}, \ldots, Y_{n,1920}, 0, \ldots, 0)$. Obviously, the information on these zero bits is shared with the decoder and can be exploited by BPA. For the purpose of the comparison, the compression performance of the Gallager construction method with ZP is also shown in the figure. As observed in Fig. 5, it can be seen that the best compression performance is obtained when the configuration method of the IEEE802.11n standard is used.

B. Comparison of Packet Collection Efficiency

We here compare the packet collection efficiency of the proposed method and individual transmission with neither aggregation nor compression. For sake of simplicity, we here assume that the entropy of all packets generated by SNs is even lower than the compression rate, and thus the decompression error probability is assumed to be zero. For example, when

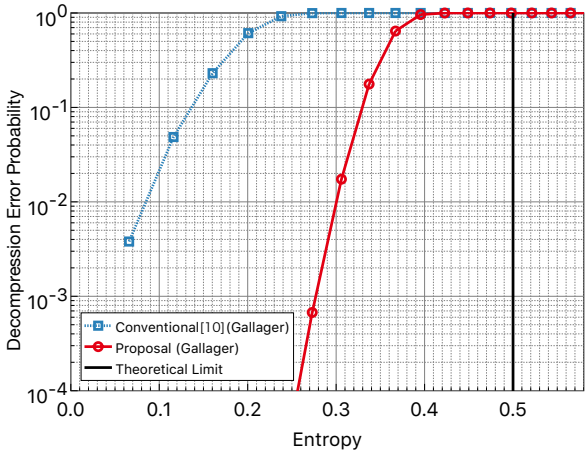

Fig. 4. Comparison of compression performances between the proposed method where $(M_{\rm agg}, N, K, R) = (15, 3, 128, 0.5)$ and conventional one where $M_{\rm agg} = 1$.

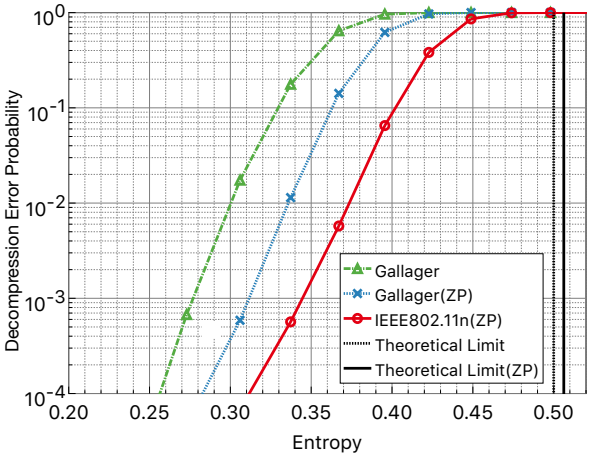

Fig. 5. Comparison of the compression performances with different LDPC matrices where $(M_{\rm agg}, N, K, R) = (15, 3, 128, 0.5)$.

TABLE I
SIMULATION PARAMETERS

Parameter		Proposed	Individual
Packet Generating Interval [min]	$T_{\rm G}$	30	
Number of SNs	M	100	
Coding Rate of LDPC	R	0.5	-
Number of Blocks	N	3	
Length of Blocks	K	128	
Voltage [V]	V	3.6	
Sleeping Current [mA]	$I_{\rm S}$	0.03	
Transmitting Current [mA]	$I_{\rm T}$	18	
Recieving Current [mA]	$I_{\rm R}$	13	
RTR size [bits]		64	
SREQ size [bits]		64	
RACK size [bits]		64	
DACK size [bits]		64	
Data rate [kbits/s]		100	
SREQ WAIT [ms]		5	
RACK WAIT [ms]		2	
DACK WAIT [ms]		2	
Observation time [hour]		24	
Number of trials		100	

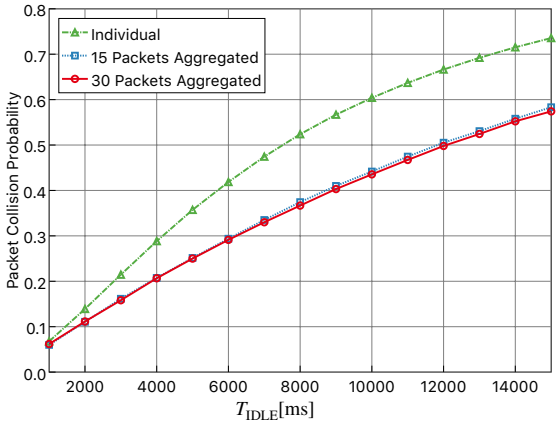

Fig. 6. Relationship between intermittent interval $T_{\rm IDLE}$ and packet collision probability.

the compression rate is half, and the entropy of data packets is 0.2, the error-free is achievable even with Gallager's LDPC codes as obvious from Fig. 5. Besides, all simulations in this subsection follow the simulation specifications shown in Table I. Note that we do not consider the energy consumption of the packet compression and decompression, which is much smaller than transmission, reception, and sleep. We also assume that number of aggregation at RN $M_{\rm agg}$ is $15, 30$. Here, SREQ WAIT, RACK WAIT, and DACK WAIT denote the upper limit of the time waiting for SREQ, RACK, and DACK, respectively.

Fig. 6 shows the relationship between intermittent interval $T_{\rm IDLE}$ and packet collision probability which is the probability that an SN cannot transmit DATA or that neither RN nor DN receives DATA correctly. As seen in the figure, the IRDT scheme employing packet aggregation achieves less SREQ collision probability than the original IRDT. According to [5], the probability of SREQ collision increases as less RTRs are observed at the transmitter. By applying the proposed packet aggregation, RN transmits the RTR more frequently, resulting in the less number of SREQ collisions.

Furthermore, Fig. 7 shows the relationship between interval intermittent $T_{\rm IDLE}$ and the average energy consumption of a node per a correctly received packet at DN. The results shown in the figure indicate that, when $T_{\rm IDLE}$ is large, the IRDT using the proposed method achieves lower energy consumption than that of original IRDT. Also, the energy consumption becomes lower as increasing the number of packet collection $M_{\rm agg}$, inducing the shortening of the waiting time of SNs to receive the RTR from RN.

Finally, Fig. 8 shows the relationship between the aggregate number of packets $M_{\rm agg}$ and the average energy consumption of RN per a correctly received packet at DN. Intermittent interval $T_{\rm IDLE}$ is assumed to be $100, 1000, 5000$, and 10000.

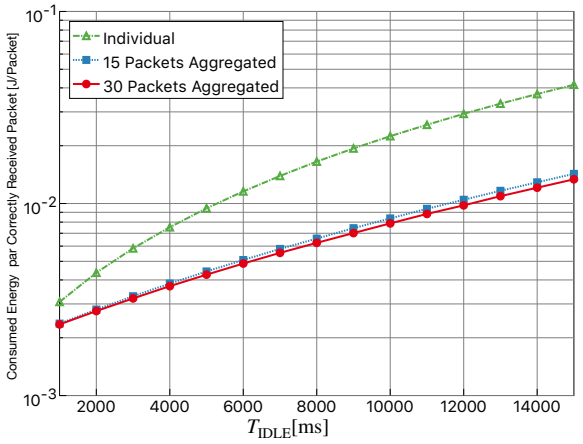

Fig. 7. Relationship between intermittent interval T_{IDLE} and average energy consumption of a node per correctly received packet.

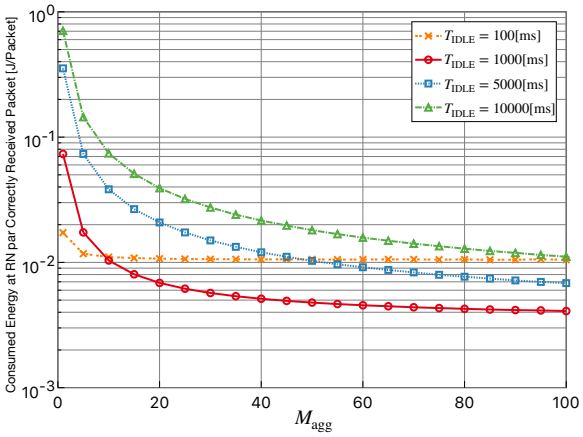

Fig. 8. Relationship between number of aggregated packets M_{agg} and average energy consumption of RN per correctly received packet.

As observed in the figure, the more the packets are aggregated, the lower energy consumption at RN in any intermittent intervals. For example, the energy consumed per a correctly received packet is set to 10^{-2}J as the system requirement. When packets are transmitted individually without aggregation and compression, i.e., $M_{\mathrm{agg}} = 1$, the target cannot be achieved at any intermittent intervals. On the other hand, our proposed method can meet the requirement thanks to the gain given by both compression and aggregation. When $T_{\mathrm{IDLE}} = 10000$, the proposed system cannot meet the requirement with any number of aggregated packets, and thus an option to reduce energy consumption is to reduce T_{IDLE}. However, the energy consumption for $T_{\mathrm{IDLE}} = 100$ is not the best performance among the results as clear from the figure. As pointed out in [5], the smaller T_{IDLE} leads to the higher RTR-DATA collision probability, resulting in the decrease of correctly received packets at DN. Also, although the larger M_{agg} provides the lower energy consumption, it also enforces the higher latency. Therefore, M_{agg} needs to be chosen carefully considering the latency requirement of the application.

V. Conclusions

In this paper, we proposed the EtC with packet aggregation which can extend the sequence. Numerical results show our proposed method can achieve a lower decompressing error probability than the conventional method at any entropy. Moreover, by applying our proposed EtC system to the IRDT protocol, we can decrease the packet collision probability and the average energy consumption of the entire network per a correctly received packet.

References

[1] M. Zorzi and R. R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Multihop performance," *IEEE Trans. Mobile Comput.*, vol. 2, no. 4, pp. 337–348, Oct.-Dec. 2003.

[2] K. Bult et al., "Low power systems for wireless microsensors," in *Proc. 1996 Int. Symp. Low Power Electron. Des.*, Monterey, CA, USA, Aug. 1996, pp. 17–21.

[3] D. Kominami, M. Sugano, M. Murata, T. Hatauchi, and Y. Fukuyama, "Performance evaluation of intermittent receiver-driven data transmission on wireless sensor networks," in *Proc. 6th Int. Symp. Wireless Commun. Syst.*, Tuscany, Italy, Sep. 2009, pp. 141-145.

[4] J. Fujiwara, R. Okumura, K. Mizutani, H. Harada, S. Tsuchiya, and T. Kawata, "Ultra-low power MAC protocol complied with RIT in IEEE 802.15.4e for wireless smart utility networks," in *Proc. 2016 IEEE 27th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Valencia, Spain, Sep. 2016, pp. 1-6.

[5] R. Tanabe, T. Kawaguchi, R. Takitoge, K. Ishibashi and K. Ishibashi, "Energy-aware receiver-driven medium access control protocol for wireless energy-harvesting sensor networks," in *Proc. IEEE Consum. Commun. Netw. Conf. (CCNC)*, Las Vegas, NV, USA, Jan. 2018, pp. 1-6.

[6] M. Zhao and Y. Yang "Bounded relay hop mobile data gathering in wireless sensor networks," *IEEE Trans. Comput.*, vol. 61, no. 2, pp. 265-277, Feb. 2012.

[7] T. P. Wang and Y. C. Chen, "Adaptive packet aggregation for header compression in vehicular wireless networks," in *Proc. 2011 IEEE Int. Conf. High Perform. Comput. Commun.*, Banff, AB, Canada, Sep. 2011, pp. 471–480.

[8] C. E. Shannon, "A mathematical theory of communication," *The Bell Syst. Tech. J.*, vol. 27, no. 3, pp. 379–423, Jul. 1948.

[9] M. Johnson, P. Ishwar, V. Prabhakaran, D. Schonberg, and K. Ramchandran, "On compressing encrypted data," *IEEE Trans. Signal Process.*, vol. 52, no. 10, pp. 2992–3006, Oct. 2004.

[10] D. Klinc, C. Hazay, A. Jagmohan, H. Krawczyk, and T. Rabin, "On compression of data encrypted with block ciphers," *IEEE Trans. Inf. Theory*, vol. 58, no. 11, pp. 6989–7001, Nov. 2012.

[11] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. IT-19, no. 4, pp. 471–480, Jul. 1973.

[12] *Data encryption standard (DES)*, FIPS PUB 197, Washington, DC, USA, Jun. 1997.

[13] *Advanced encryption standard (AES)*, FIPS PUB 197, Gaithersburg, MD, Denmark, Nov. 2001.

[14] R. G. Gallager, *Low-density parity-check codes*, MIT Press, Cambridge, MA, USA, 1963.

[15] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Commun. Lett.*, vol. 6, no. 10, pp. 440–442, Oct. 2002.

[16] *802.11n-2009 - IEEE standard for information technology– Local and metropolitan area networks– Specific requirements– Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications amendment 5: Enhancements for higher throughput*, Sep. 2009.