**Indonesian Journal of Electrical Engineering and Computer Science**
Vol. 30, No. 3, June 2023, pp. 1440~1448
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v30.i3.pp1440-1448 

# A novel multiple access communication protocol for LoRa networks without LoraWAN

**Jeremiah Prasetyo[1], Musayyanah[1], Jusak Jusak[1,2]**
[1]Department of Computer Engineering, Faculty of Technology and Informatics, Dinamika University, Surabaya, Indonesia
[2]School of Science and Technology, James Cook University, Townsville, Singapore

**Article Info**

***Article history:***

Received Oct 17, 2022
Revised May 6, 2023
Accepted May 10, 2023

***Keywords:***

Internet of things
LoRa
LoRaWAN
Multi-communication
Multiple access

**ABSTRACT**

Current studies predict there will be approximately 75 billion internet of things (IoT) devices connected to the internet by 2025. Such enormous IoT networks will require efficient communications systems to support those networks running well. We consider one of the emerging communications technologies that is widely used for wireless wide-area network, i.e., long range (LoRa). By default, LoRa wide area networking (LoRaWAN) has been equipped with a medium access control (MAC) protocol, however, several studies showed that LoRaWAN might not be the most excellent choice for certain applications of low-power wide area network, for example, peer-to-peer and mesh networks. In this work, we propose an application layer LoRa multi-communication (LMC) protocol to resolve multiple access problems in LoRa networks without employing LoRaWAN. The algorithm was embedded and tested on energy-constraint devices as building blocks of most IoT systems. Our examination in a controlled environment showed that the proposed algorithm achieved 94% averaged packet reception ratio (PRR) and 33.3 hours longer averaged battery lifetime compared to the default operation (without LMC algorithm) of LoRa networks. The proposed algorithm also introduced longer averaged time on air (ToA) compared to the default LoRa network mainly due to its best-effort service scenario applied to the proposed algorithm.



***Corresponding Author:***

Jusak Jusak
Department of Computer Engineering, Faculty of Technology and Informatics, Dinamika University
Raya Kedung Baruk Road no 98, Surabaya, Indonesia
Email: Jusak.jusak@jcu.edu.au

## 1. INTRODUCTION

A study performed by IRENA in 2019 shows the possibility to have approximately 75 billion internet of things (IoT) active devices deployed worldwide to enable demand-side applications such as smart house, smart grids, smart city, and other sensor-based industrial systems operations by 2025 [1]. These devices not just only connect sensors to a controller, but also create communication without human intermediaries which is commonly called the machine to machine (M2M) communications. The physical nature of these small devices involved in the IoT systems reveals several limitations including low power, low memory capacity, and low computation power. Therefore, a low-power wide-area network (LPWAN) was introduced to accommodate the need for long-range and energy-efficient communication for such devices [2]-[4]. LPWAN aims to connect devices with long-range communications capabilities at a low bit rate. Example of such LPWAN-type technology is the LoRa technology. The LoRa technology has found its implementation in wider communities [5]-[8]and enjoyed broad industry support compared to other LPWAN schemes such as narrowband IoT (NB-IoT), long term evolution machine (LTE-M), and Sig Fox mainly due to its license-free operating frequency privilege, i.e., long range (LoRa) works on chirp spread spectrum (CSS) modulation operated on the unlicensed industrial, scientific, and medical (ISM) frequency sub band [9]-[12]. In Indonesia and around Asia particularly, LoRa networks are operated on the radio spectrum ranging from 920 MHz to 925 MHz [13].

LoRa wide area networking (LoRaWAN), on the other hand, is a protocol stack built on top of LoRa physical layer. With its data link layer protocols support, this LoRaWAN shapes LoRa networks architecture into a typical gateway-nodes model, which allows nodes to transmit data packets to other LoRa devices or to a gateway that acts as a bridge between nodes, network servers, and application servers over a backhaul interface [14]. Figure 1 shows an IoT-based system architecture built on the LoRaWAN networks. A gateway has a specific task to collect data from sensor nodes (end-devices) and relays those to the application server (cloud) through the network servers. The network servers act as the core of the LoRaWAN network that maintains connectivity, routing, and security among devices.

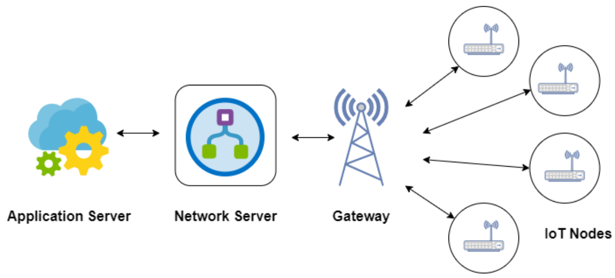



Figure 1. Standard LoRaWAN architecture

Network servers together with the gateways run LoRaWAN protocols to coordinate all nodes in its network. They synchronize data transmission under the so-called access control technique to avoid packets collision under the medium access control (MAC) protocol that is defined in LoRaWAN to guarantee the best communication performances for multiple access connected things or devices to a gateway. The MAC protocol of LoRaWAN was developed to mimick the primitiveareal locations of hazardous atmospheres (ALOHA) model, which permits nodes to transmit data without time synchronization and channel detection [15]-[17]. Although LoRaWAN is popular for wide-area networking deployment, the adopted ALOHA model for its multiple access scheme is widely known to inherit drawbacks such as the increase of average power consumption and collision rate in a dense network deployment [4], [16]. Users can alleviate this downside by properly selecting the end-device's class in which LoRaWAN networks are deployed, i.e. class A (ALOHA) communications that allow an end-device to wake up and start transmitting data at any time, class B (beacon) that permits an end-device to open a receive window and transmit data on a periodic beacon signal, and class C (continue) that sets an end-device to constantly listen to downlink signal from the network. Several MAC layer algorithms are also available in the literature with their main efforts to reduce network collision, increase the efficiency of multiple access algorithms, and enhance the battery life of the end-devices [18]-[22].

Despite the euphoria of LoRaWAN networks for long-range communications spreading fast among communities, several recent works have pointed out collision sense multiple access/collision avoidance (CSMA/CA) that it might not be the most excellent choice for certain applications of LPWAN [23]-[25]. For example, a network with a few numbers of LoRa nodes (e.g., private networks) that transmit data occasionally might find that LoRaWAN deployment is too expensive in terms of both investment and operational cost. Also, a network that relies on peer-to-peer or mesh structures does not require LoRaWAN to support its functional system. Moreover, in some cases, users realize that cloud storage might suit their needs instead of utilizing the standard architecture LoRaWAN that involves the use of network servers and application servers. For these LoRa's use cases, users require to implement their proprietary medium access control to avoid severe collision when the number of nodes is increasing.

In this paper, we propose a novel application layer multiple access control protocol called LoRa multi-communication (LMC) with advantages to reducing the overall power consumption of the LoRa networks and at the same time increasing its packet reception ratio (PRR). The proposed algorithm is a modification of the CSMA/CA principles that has been widely implemented for Wi-Fi (IEEE 802.11) networks. The use of the CSMA model in LoRa network was supported by several preliminary studies in [26], [27]. For example, it is reported that CSMA access control could achieve 90% PRR for simulation using more than four thousand devices in LoRa networks. To test the proposed model in a real environment, we used the RFM9x LoRa modules that act as either transmitter or receiver modules. Therefore, the proposed model in this work is unique in a way that we have leveraged a LoRa module capacity from its common function as a transmitter module (only) to become a transceiver module. The main contributions of the study can be summarized as: i) development of a proposed application layer LMC protocol for creating efficient access control in LoRa networks without employing LoRaWAN protocols. The efficiency of the proposed model was measured by averaged PRR, averaged time on air (ToA), and averaged battery lifetime and ii) we evaluated the proposed multi-communication model in the real environment testbed by using the RFM9x LoRa module.

The paper is organized as follows: we surveyed previous works characterizing LoRa and LoRaWAN networks in the first section. We also elaborated on specific use cases of LoRa networks operated without LoRaWAN to underline the prominence of our proposed model, then we outlined our main contribution. In section 2, we will explore the system architecture of the proposed work and detail the LoRa LMC algorithm followed by results and discussion in section 3 in which evaluation and experimental testbed will be examined. Finally, conclusions will be drawn in the last section.

## 2. METHOD

### 2.1. System architecture

We developed a peer-to-peer network involving three LoRa devices (two IoT nodes and one gateway) separated by several hundred distance to examine the performance of the LMC algorithm. For this purpose, we replaced LoRaWAN protocol stack with the LMC algorithm to manage multiple access traffic into the gateway. Figure 2 depicts a device embedded with LMC algorithm that acted as a gateway and the other two devices equipped with some sensors operated as LoRa nodes. Each of the IoT nodes in Figure 2 transmitted telemetry data temperature, humidity, and direct current consumed by the IoT devices to the gateway periodically using a LoRa module. The LoRa module used in the study is the HopeRF9x with several supporting modules attached to it such as a serial peripheral interface (SPI) module, a modulator-demodulator (modem), and anti-interference and low power spread spectrum communication. In this work, we developed a gateway-like algorithm that ran on top of the LoRa node module. We also employed frequency-hopping and time-slot reservation algorithms to manage a well-run communication system. To guarantee that the algorithm would run well as a gateway, the LoRa node module was designed to act as a transceiver. The selected LoRa module has an inbuilt function to become a gateway by adding applications level components so that it can perform as a transceiver.

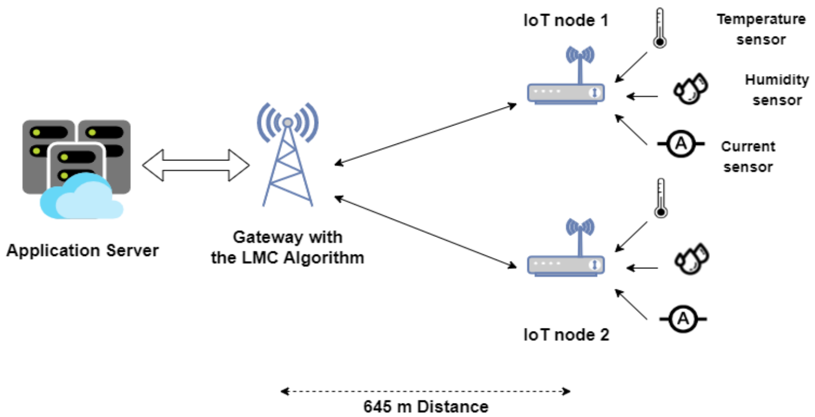
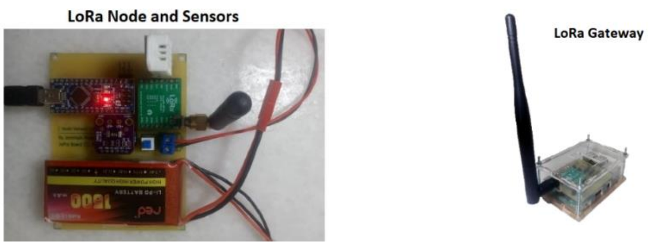



Figure 2. Peer-to-peer network in the study

The proposed LoRa LMC protocol was embedded as firmware in two LoRa nodes and a gateway. Although the LMC protocol has a similar function as the widely known CSMA/CA for managing channel multiple access, they have a significant difference, i.e., while the CSMA/CA protocol runs on the data link layer, the LMC protocol in this work runs on the application layer. The roles and tasks of each node, the gateway, and the application server are elaborated as shown in:

- Nodes were battery-powered devices with 1,500 mAh capacity. The nodes were designed as clients that received commands from a gateway and, in return, the nodes responded requests from the gateway with sensor data that are associated with each node. In our testbed, each node was equipped with a DHT-22 sensor to capture temperature and humidity and an INA219 current sensor. The sensors were integrated into a LoRa HopeRFM95 module to form an IoT node and especially the INA219 sensor performed DC current and voltage measurements to assist us in estimating the battery lifetime of the IoT node. In addition to that, each IoT node that participated in a network was identified by a unique address to allow a gateway sends packet data to the targeted node. Furthermore, the firmware that was embedded in all nodes also supports a sleep mode by activating a watchdog timer feature of a microcontroller. In this way, a node will get into its hibernate mode when it is instructed and switch to its active mode according to the gateway command. Figure 3 shows the physical model used in our testbed that consists of the LoRa node shown in Figure 3(a) and a LoRa gateway shown in Figure 3(b).

- A gateway worked as a commander, receiver, and data logger. Before nodes can respond and send a packet, a gateway initiates communication by sending a broadcast command packet to all nodes in its network with encapsulated node address. Therefore, only a node that has a matched targeted address will respond to the gateway request. Upon receiving the data, a gateway subsequently stores the data temporarily in its data logger. In this work, we employed a Raspberry Pi 3 B+ as the core of gateway firmware. It has an integrated HopeRF95 to support wireless communication through its serial peripheral interface (SPI).
- An application server records all received packet data and allocates the data into a database.




(a) (b)

Figure 3. Physical model of the peer-to-peer networks used in the testbed composed of a LoRa (a) node equipped with sensors and (b) gateway

## 2.2. LoRa multi-communication (LMC) protocol algorithm

The LMC protocol works on the application layer as firmware embedded in both nodes and the gateway. In the initialization phase, a gateway broadcasts by either sending a "sensor command" or a "Sleep Command". Upon receiving the broadcast message, nodes respond it by transmitting either a "sensor packet" or a "sleep acknowledgment (ACK)" command as can be seen in the communication diagram in Figure 4. Detail of the packet structure for the "sensor packet" and the "sleep ACK" commands are shown in Figure 5 and Figure 6, respectively. The packet size of the "sensor packet" command varied between 33 bytes and 46 bytes while the packet size of the "sleep ACK" command can be between 34 bytes and 36 bytes.

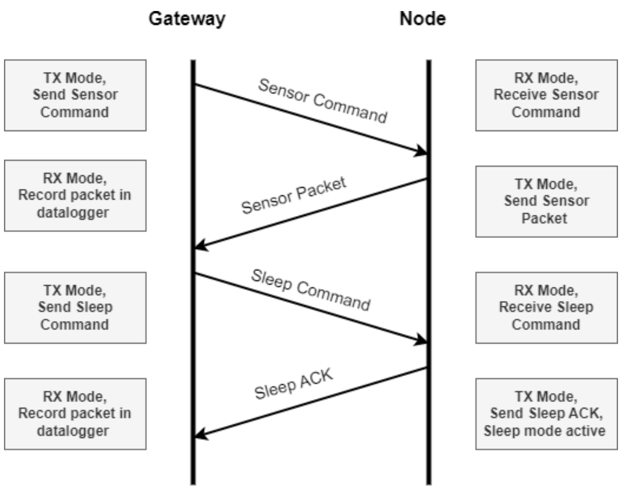



Figure 4. Time diagram of the LMC protocol

| rxAddress | txAddress | msgID | msgLength | status | temperature | humidity | current |
|---|---|---|---|---|---|---|---|

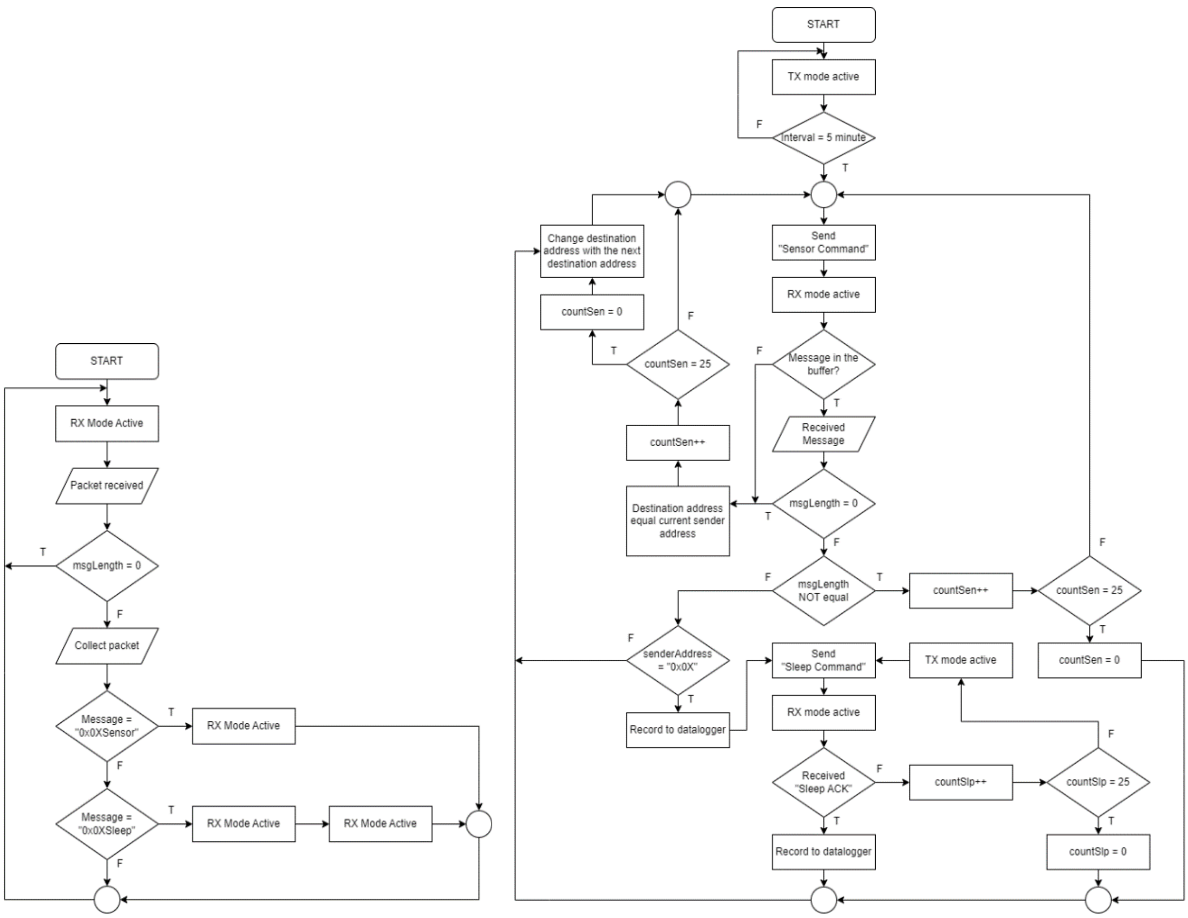
Figure 5. Sensor packet structure

| rxAddress | txAddress | msgID | msgLength | status |
|---|---|---|---|---|

Figure 6. ACK packet structure

### 2.2.1. A node with the LMC algorithm

By default, when nodes are not communicating with the gateway (idle), they are set in receive (RX) mode to wait for commands from the gateway. When a node receives a packet containing a matched node address, it instantly switches to transmit (TX) mode and executes the gateway commands. For example, when the packet contains a “sensor command” command, the node will respond to the gateway by sending sensor data. On the other hand, when it receives a “sleep command” command, the node will respond to the gateway by sending an acknowledgment (ACK) packet and sleep for five minutes to maintain a long battery lifetime.With this algorithm, the collision between nodes can be avoided. Gateway in our algorithm takes control of all communication handshakes between nodes and gateway, therefore, the nodes are strictly prohibited to send the data packet to the gateway without request from the gateway. The algorithm will run continuously in the nodes until the nodes run out of battery. Figure 7 depicts flowchart diagrams of the proposed LMC algorithm. Figure 7(a) shows a flowchart for the LMC algorithm running of the nodes and Figure 7(b) shows a flowchart for the LMC algorithm running on the gateway.




(a) (b)

Figure 7. Flowchart of the LMC algorithm operated on the (a) nodes and (b) gateway

### 2.2.2. A gateway with the LMC algorithm

Figure 7(b) shows a flowchart diagram of the LMC algorithm operated in the gateway. The LMC algorithm that runs in the gateway can be considered the most important part of the whole system to ensure free collision communication between nodes and the gateway. The algorithm adopts the functionality of the well-known CSMA/CA protocol without involving the request to send and clear to send (RTS/CTS) part of the protocol. As shown in Figure 4, a gateway can only initiate two types of commands, i.e., “sensor command” or “sleep command”. For example, when a gateway sends a sensor command, it makes itself ready to receive packets from a targeted node that might contain sensor data. Subsequently, after receiving the data, the gateway records it in its datalogger. However, when the packet length received is longer or shorter than that as expected or empty, a gateway will repeatedly send “sensor command” commands for maximum twenty-fivetimestothe targeted nodes until the expected length of the packet is received. When the maximum number is exceeded, the gateway will try to send the “sleep command” command to the node. If the targeted node does not respond with ACK, the gateway will change the destination address and assume the previous node has been in sleep mode. This algorithm in the gateway will run continuously until the power source is plugged out. The twenty-five times threshold value used in this algorithm was determined based on the experimental testbed. This number is a moderate value that represents a tradeoff between the waiting time for the gateway to get a response from the nodes and the time for enabling gateway to query another active node.

Figure 8 shows the detailtime slot of the gateway operation. It can be seen that the LMC algorithm introduced 200 ms transition time as well as 200 ms delay. The first transition was set to provide adequate time for the gateway to switch from TX mode (sending “sensor command”) to RX mode (prepare for receiving the data packet from a node), whereas the second transition was introduced to provide sufficient time for the gateway to switch from TX mode (sending “sleep command”) to RX mode (prepare for receiving “sleep ACK” from a node). The algorithm allowed 200 ms delay for receiving the data packet from nodes.

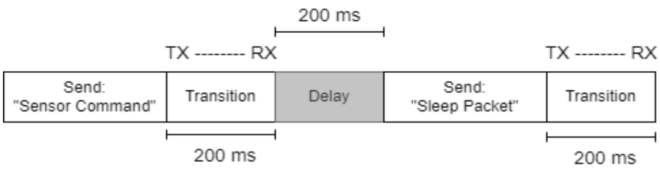



Figure 8. Gateway operation time slot

## 3. RESULTS AND DISCUSSION

To evaluate the performance of the proposed LMC algorithm we exercised parameters PRR to measure the percentage of data packets received by the gateway, ToA to represent transmission time between nodes and gateway, and estimation of the battery life time of all nodes included in the experiment. Results were recorded in a data logger. We utilized parallax data aquisition (PLX-DAQ) software during the experiment to provide an easy analysis of all data. In all our experiments, we used the factory default configuration LoRa module with approximately 645 meters distance between the gateway and nodes. The gateway was located on a high-rise building to minimize obstructions during the experiment. We also attached a battery with 1,500 mAh capacity for each node. We ran two hours of observation for each experiment to get accurate data for analysis during which it was estimated that a gateway could receive a total of 150 packets.

Figure 9 shows a performance comparison in terms of PRR, ToA, and battery lifetime for both cases, i.e., a LoRa network employing the proposed LMC protocol and a LoRa network without running the LMC protocol, while numerical measurement detail of the PRR, TOA, and battery lifetime shown in Table 1 and Table 2 for both cases, consecutively. The figure clearly shows that LoRa network controlled by our proposed LMC protocol outperforms the default operation of the LoRa network (without LMC protocol) in terms of the PRR and battery lifetime. For example, Table 3 shows the average PRR achieved 94% when using the proposed LMC protocol, while it could only achieve average PRR of approximately 9% without using the proposed protocol. The high percentage of PRR in this experiment is mainly due to a small number of packet collisions managed by the gateway when receiving packet data from nodes. Without this LMC algorithm, all nodes in the networks competed with each other to get access to the gateway resulting low percentage of correct packet data received by the gateway. Table 2 also shows the averaged battery lifetime of each node at 71.1 hours for continuous data transmission using the LMC algorithm, which is 33.3 hours or 189% longer time than the node without running the LMC algorithm.

However, while enjoying advantages with its high PRR, implementation of the LMC protocol in the networks contributed to the increase of the averaged ToA at as much as 124 ms compared to the LoRa networks without operating the LMC algorithm. The repetition procedure run by the LMC algorithm as shown in Figure 7 bis largely responsible for the increment of this ToA, i.e., the LMC algorithm was set to continuously query a node up to 25 times at the most extreme conditions when a node does not respond properly or when the received packet size is either larger or smaller than the standard. Table 3 shows a performance comparison for LoRa network using LMC protocol (A) and LoRa network using default operation, i.e., without employing LMC protocol (B). The table showed clearly that the averaged PRR difference between scheme “A” and scheme “B” is approximately 85.1%, averaged ToA difference is approximately 124 ms, and averaged battery lifetime difference reaches as much as 33.3 Hours. Our proposed LMC protocol significantly providesan advantage in energy usage reduction due to its implementation of sleep mode when a node is in its idle time, as such it can lower the power usage and preserve the battery lifetime.Based on this evaluation, our study proved the effectiveness of our proposed multi-communication protocol in achieving better quality communication between nodes and gateway in LoRa networks without operating LoRaWAN.

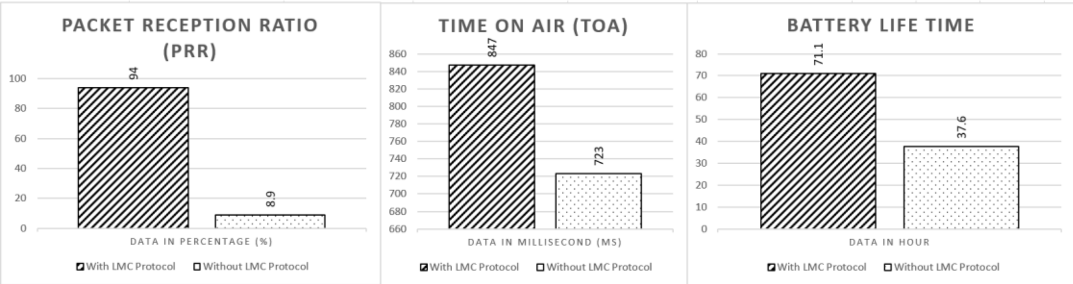



Figure 9. Bar graph comparison of the PRR, ToA, and battery lifetime performance for nodes examined using the LMC protocol and without the LMC protocol

Table 1. Experimental results using the LMC protocol in 3 trials

| PRR (%) | | | ToA (ms) | | | Wake time (2xToA) (ms) | | | Battery lifetime(hours) | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1st | 2nd | 3rd | 1st | 2nd | 3rd | 1st | 2nd | 3rd | 1st | 2nd | 3rd |
| 96 | 88 | 98 | 750 | 940 | 850 | 1500 | 1880 | 1700 | 71.2 | 70.8 | 71.2 |
| Avg=94 | | | Avg=847 | | | | | | Avg=71.1 | | |

Table 2. Experimental results without the LMC protocol in 3 trials

| PRR (%) | | | ToA (ms) | | | Battery lifetime (hours) | | |
|---|---|---|---|---|---|---|---|---|
| 1st | 2nd | 3rd | 1st | 2nd | 3rd | 1st | 2nd | 3rd |
| 4.3 | 18.5 | 3.9 | 721 | 723 | 724 | 38.9 | 38.5 | 35.9 |
| Avg=8.9 | | | Avg=723 | | | Avg=37.8 | | |

Table 3. Performance comparison for LoRa network using LMC protocol (A) and default operation – without using LMC protocol (B)

| Parameter | PRR (%) | | ToA (ms) | | Battery lifetime (hours) | |
|---|---|---|---|---|---|---|
| | A* | B* | A* | B* | A* | B* |
| | 94 | 8.9 | 847 | 723 | 71.1 | 37.8 |
| Difference | 85.1 | | 124 | | 33.3 | |

**A: LoRa networks with LMC algorithm; B: LoRa networks without LMC algorithm*

## 4. CONCLUSION

In this study, we proposed an application layer-based LMC protocol. The proposed multi-communication algorithm was adopted from the CSMA/CA algorithm that was originally designed to provide collision resistance in a network. Each node is identified by its unique address. A gateway in this study has a main task to coordinate and ensure all nodes have their slot time to send its data packet. After being instructed by the gateway, the targeted node makes it self-available to send data packets that contain sensor values. In addition to that, a gateway can force the nodes to run in their sleep mode taking advantage of its small power consumption to save the battery lifetime. The node will active after 300 s (5 minutes).

Experiment results showed that a LoRa network employing the LMC protocol provided 94% averaged PRR which is much greater than the LoRa network without the LMC protocol. Moreover, the proposed algorithm outperformed the default LoRa network at approximately 189% (i.e., 33.3 hours) longer averaged battery lifetime. The only drawback introduced by the proposed LMC algorithm is that it gave 124 ms slower ToA compared with the default LoRa network. This is because the protocol was purposely designed to continuously query a node up to 25 times when a node does not respond properly or when the received packet size is either larger or smaller than expected.

## REFERENCES


[1] "Innovation landscape brief: internet of things," IRENA International Renewable Energy Agency, Abu Dhabi, 2019.

[2] M. A. Ertürk, M. A. Aydın, M. T. Büyükakkaşlar, and H. Evirgen, "A Survey on LoRaWAN architecture, protocol and technologies," *Future Internet*, vol. 11, no. 10, p. 216, Oct. 2019, doi: 10.3390/fi11100216.

[3] B. S. Chaudhari, M. Zennaro, and S. Borkar, "LPWAN technologies: emerging application characteristics, requirements, and design considerations," *Future Internet*, vol. 12, no. 3, p. 46, Mar. 2020, doi: 10.3390/fi12030046.

[4] Z. Sun, H. Yang, K. Liu, Z. Yin, Z. Li, and W. Xu, "Recent advances in LoRa: a comprehensive survey," *ACM Transactions on Sensor Networks*, vol. 18, no. 4, pp. 1–44, Nov. 2022, doi: 10.1145/3543856.

[5] D. Tamang, A. Pozzebon, L. Parri, A. Fort, and A. Abrardo, "Designing a reliable and low-latency LoRaWAN solution for environmental monitoring in factories at major accident risk," *Sensors*, vol. 22, no. 6, p. 2372, Mar. 2022, doi: 10.3390/s22062372.

[6] M. O. Ojo, D. Adami, and S. Giordano, "Experimental evaluation of a LoRa wildlife monitoring network in a forest vegetation area," *Future Internet*, vol. 13, no. 5, p. 115, Apr. 2021, doi: 10.3390/fi13050115.

[7] M. El-Aasser, R. Badawi, M. Ashour, and T. Elshabrawy, "Examining carrier sense multiple access to enhance LoRa IoT network performance for smart city applications," in *IEEE International Conference on Consumer Electronics - Berlin, ICCE-Berlin*, Sep. 2019, vol. 2019-September, pp. 168–173, doi: 10.1109/ICCE-Berlin47944.2019.8966182.

[8] B. Citoni, S. Ansari, Q. H. Abbasi, M. A. Imran, and S. Hussain, "Comparative analysis of an urban LoRaWAN deployment: real world versus simulation," *IEEE Sensors Journal*, vol. 22, no. 17, pp. 17216–17223, Sep. 2022, doi: 10.1109/JSEN.2022.3193504.

[9] A. Augustin, J. Yi, T. Clausen, and W. M. Townsley, "A study of LoRa: Long range and low power networks for the internet of things," *Sensors (Switzerland)*, vol. 16, no. 9, p. 1466, Sep. 2016, doi: 10.3390/s16091466.

[10] J. Haxhibeqiri, E. De Poorter, I. Moerman, and J. Hoebeke, "A survey of LoRaWAN for IoT: from technology to application," *Sensors (Switzerland)*, vol. 18, no. 11, p. 3995, Nov. 2018, doi: 10.3390/s18113995.

[11] J. D. C. Silva, J. J. P. C. Rodrigues, A. M. Alberti, P. Solic, and A. L. L. Aquino, "LoRaWAN - A low power WAN protocol for internet of things: A review and opportunities," in *2017 2nd International Multidisciplinary Conference on Computer and Energy Science, SpliTech 2017*, 2017, pp. 1–6.

[12] N. Sornin and A. Yegin, *LoRaWAN™ 1.1 Specification 2*, Beaverton: LoRa Alliance, Inc. 2017.

[13] Y. Triwidyastuti, M. Musayyanah, F. Ernawati, and C. D. Affandi, "Multi-hop communication between LoRa end devices," *Scientific Journal of Informatics*, vol. 7, no. 1, pp. 125–135, Jun. 2020, doi: 10.15294/sji.v7i1.21855.

[14] J. S. Andersen and J. Eriksson, "Investigating the practical performance of the LoRaWAN technology," Master thesis, Department of Computer Science, Linköping University, 2017.

[15] A. Furtado, J. Pacheco, and R. Oliveira, "PHY/MAC uplink performance of LoRa class a networks," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6528–6538, Jul. 2020, doi: 10.1109/JIOT.2020.2974429.

[16] G. Callebaut, G. Leenders, C. Buyle, S. Crul, and L. Van der Perre, "LoRa physical layer evaluation for point-to-point links and coverage measurements in diverse environments," *arXiv:1909.08300*, Sep. 2019.

[17] O. Iova *et al.*, "LoRa from the city to the mountains: Exploration of hardware and environmental factors," *International Conference on Embedded Wireless Systems and Networks*, pp. 317–322, 2017.

[18] K. Q. Abdelfadeel, D. Zorbas, V. Cionca, and D. Pesch, "FREE - Fine-grained scheduling for reliable and energy-efficient data collection in LoRaWAN," *IEEE Internet of Things Journal*, vol. 7, no. 1, pp. 669–683, 2020, doi: 10.1109/JIOT.2019.2949918.

[19] P. J. Marcelis, N. Kouvelas, V. S. Rao, and R. V. Prasad, "DaRe: Data recovery through application layer coding for LoRaWAN," *IEEE Transactions on Mobile Computing*, vol. 21, no. 3, pp. 895–910, Mar. 2022, doi: 10.1109/TMC.2020.3016654.

[20] R. Sakauchi, H. Yabe, S. Ishikawa, Y. Nomoto, S. Segawa, and M. S. Tanaka, "Study of communication reliability improvement using multi-channel communication for LoRa," in *IEEE International Conference on Consumer Electronics - Berlin, ICCE-Berlin*, Nov. 2021, vol. 2021-November, pp. 1–4, doi: 10.1109/ICCE-Berlin53567.2021.9719875.

[21] L. Leonardi, F. Battaglia, and L. Lo Bello, "RT-LoRa: A medium access strategy to support real-time flows over LoRa-based networks for industrial IoT applications," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 10812–10823, Dec. 2019, doi: 10.1109/JIOT.2019.2942776.

[22] C. Zhang, L. Wang, L. Jiao, S. Wang, J. Shi, and J. Yue, "A novel orthogonal LoRa multiple access algorithm for satellite Internet of Things," *China Communications*, vol. 19, no. 3, pp. 279–289, Mar. 2022, doi: 10.23919/JCC.2022.03.020.

[23] R. Fujdiak, K. Mikhaylov, J. Pospisil, A. Povalac, and J. Misurec, "Insights into the issue of deploying a private LoRaWAN," *Sensors*, vol. 22, no. 5, p. 2042, Mar. 2022, doi: 10.3390/s22052042.

[24] R. Berto, P. Napoletano, and M. Savi, "A LoRa-based mesh network for peer-to-peer long-range communication," *Sensors*, vol. 21, no. 13, p. 4314, Jun. 2021, doi: 10.3390/s21134314.

[25] H. C. Lee and K. H. Ke, "Monitoring of large-area IoT sensors using a LoRa wireless mesh network system: design and evaluation," *IEEE Transactions on Instrumentation and Measurement*, vol. 67, no. 9, pp. 2177–2187, 2018, doi: 10.1109/TIM.2018.2814082.

[26] T.-H. To and A. Duda, "Simulation of LoRa in NS-3: Improving LoRa performance with CSMA," in *2018 IEEE International Conference on Communications (ICC)*, May 2018, vol. 2018-May, pp. 1–7, doi: 10.1109/ICC.2018.8422800.

[27] T. Qi and Y. Wang, "For massive access with sporadic traffic in M2M communication: Collision avoidance or collision resolution," *IEEE Access*, vol. 8, pp. 95312–95320, 2020, doi: 10.1109/ACCESS.2020.2995758.

**BIOGRAPHIES OF AUTHORS**

**Jeremiah Prasetyo, S.T.** did his higher studies in Computer Engineering, Dinamika University from 2016-2020. He is attached to the Faculty of Technology and Information. His research area then was on development of internet of things technology. With his lecturers, he developed the LoRa multi-communication protocol based on the low-power wide-area network to provide today's internet of things technology needs with long-distance and power-saving communication capabilities. Currently he served as a Junior System Administrator at a private company with cloud service products that are able to support internet of things. Now he is still active in doing research on cloud and internet of things systems. He can be contacted at email: jere.prasetyo@gmail.com.

**Musayyanah** obtained a bachelor's degree in Telecommunications Engineering Politeknik Elektronika Surabaya (PENS). She also obtained a Master of Engineering degree in 2015 from Institut Teknologi Sepuluh Nopember. Currently, she is a lecturer in the Department of Computer Engineering, Faculty of Information and Technology (FTI), Universitas Dinamika, Surabaya, Indonesia. She is researching wireless sensor networks and the internet of things. She has some articles indexed by G-scholar with 15 citations and h-index of 2. She can be contacted at email: musayyanah@dinamika.ac.id.

**Jusak Jusak** received B.S. degree in electrical engineering from Brawijaya University, Malang, Indonesia and Ph.D. degree in electrical engineering from Royal Melbourne Institute of Technology (RMIT) University, Melbourne, Australia. From 2009 to 2011, he was a Postdoctoral Research with Massey University, Palmerston North, New Zealand working in a next generation networks research project supported the Telecom New Zealand. Between 2011 and 2021, he was a senior lecturer in Computer Engineering Department, Dinamika University, Surabaya, Indonesia. He is currently a senior lecturer IoT at James Cook University Singapore. His research interest includes signal processing for wireless communication networks, biomedical signal processing, and internet of things for medical applications and its security. He was a recipient of Ph.D. research award from the School of Electrical and Computer Engineering, RMIT University. He received several national level research competitive grants from 2011 to 2020 and the best paper award in the International Conference on Information Technology Applications and Systems in 2018. He can be contacted at email: Jusak.jusak@jcu.edu.au.