2022 International Symposium on Nonlinear Theory and Its Applications,
NOLTA2022, Virtual, December 12-15, 2022

Performance evaluation of CSS-APCMA by Experiments using 500 Devices for Massive IoT

Kentaro Honda†, Atsushi Nakamura†, Ferdinand Peper‡, Kenji Leibnitz‡, Naoki Wakamiya§,Mikio Hasegawa†

†Department of Electrical Engineering Tokyo University of Science
6–3–1 Niijuku, Katsushika, Tokyo 125-8585, Japan
‡Center for Information and Neural Networks, National Institute of Information and Communications
1–4 Yamadaoka, Suita, Osaka Prefecture 565-0871, Japan
§Graduate school of Information Science and Technology, Osaka University
1–5 Yamadaoka, Suita, Osaka Prefecture 565-0871, Japan
Email: 4321542@ed.tus.ac.jp, 4322535@ed.tus.ac.jp, peper@nict.go.jp, leibnitz@nict.go.jp,
wakamiya@ist.osaka-u.ac.jp, hasegawa@ee.kagu.tus.ac.jp

Abstract—Low Power Wide Area (LPWA) systems suitable for IoT have been widely deployed. However, collisions and interferences become a severe problem in high-density networks. This paper applies the *Asynchronous Pulse Code Multiple Access (APCMA)* communication scheme to a massive IoT environment. APCMA uses pulses for data transmissions and is effective for high-density wireless environments. We have designed a CSS-APCMA that satisfies the Japanese regulation standard ARIB STD-T108 and evaluated the performance of simultaneous communications with up to 500 transmitters by experiments in an actual environment. Our experimental results show that the proposed system has a high performance even in a high-density environment.

1. Introduction

In 6G, it is expected that IoT devices will be deployed at a density of 10^7 per km^2 [1]. In such a high-density environment, also called *Massive IoT*, collisions of transmitted messages are a serious problem. In IoT implementations of LPWA networks, like the well-known *LoRa* [2] and *Sigfox* [3] schemes, these problems are especially prominent, since these schemes use ALOHA for the medium access layer. To solve this problem, *Asynchronous Pulse Code Multiple Access (APCMA)* was proposed in [4]. APCMA is a protocol that encodes messages by trains of pulses in a way such they can be decoded correctly with a high probability even when there are collisions. APCMA is discussed in detail in [4, 5, 6, 7]. In [5, 6] it is shown that APCMA outperforms Code Division Multiple Access (CDMA) and ALOHA in Massive IoT environments. In [7] the effectiveness of APCMA was shown by experiments using 100 Transmitters.

In this paper, we implement 500 transmitters and evaluate the performance of APCMA in a high-density environment. To improve receiver sensitivity, we use chirp spread spectrum (CSS)-APCMA proposed in [8]. CSS is one of the spectrum spreading techniques using a chirp signal whose frequency increases linearly with time, and it is also used for LoRa. CSS-APCMA uses CSS to encode APCMA pulses. In order to communicate in the 920MHz band used in LPWA systems, we have designed and implemented CSS-APCMA to meet the Japanese wireless standard specified in ARIB STD-T108 [9].

2. APCMA

APCMA is a protocol that shares ideas with Communication Through Silence (CtS) [10] and encodes information as pulse time intervals. On the other hand, unlike CtS, APCMA has the feature that it can be decoded even if messages collide. Fig. 1 shows an APCMA system. The transmitter encodes a message using a set of codewords composed of 4 pulses shown in Fig. 2. There is a trade-off in the number of codewords N in a set. Increasing N increases the amount of information that can be sent by one codeword, but the length C of the codeword then becomes longer. In APCMA, each transmitter sends codewords without synchronization, and, since no scheduling of transmissions take place, collisions may occur. In the decoding process, a receiver refers to the received pulse train and the set of codewords. searching for a pattern of pulses that matches a codeword. If it finds a matching codeword, the corresponding message is decoded. APCMA is designed to provide redundancy in codewords, in a way such that decoding can be achieved with a high probability even if multiple codewords collide, as shown in Fig. 1.

The design of codewords is important because it greatly affects the performance of APCMA. Fig. 3 shows an APCMA codeword when m pulses are used. The functions $f_1(x)$ to $f_(m-2)(x)$ are called encoding functions and they are uniquely determined by x. Here, x is the value of the message to be sent, and Δp is the pulse width. Codes

ORCID iDs Kentaro Honda: 0000-0001-9829-346X, Atsushi Nakamura: 0000-0002-9530-639X, Ferdinand Peper: 0000-0002-8576-7934, Kenji Leibnitz: 0000-0002-3691-3675, Naoki Wakamiya: 0000-0002-6195-6087, Mikio Hasegawa: 0000-0001-5638-8022

are designed such that correlations between codewords are minimized [4]; this makes a message robust against false detection when a collision occurs. Depending on the number of pulses per codeword and the way pulses are modulated, the robustness of codewords against false detections in communication will change.

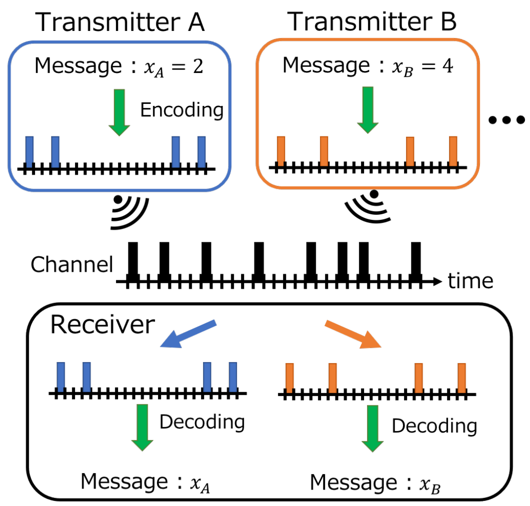

Figure 1: APCMA communication system

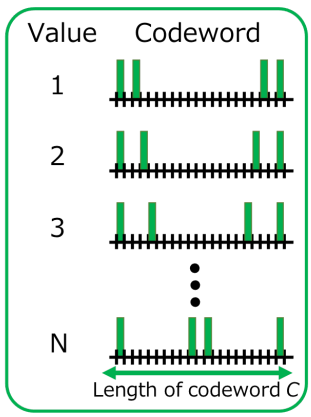

Figure 2: Example of APCMA codeword set

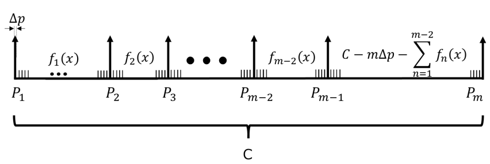

Figure 3: Format of APCMA codeword

3. Design and Implementaion

ARIB-STD T108 is a Japanese radio standard for the 920MHz band used by schemes such as LoRa and Sigfox. In order to meet the standard, we have designed an APCMA pulse code such that the duty cycle of transmission time is less than 0.1%. Since a code word becomes too long if it encodes a large value, it is necessary to divide it into multiple frames of shorter lengths [5]. In this paper a message is divided into two frames: the address is encoded in the first frame and the data is encoded in the second frame. The last pulse of the first frame and the first pulse of the second frame are shared, and if all frames are valid codewords, they are considered as having been decoded without false detections.

Table 1: Experimental Parameters

Parameter	**Value**
Antenna power	1mW
Center Frequency	922.8~927.6MHz
Band width	125kHz~1MHz
The number of codewords N	64~4096
Pulse width Δp	128~8192us

Fig. 4 shows the implemented transmitter. The transmitter consists of a TILMX2491 synthesizer and an MCU. The MCU controls the pulse time interval and tells the TILMX2491 synthesizer when to send the pulse.It generates and sends either a single frequency pulse (Constant Wave; CW) or a chirp pulse. The transmitter is designed with appropriate APCMA codewords according to the wireless parameters in Table 1, and certified by the Telecom Engineering Center (TELEC). Fig. 5 shows the transmitter signal when the number of pulses is 4 per frame, there are two frames, a pulse is a chirp with a linearly increasing frequency, the number of codewords N is 64, and the pulse width Δp is 512us.

Figure 4: Implemented transmitter
Left 1 transmitter enlarged, right 500 transmitters

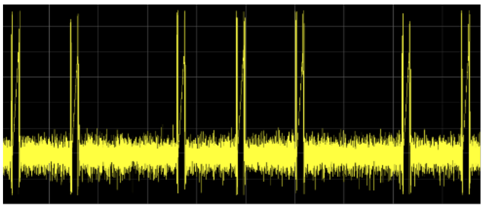

Figure 5: Transmitter signal
(Vertical axis: time, Horizontal axis: frequency[kHz])

Fig. 6 shows the receiver, implemented by a USRPx310 connected to a PC (GNU Radio, software written in C). USRP samples the received signal as a pair of real and imaginary parts. Then, using Gnuradio, it multiply the result by a down chirp signal to detect a chirp pulse signal. Finally, the software decodes the APCMA codeword and outputs the message.

Figure 6: Implemented receiver
Left USRPX310, right PC

4. Experiment

In APCMA, the receiver can decode even if multiple messages collide, but in a high-density environment, misdetection of messages may occur. Fig. 7 shows an example of misdetection in which an invalid pattern is accidentally generated and decoded as valid when two messages overlap to cause ambiguities. In this case, the receiver decodes three messages, two correct messages, and a message that has not been sent. Such a misdetected message is called a *ghost message*.

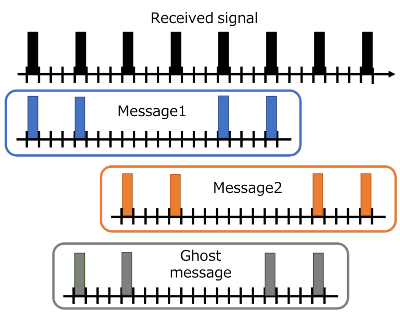

Figure 7: Misdetection

For evaluation, the messages are classified into three categories. m_d are messages decoded by the receiver, m_r are proper messages sent from the transmitter, and m_g are ghost messages. Therefore, $m_d \supset m_r$, $m_d \supset m_g$ and $m_r \cap m_g = \emptyset$. From these messages, we define the communication success probability as Eq. (1).

$$SuccessProbability = 1 - \frac{m_g}{m_d} \quad (1)$$

Since each transmitter sends a message in each transmission cycle IT, the total transmission rate can be defined as in Eq. (2) with the number of codewords being N and the number of transmitters being n. Therefore, increasing the total transmission rate increases the duty ratio of each transmitter, resulting in heavy traffic.

$$Totaltransmissionrate = n \times \frac{2 \times log_2(N)}{IT} \quad (2)$$

In this paper, the center frequency of the transmitter is 922.0MHz, the Spreading Factor of the chirp pulse used for modulation is 7, the pulse width Δp is 512us, and the number of transmitters n is 500. Each transmitter in the rack in Fig.4 sends a message, after which the receiver at a distance of 1.9m receives the messages. To analyze the number of received messages with valid values, we set the transmitter address to a value in the range from 1 to 500, so that the validity of messages from each transmitter individually can be counted.

4.1. Communication verification

Fig. 8 shows the decoding process when $N = 4096$, each codeword has 5 pulses, the message length is 13.5s, and total transmission rate is 100bps. The horizontal axis indicates time, and the vertical axis indicates the address part of the received message. As shown in Fig. 8 simultaneous communication with 500 units was verified. Also, since it is decoded at intervals shorter than the message length, it can be confirmed that it can be decoded even if the messages collide. On the other hand, since invalid messages with addresses higher than 500 were decoded, it was also confirmed that ghost messages occurred.

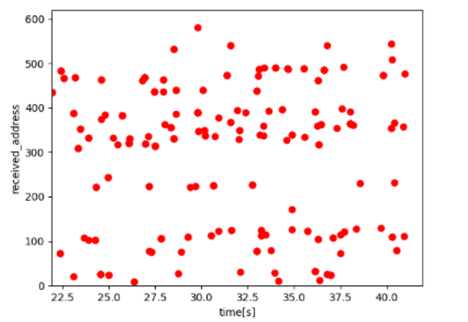

Figure 8: Communication verification

4.2. Performance evaluation

Fig. 9 shows the success probability of the code when the number of codewords N is 2048 or 4096, and the number of pulses is 5 or 4. The message length is 13.5s for 5 pulses and $N = 4096$, 7.2s for 5 pulses and $N = 2048$, 9.1s for 4 pulses and $N = 4096$, and 4.9s for 4 pulses and $N = 2048$. As the total transmission rate increases, the transmission cycle IT decreases according to Eq. (2), so the duty ratio increases, and the probability of message collision increases. Therefore, as the total transmission rate increases, the communication success probability decreases. In any code design, increasing the total transmission rate makes the network traffic denser and decreases the success probability. Among them, in the case of 5 pulses and $N = 2048$, the success probability was suppressed to 0.89 with a bit rate of 100bps. It was confirmed that increasing the number of pulses in a codeword makes it more robust to misdetection. This is a more sparse codeword for larger N, so the chances of a ghost message are reduced. However, increasing N increases the length of the message and increases the likelihood of collisions from different transmitters. Since this latter influence is more dominant than the former, it is considered that the result is as shown in Fig. 9.

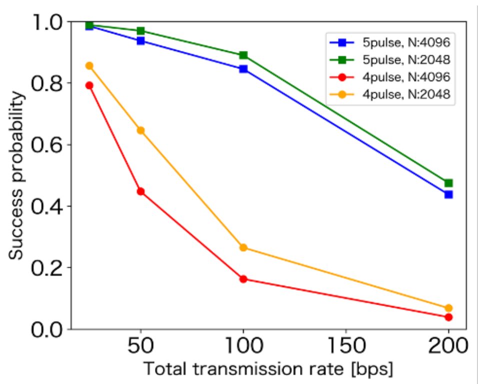

Figure 9: Success Probability

5. Conclusion

In this paper, we implemented 500 transmitters and 1 receiver that satisfied the ARIB-STD T108 standard and we succeeded in simultaneous communication originating with 500 transmitters. We also evaluated the performance of APCMA in a high-density environment using the transmitters and one receiver. As a result, it was confirmed that with a number of codewords N equalling 4096 with 5-pulse codewords a high communication success rate is maintained even in a high-density environment.

Acknowledgments

This research and development work was supported by the Strategic Information and Communications R&D Promotion Programme (SCOPE), Ministry of Internal Affairs and Communications (JP205007001), Japan.

References

[1] Z. Zhang et al., “6G Wireless Networks: Vision, Requirements, Architecture, and Key Technologies,” IEEE Vehicular Technology Magazine, vol. 14, no. 3, pp. 28–41, September 2019.

[2] LoRa Alliance Technical Committee, “LoRaWAN 1.1 specification,” LoRa Alliance, October 2017; www.lora.alliance.org, accessed June 2022.

[3] A. Lavric, A. I. Petrariu and V. Popa, “SigFox Communication Protocol: The New Era of IoT?,” 2019 International Conference on Sensing and Instrumentation in IoT Era, pp. 1-4, 2019.

[4] F. Peper et al., “High-Density Resource-Restricted Pulse-Based IoT Networks,” IEEE Transactions on Green Communications and Networking, vol. 5, no. 4, pp. 1856–1868, December 2021.

[5] N. Wakamiya, K. Leibnitz, F. Peper, M. Hasegawa, “Evaluation and optimization of asynchronous pulse code multiple access,” Nonlinear Theory and Its Applications, vol. 13, no. 2, pp.318–323, April 2022.

[6] K. Leibnitz, F. Peper, K. Theofilis, M. Hasegawa, and N. Wakamiya, “Evaluating multiple-access protocols: asynchronous pulse coding vs. carrier-sense with collision avoidance,” Proc. MUSICAL, November 2021.

[7] Tanaka et al., “Performance evaluation of pulse-based multiplexing protocol implemented on massive IoT devices,” Nonlinear Theory and Its Applications, vol. 12, no. 4, pp. 726–737, October 2021.

[8] K. Honda et al., ”Design and Implementation of Pulse-Based Protocol with Chirp Spread Spectrum for Massive IoT,” International Symposium on Communications and Information Technologies, pp. 172–175, October 2021.

[9] “STD-T108:920MHz-Band Telemeter, Telecontrol and Data Transmission Radio Equipment,” ARIB STD-T108 Version 1.4. April 2021.

[10] Y. Zhu and R. Sivakumar, “Challanges: Communication through silence in wireless sensor networks”, Proc. MOBICOM, pp. 140-147, August 2005.

[11] E. N. Lorenz, “Deterministic non-periodic flow,” *J. Atoms. Sci.*, vol.20, pp.130–141, 1963.