Research Article

Towards an Improved Energy Efficient and End-to-End Secure Protocol for IoT Healthcare Applications

Arshad Ahmad [1,2] **Ayaz Ullah,**[2] **Chong Feng** [1] **Muzammil Khan** [3] **Shahzad Ashraf,**[4] **Muhammad Adnan,**[5] **Shah Nazir** [2] **and Habib Ullah Khan** [6]

[1]*School of Computer Science & Technology, Beijing Institute of Technology, Beijing, China*
[2]*Department of Computer Science, University of Swabi, Anbar, Pakistan*
[3]*Department of Computer Science, University of Swat, Swat, Pakistan*
[4]*College of Internet of Things Engineering, Hohai University, Changzhou, Jiangsu, China*
[5]*Department of Computer Science, Abdul Wali Khan University, Mardan, Pakistan*
[6]*Department of Accounting & Information Systems, Qatar University, Doha, Qatar*

Correspondence should be addressed to Chong Feng; fengchong@bit.edu.cn

In this paper, we proposed LCX-MAC (local coordination X-MAC) as an extension of X-MAC. X-MAC is an asynchronous duty cycle medium access control (MAC) protocol. X-MAC used one important technique of short preamble which is to allow sender nodes to quickly send their actual data when the corresponding receivers wake up. X-MAC node keeps sending short preamble to wake up its receiver node, which causes energy, increases transmission delay, and makes the channel busy since a lot of short preambles are discarded, as these days Internet of Things (IoT) healthcare with different sensor nodes for the healthcare is time-critical applications and needs a quick response. A possible improvement over X-MAC is that local information of each node will share with its neighbour node. This local information exchanged will cause much less overhead than in the nodes which are synchronized. To calculate the effect of this the local coordination on X-MAC in this paper, we built an analytical model of LCX-MAC that incorporates the local coordination in X-MAC. The analytical results show that LCX-MAC outperformed X-MAC and X-MAC/BEB in terms of throughput, delay, and energy.

1. Introduction

As with rapid development in IoT healthcare devices, a large amount of data is transmitted over limited energy resources, so there is a requirement for IoT healthcare environment which mostly consists of wireless sensor nodes that collect different data from the environment [1, 2]. MAC protocols with duty cycle mechanism have been established mainly for saving energy in IoT healthcare and wireless sensor networks (WSNs). IoT healthcare and WSNs have many numbers of nodes; each node has a transceiver and more than one sensor to monitor changes in the environment and send the sensed data to collection centre directly or through some middle nodes. As the events to inform infrequently occur, the node only wakes up periodically for a short time to send or receive data known as duty cycle mechanism. Node only wakes up for a short time and remains in the sleep mode most of the time. In the sleep mode, the node turns off transceiver to save energy by avoiding idle listening. Thus, a duty cycle interchanges an extended sleep time duration and a short wakeup time duration to save energy.

MAC protocols with duty cycle behaviour are broadly divided into two categories, asynchronous and synchronous. Nodes in asynchronous MAC protocols randomly wakeup and keep sending preamble until its corresponding receivers become awake. Synchronous MAC protocols from time to time exchange their schedule information with their neighbour nodes. The first synchronous MAC protocol was S-MAC [3], and there are many modifications of S-MAC

such as T-MAC [4], RMAC [5], DW-MAC [6], and SCP-MAC [7] with the main purpose being to save energy.

To increase energy efficiency, asynchronous MAC protocols consisting of B-MAC [8], WiseMAC [9], and X-MAC [10] use different method compared to synchronous MAC protocols. In asynchronous MAC protocols, each node is allowed to function independently from its neighbour nodes. The advantage of working independently is that the nodes do not periodically exchange their schedule information; however, sender nodes in asynchronous MAC protocols required continuing to send a long preamble until their corresponding receiver nodes wake up. Even the receiver detects the presence of preamble, but it has to stay awake, unable to reply to its sender node. The sender node at the end of the non-preemptive preamble sends the actual data. Although the asynchronous MAC protocols perform better, for types of networks with infrequent traffic, these protocols have the problem of long preambles duration of one duty cycle. Those WSNs which are less frequently sending data with long duty cycle will consume more energy as it required long preamble to wake up receivers.

B-MAC has many variants. X-MAC is one of the B-MAC variants with two advanced functions. The first function is that X-MAC allows receiver nodes to reply instantly to their sender nodes of the availability, even during the transmission of the preamble. X-MAC divides the long interruptible preamble into short preambles with a break in which the receiver which woke up can promptly respond, which is known as early acknowledgement (ACK). The other function is that X-MAC inserts the target address in each preamble, so the overhearing nodes problem is solved.

Although X-MAC reduced the duration of the long preamble, it still has the performance degradation in the crowded networks due to collisions. There is no mechanism in X-MAC to deal with collisions. As in the X-MAC, the sender node cannot identify the occurrence of the collision, so the nodes send preamble for the complete cycle once a collision occurred. There are possibilities of collisions at the retransmission because of using constant contention window. Note that, in binary exponential backoff (BEB) algorithm, the CW is double as collisions consecutively occur.

Nodes in IoT healthcare and WSNs are overcrowded, so the probability of collision is high, because of *spatially correlated contention (SCC)* [11], in which many nodes closely located sense the same event and concurrently start sending data. For solving the problem, we need to design and control SCC due to the fact that collisions can occur by reducing throughput and energy efficiency. For this problem, we proposed X-MAC/CA [12] and X-MAC/BEB [13] protocols; X-MAC/CA is X-MAC with collision avoidance (CA), and X-MAC/BEB is X-MAC with BEB algorithms to randomize the transmission time. As WSNs become congested, the BEB decrease the effect of collisions by dynamically randomizing the transmission time.

As for X-MAC and X-MAC/BEB, sender node continuously sends short preamble until its receiver node wakes up, which causes energy consumptions and makes the medium busy as many short preambles are discarded because the receiver node is not awake. The possible improvement over X-MAC and X-MAC/BEB is that each node will share its wakeup information with its neighbour node so that each node can know the wakeup time of its neighbour nodes, which is named as LCX-MAC.

The exchange of this wakeup information will cause very low overhead compared to the synchronization of the nodes used in other synchronized protocols, and this will reduce the problem of sending many numbers of sending short preamble, as now each node will only need to send one short preamble. This wakeup information share can help the neighbouring nodes to collaborate access to the shared medium, so this will solve the spatially correlated contention. This research aims to develop an energy-efficient and low latency X-MAC protocol for IoT healthcare and WSNs. To reduce the duration of short preambles of X-MAC/BEB, we will synchronize the wakeup time of the nodes between source and destination.

This research work is an extension of our previous work X-MAC/BEB. In X-MAC/BEB only BEB was added to X-MAC protocol, but in this research using X-MAC/BEB we will adjust the wakeup time of sender nodes according to their corresponding receiver nodes, as the sender nodes in X-MAC and X-MAC/BEB have the problem of sending the short preamble until their receiver replies giving more considerable delay and energy consumptions.

The X-MAC and our proposed X-MAC/CA and X-MAC/BEB are based on CSMA. In CSMA, each node tries to access the shared medium when the data ready to send makes the MAC protocols simple but leads to the "busy medium" as a node keeps sending short preambles to wake up its receivers nodes, which causes energy consumptions and keeps the medium busy as many of short preambles are discarded. The busy medium problem can also affect the result of applying a long duty cycle approach, as the sender node keeps sending many short preamble which increases energy consumptions to wake up its receiver nodes.

Busy medium by sending many short preambles has occurred because sender nodes have no information of their receiver nodes duty cycle information (X-MAC and X-MAC/BEB are asynchronous MAC protocols). The possible extension for this problem over X-MAC and X-MAC/BEB is that each node exchanges wakeup information with its neighbour nodes so that a sender node has wakeup information of its receiver node.

It is important to note that the exchanges of this wakeup information will cause very low overhead compared to nodes synchronization used in synchronized MAC protocols. Still, it will efficiently solve the busy medium problem as each sender node only sends one short preamble by knowing that its receiver node wakes up.

The exchange of this wakeup information allows the neighbouring nodes to coordinately access the shared medium and thus solve the problem spatially correlated contention. For example, each sender node can access medium according to its neighbour nodes access, which causes an increase in the throughput and reduction in energy consumptions. This exchange of neighbour information and how to coordinate are hot research issue that needs more research study.

The remainder of this paper is composed of the following. Section 2 is about related work and Section 3 introduces the LCX-MAC protocol. Section 4 describes Markov model of MAC protocols for WSNs and IoT healthcare. Section 5 presents the extended performance model of LCX-MAC, and Section 5 evaluates the performance of LCX-MAC with that of X-MAC/BEB and X-MAC. Finally, Section 6 concludes the paper.

2. Related Work

For better performance at MAC layer, there are many different extensions of X-MAC proposed such as DDC MAC [14], CL-MAC [15], EX-MAC [16], (VT) [17], RIX-MAC [18], TRIXMAC [19], and LA-MAC [20].

Dynamic Duty Cycle MAC (DDC-MAC) protocol is proposed for the IoT environment which adjusts the duty cycle (DC) ratio of the receiver node and the contention window (CW) size of the sender node according to the traffic congestion for various devices in the IoT. DDC-MAC only reduces transmission delay and energy consumption. Our proposed LCX-MAC not only decreases delay and energy consumption but also increases the throughput.

CLMAC was proposed for decreasing delay in multihop networks; CLMAC uses a synchronous function by including the receiver node address and rendezvous (RDV) point in the short preamble. All the neighbour nodes around the sender nodes should wake up which is shown by RDV point. All the neighbour nodes, including the receiver nodes after reading the preamble, schedule their next wakeup time according to the RDV point. To mitigate collision at RDV point, CL-MAC uses a backoff algorithm with constant CW. CL-MAC did not give any results on how the backoff algorithm has effects on throughput, latency, and energy consumptions.

EX-MAC also wants to reduce latency to save energy in multi-hop networks and IoT health by using reservation requests to synchronize middle nodes between a source and destination. However, EX-MAC has no mechanism for the collisions when the network is overloaded.

Virtual tunnel (VT) was proposed to reduce the end-to-end delay in multihop WSNs and IoT health by implementing synchronization at the neighbour's duty cycle (SND) mechanism. For this, VT inserts the schedule information in each preamble, so each node knows the next wakeup time of its neighbours. VT also does not provide a mechanism for the collisions problem.

There are receiver-initiated MAC protocols, RIX-MAC and TRIX-MAC. Both have the same working mechanism, by which sender nodes adjust wakeup according to their receiver wakeup time by early ACK. Both protocols clearly include the wakeup time of the receiver nodes by including the wakeup time into early ACK. However, it is needed that the nodes periodically synchronize clock time of nodes, which is very difficult in large WSNs and IoT healthcare has many nodes, giving clock drift problem.

LA-MAC lists sender nodes to wake up depending on the priority order of their corresponding receiver nodes wakeup time. LA-MAC also decreases the delay by allowing senders nodes to transmit more than one frame to their receiver.

However, the X-MAC allows only two senders to send their data frames in one cycle when the destination node is the same receiver.

LA-MAC does not provide any mechanism on how it will work under overloaded WSNs to monitor collision.

There is another category of MAC protocols recently used for WSNs known as Quorum-system-based MAC protocols. QTSAC [21] MAC protocol is introduced for minimizing the latency and energy consumptions in WSNs. QTSAC is different from earlier Quorum based MAC protocols using two advanced features. First, it used more QTSs (Quorum time slots) for the nodes far from the sink node; second, QTSs are allocated only, when data is transmitted. However, using more number of QTSs will increase energy consumptions.

All these extensions of X-MAC protocol are proposed for wireless sensor networks. Still, our local coordination MAC with BEB and LCX-MAC not only is suitable for WSNs but also covers IoT healthcare solutions in which the data are very critical and time-sensitive.

For securing end-to-end communication between the source and destination nodes in WSNs, different approaches are used to identify the malicious nodes in the network. Security Disjoint Routing-Based Verified Message (SDRVM) [22] is proposed with different features. The first two sets are created based on remaining energy of a node, a data and v-message set and the second is based on the remaining energy; the node records the ID in the data packet with a specific probability. Third, the duty cycle of a node is adjusted and the energy of the node is divided into three levels. In a dataset, the duty cycle of the node is the longest and the node not in any dataset is the shortest duty cycle. Fourth, if a node has sufficient energy, then data is transmitted several times, and the v-message which is stored in the node is transmitted to the final destination node. However, using these different schemas will increase the energy consumptions of nodes.

3. LCX-MAC (Local Coordination X-MAC)

Figure 1 presents the flow chart of the LCX-MAC algorithm of the sender node. When a sender node wakes up and has some data to send, the node executes *preamble backoff* by randomly drawing an integer number i from the minimum CW (contention window) W_0 of $(0, W_0 - 1)$ and holds the transmission by i number time slots.

After counting to zero of the backoff timer and clear channel assessment (CCA), when the medium is free, then the sender starts transmitting short preambles and waits to receive ACK at the end of each preamble. When the medium is free, then the sender starts transmitting short preambles and waits to receive ACK at the end of each preamble. Once the sender node got ACK, it sends one data frame and goes to sleep mode. If the sender node does not get an ACK for the whole time period due to the collision, the backoff counter k increments. LCX-MAC tries to retransmit at the next cycle until either k reaches a predetermined maximum or the transmission succeeds. When the sender node receives early ACK, so the receiver node sends its wakeup time to its corresponding sender node. The sender node adjusts its
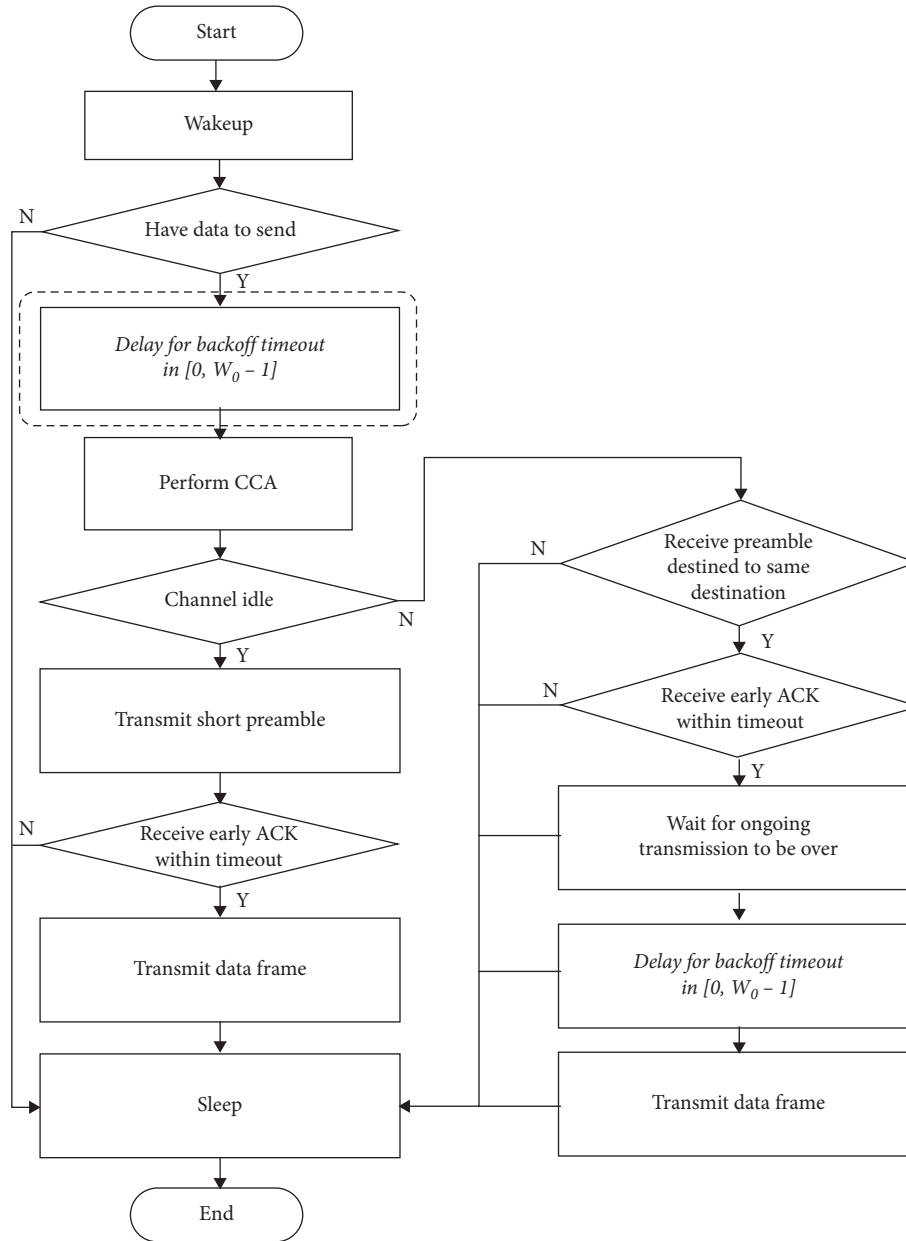
FIGURE 1: Sender side flow chart of LCX-MAC algorithm.

wakeup time according to its receiver node. At the end of the successful transmission, k decremented, unless $k = 0$.

When the medium is busy after counting to zero of the backoff timer, the sender node stays awake and receives the other node preamble; when it comes to knowing that destination address is the same, it wants to send data, so it stays awake for the current transmission to be finished. At the end of this current transmission, the other sender nodes use constant contention window backoff to avoid the collision because maybe more than one sender is waiting to send its data frame.

The behaviour of X-MAC/BEB and LCX-MAC is shown in Figure 2. Two sender nodes, S_1 and S_2, competing to win the channel send their data frames to their receiver R_1 and R_2, respectively. As sender node S_1 chooses by chance the randomly smaller backoff timer than S_2, sender node S_1 transmits three short preambles at the expiration of its backoff timer. As the sender node S_2 detects the medium busy at the expiration of its backoff timer, so it holds to send its short preamble to the next cycle. In the next duty cycle, sender node S_2 restarts its BEB algorithm. As the sender node S_2 is the only sender as shown in Figure 2, so sender node S_2 sends two short preambles at the expiration of its backoff timer. As for LCX-MAC, after exchanging wakeup information between the sender and the corresponding receiver, the sender sends only one short preamble, which results in saving energy and reducing delay.
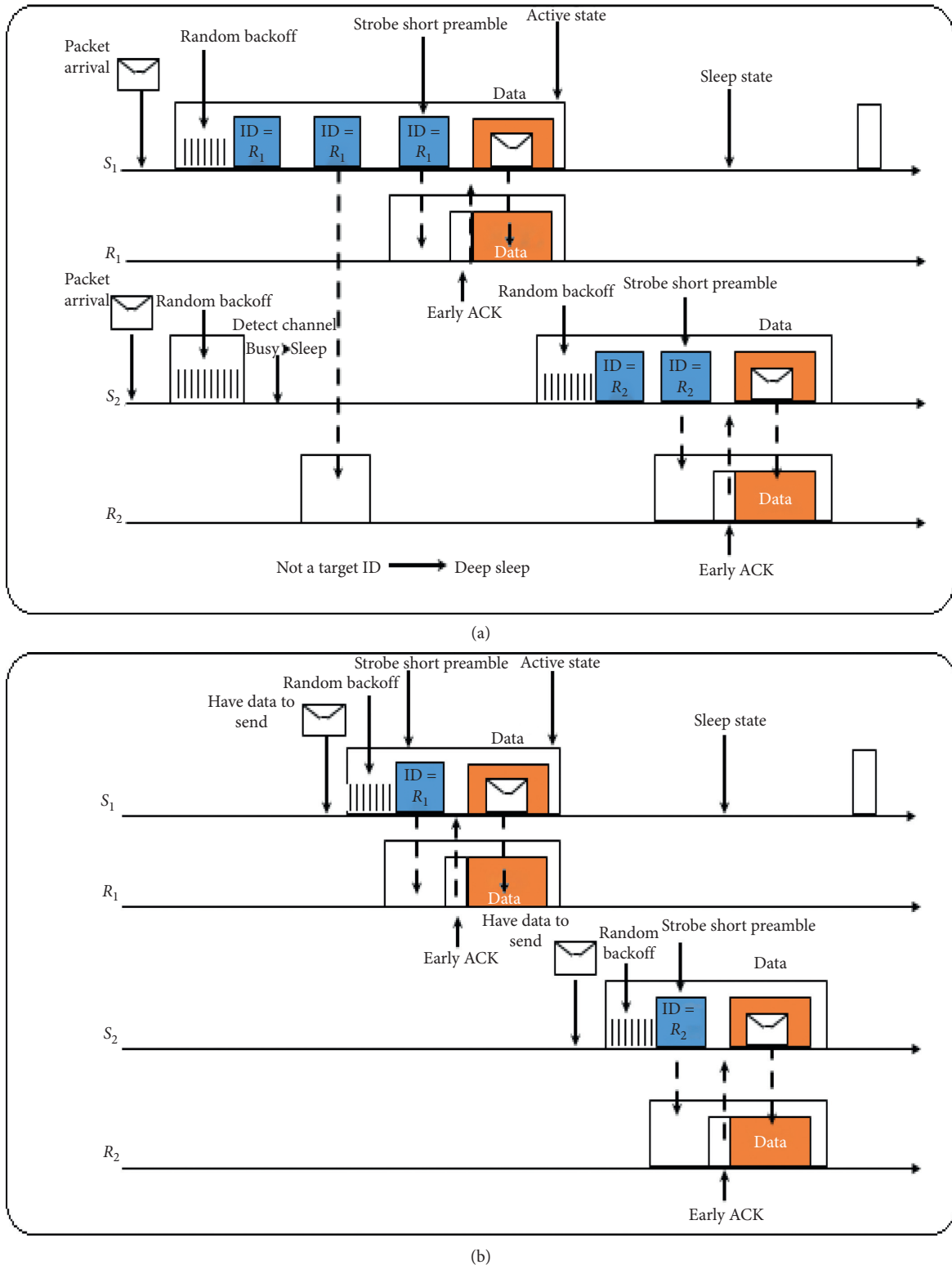
(a)

(b)

FIGURE 2: Comparing (a) X-MAC/BEB and (b) LCX-MAC in time domain.

4. Markov Model for Duty Cycle MAC Protocol

To investigate the performance of the MAC protocols with duty cycle behaviour, the authors [23] proposed a general mechanism consisting of two models: the first model for the duty cycle behaviour and the second model for the behaviour specific to any individual MAC protocol. The first Markov model is built according to the number of data frames stored in the buffer that reaches from the upper layer or set out to the underlying channel. The second model is built according to

the medium access behaviour of an individual MAC protocol to send data frames. Using these two models, we will get two broad equations with two unknown variables.

The first Markov model of duty cycle MAC protocols [23] describing transitions among states of a node is shown in Figure 3. A node's buffer state is represented as a number in a circle that denotes the number of frames buffered in the queue in MAC protocols. Events shown by arrows correspond to either a frame's departure to the link or a frame's arrival from the upper layer, respectively. In all the allowed transitions, this model makes some assumptions that only one frame will be transmitted in one cycle, while more than one frame can arrive within one cycle. Table 1 contains all the important notations.

The Markov model of a node queue has three types of probabilities. Probability of generating i-frame in a cycle is A_i, stationary probability of empty queue state is π_0, and a node can only transmit one data frame per cycle with probability p. Equation (1) is a set of equations, expressing $P_{i,j}$, a state transition probability, with A_i and p, where i and j are the number of frames in node queue in the earlier cycle and the number of frames in the present cycle, respectively. The first two equations in (1) described the transition from an empty queue to nonempty queue depending on the number of frames i taking place, where i-frames arrive in one cycle. For more details, refer to [23].

$$
\begin{cases}
P_{0,i} = A_i, & i = 0, 1, \ldots, Q-1, \\[2mm]
P_{0,Q} = A_{\geq Q}, & \\[2mm]
P_{i,i-1} = p \cdot A_0, & i = 1, \ldots, Q, \\[2mm]
P_{i,j} = p \cdot A_{j-i+1} + (1-p) \cdot A_{j-1}, & i = 1, \ldots, Q,\ j = i, \ldots, Q-1, \\[2mm]
P_{i,Q} = p \cdot A_{\geq Q-i+1} + (1-p) \cdot A_{\geq Q-i}, & i = 1, \ldots, Q, \\[2mm]
P_{i,j} = 0, & i = 2, \ldots, Q,\ j = 0, \ldots, i-2.
\end{cases}
\tag{1}
$$

The performance model in [23] builds (2) and (3) by relating $\pi = (\pi_0, \pi_1, \pi_2, \ldots, \pi_Q)$, the unique stationary distribution, and P the transition matrix. In (2) π_i is the stationary probability that a node has i-frames.

$$
\sum_{s_i \in S} \pi_i = 1.
\tag{2}
$$

Equation (3) shows a condition of stable node. By knowing the data frame arrival information, we can combine three equations, (1)–(3), to remove $P_{i,j}$ and π_i by expressing only with A_i and p.

$$
\pi P = \pi.
\tag{3}
$$

If the data frame arrival rate is known, then the equation for A_i, as with the Poisson process (4), is used to express π_0 with a function of p [23].

$$
\pi_0 = f(p).
\tag{4}
$$

In (4), p is unknown variable for we required another function given in (5) which relates p to π_0. For (5), the second model for a MAC protocol is needed since p depends on the medium access behaviour. In the next section, we derived (5) involving p and π_0 according to the behaviour of LCX-MAC.

$$
p = f(\pi_0).
\tag{5}
$$

5. Statistical Analysis

5.1. Throughput of LCX-MAC. To calculate the throughput of the LCX-MAC, (6), which is used in [23], is the ratio of the total data successfully transmitted in one cycle T to total cycle length T.

In (6), S is the size of a data frame, π_0 is the probability where a node have no frame in the queue, and P_{Succ} is for each node successfully transmitting a data frame.

$$
\text{THR} = \frac{N \cdot (1 - \pi_0) \cdot P_{\text{Succ}} \cdot S}{T}.
\tag{6}
$$

The only difference between X-MAC/BEB and LCX-MAC is that X-MAC/BEB on average sends $T/2$ time of the one cycle short preamble, while LCX-MAC only sends one short preamble.

Equation for X-MAC/BEB is as follows:

$$
\begin{aligned}
E_{\text{busy}}(W_k) = \sum_{n=0}^{\infty} \sum_{t_{\text{fr}}=0}^{T-1} &\left(\left(\frac{T}{2} + t_{\text{Data}} \right) \right. \\
&\left. \cdot P_{\text{busy}}^{\text{succ}}(nT, t_{\text{fr}}, W_k) + T \cdot P_{\text{busy}}^{\text{coll}}(nT, n_{\text{fr}}, W_k) \right).
\end{aligned}
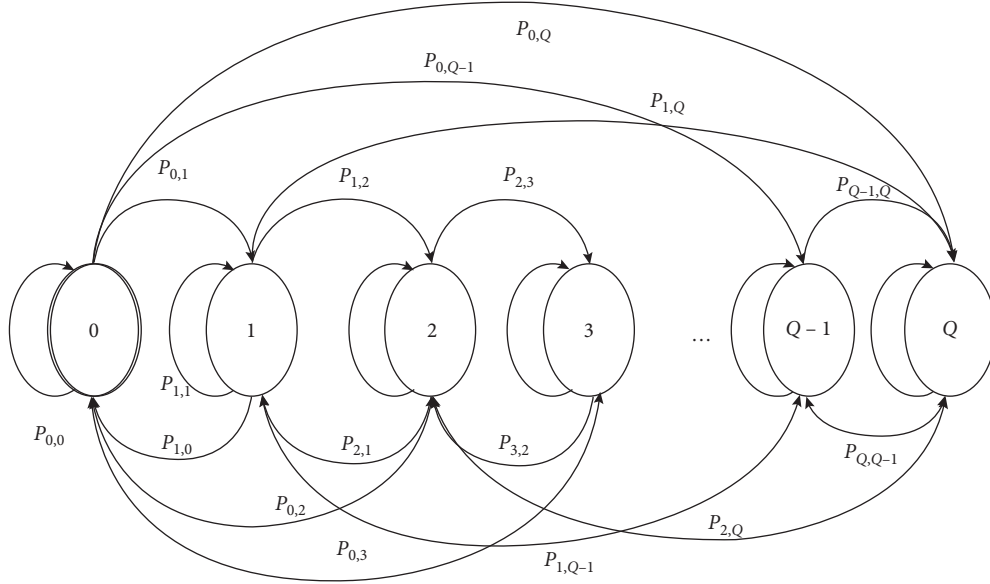\tag{7}
$$

Equation for LCX-MAC is as follows:

FIGURE 3: Markov model of a node queue with duty cycle.

TABLE 1: Notations.

Symbol	Description
N	Number of nodes
N_{ac}	Nodes wake up during T_a
n_i	Nodes wake up at the ith time slot
a_i	Nodes have data to send
T	One cycle length
T_a	Time nodes wakeup can affect transmission at t
T_{un}	Time nodes cannot affect transmission at t
Q	Queue size
S	One data frame size
λ	Arrival rate of data frame
W_0	Initial CW size
W_m	Maximum CW size
m	Maximum backoff stages
τ	One time-slot duration
π_0	Empty queue stationary probability
p	Node transmission probability $p = P_{Succ} + P_{Coll}$
π_i	State I stationary probability
A_k	k frames are created in time T with probability $A_k = (e^{-\lambda T}(\lambda T)^k/k!)$
$A_{\geq k}$	No less than k frames are created in time T with probability $A_{\geq k} = 1 - \sum_{i=0}^{k-1} A_i$

TABLE 2: Network parameters.

Symbol	Value
Bitrate	250 kbps
T	20 μs
Λ	1 frame/s
T_{Active}	15 ms
t_{ACK}	1 ms
t_{pre}	3 ms
t_{DATA}	5 ms
T	50–300 ms
Q	10
S	50 bytes
Txp	59.1 mW
Rxp	52.2 mW

$$E_{busy}(W_k) = \sum_{n=0}^{\infty} \sum_{t_{fr}=0}^{T-1} \left(\left(t_{pre} + t_{Data} \right) \right.$$
$$\left. \cdot P_{busy}^{succ}(nT, t_{fr}, W_k) + T \cdot P_{busy}^{coll}(nT, n_{fr}, W_k) \right). \tag{8}$$

5.2. Delay of LCX-MAC. Equation (9) is used to calculate the time a frame generation at sender to its successful transfer at receiver is divided in three different types of delays, D_Q queuing delay, D_{CW} random backoff delay, and D_C contention delay.

Equation for X-MAC/BEB is as follows:

$$D = D_Q + D_{CW} + D_C. \tag{9}$$

Equation for LCX-MAC is as follows:

$$D = D_Q + D_{CW}. \tag{10}$$

As in LCX-MAC, the sender node adjusts its wakeup time to its corresponding receiver, and there is no contention delay.

5.3. Power Consumption of LCX-MAC. Equation (11) is used to find the total energy used per cycle by each node and can be expressed as the addition of energies a node can spend at five different states. E_1 is a node that successfully transmits a data frame, E_2 is a node that successfully receives a data frame, E_3 is a node that has collision when transmitting a data frame, E_4 is a node that is a potential receiver to a data frame but failed because of
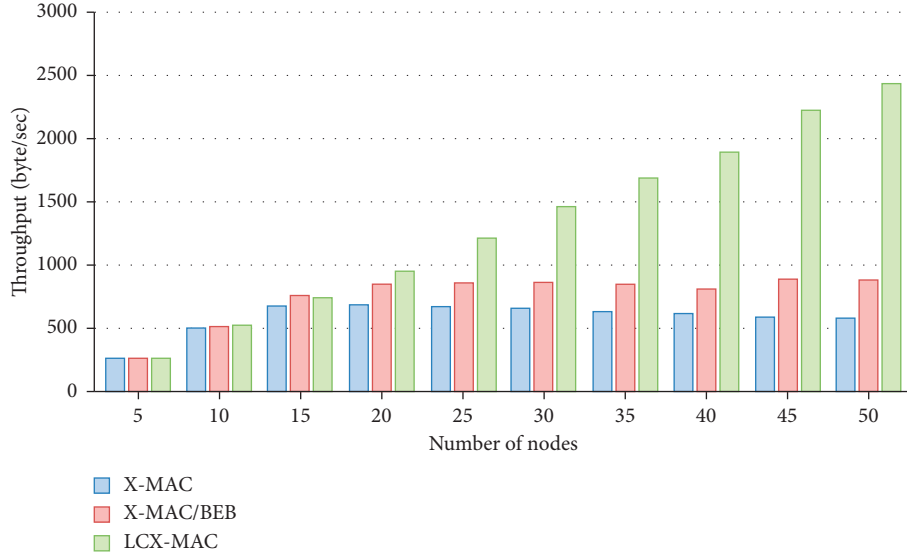
FIGURE 4: Throughput (bytes/sec) versus number of nodes. Comparing X-MAC, X-MAC/BEB, and LCX-MAC throughout.

collision, and, finally, E_5 stays awake for the entire active time. As the receiver node of LCX-MAC works as X-MAC, so the states E_2, E_4, and E_5 have the same energy consumptions from [23].

$$E = E_1 + E_2 + E_3 + E_4 + E_5. \tag{11}$$

The only difference between X-MAC/BEB and LCX-MAC that X-MAC/BEB on average $T/2$ time of the one cycle energy is consumed in sending short preamble and receiving early ACK, while LCX-MAC only sends one short preamble. Equation for X-MAC/BEB is as follows:

$$E_1 = (1 - \pi_0) \cdot P_{\text{Succ}}(W_k) \cdot \tau \cdot \left(\left(\frac{1}{m+1} \sum_{k=0}^{m} \left(\frac{W_k}{2} \right) \right) \cdot \text{rxp} + \frac{T}{2} \cdot \left(\frac{t_{\text{pre}}}{(t_{\text{pre}} + t_{\text{ACK}})} \right) \cdot \text{txp} + \frac{T}{2} \cdot \left(\frac{t_{\text{ACK}}}{(t_{\text{pre}} + t_{\text{ACK}})} \right) \cdot \text{rxp} + t_{\text{DATA}} \cdot \text{txp} \right). \tag{12}$$

Equation for LCX-MAC is as follows:

$$E_1 = (1 - \pi_0) \cdot P_{\text{Succ}}(W_k) \cdot \tau \cdot \left(\left(\frac{1}{m+1} \sum_{k=0}^{m} \left(\frac{W_k}{2} \right) \right) \cdot \text{rxp} + \left(\frac{t_{\text{pre}}}{(t_{\text{pre}} + t_{\text{ACK}})} \right) \cdot \text{txp} + \left(\frac{t_{\text{ACK}}}{(t_{\text{pre}} + t_{\text{ACK}})} \right) \cdot \text{rxp} + t_{\text{DATA}} \cdot \text{txp} \right). \tag{13}$$

6. Evaluations

This section evaluates the throughput, delay, and per-frame energy consumption of LCX-MAC, X-MAC/BEB, and X-MAC by their analytical models. Table 2 contains the important parameters and values used in the mathematical calculations.

Figure 4 shows that LCX-MAC, represented by green bars, outperforms X-MAC/BEB shown by red bars and X-MAC as indicated by blue bars as the number of nodes increased.

Figure 5 shows that the delay of LCX-MAC is minimal because the only small amount of time is used to send one short preamble. In contrast, the X-MAC/BEB and X-MAC send on average half of cycle ($T/2$) time short preamble; that is why the delay of X-MAC/BEB and X-MAC went up when the number of nodes increased.

Figure 6 shows that $E_{\text{frame/node/s}}$ of LCX-MAC, X-MAC/ BEB, and X-MAC falls as the number of nodes in the network increased because nodes have less chance to send data frames due to busy medium, leading to termination of data transmission, so the nodes stay in sleep mode most of the time.
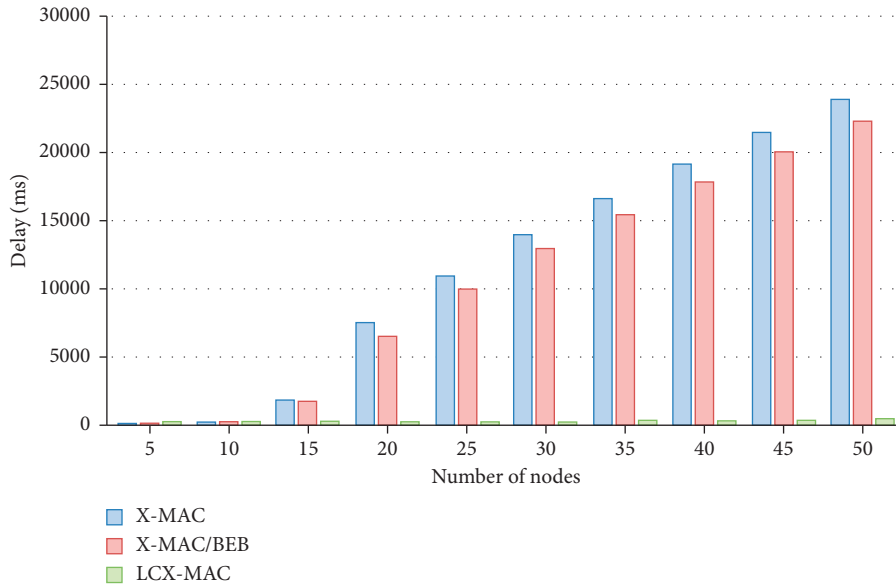
FIGURE 5: Delay (ms) versus number of nodes. Comparing X-MAC, X-MAC/BEB, and LCX-MAC delay.
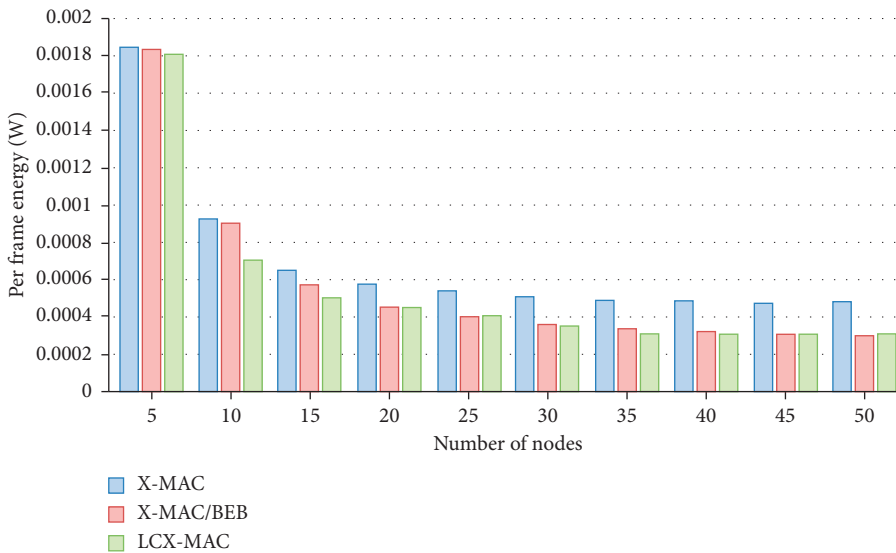

FIGURE 6: Energy ($E_{\text{frame/node/s}}$) versus number of nodes. Comparing X-MAC, X-MAC/BEB, and LCX-MAC per frame energy.

7. Conclusions and Future Work

We proposed LCX-MAC for IoT healthcare and WSNs, in which the sender nodes coordinate their wakeup time with their corresponding receivers to increase throughput, decrease delay, and reduce energy consumptions. On average, in half of the cycle of X-MAC and X-MAC/BEB, each node sends short preambles and listens for early ACK before sending the actual data. We also extend the model of X-MAC/BEB. The results show that LCX-MAC increases the performance of X-MAC/BEB when nodes in IoT and WSNs become populated. The throughput of LCX-MAC is increased by 130% compared to the throughput of X-MAC/BEB for the number of nodes 40. LCX-MAC is a kind of synchronous MAC protocol with much low overhead compared to the synchronous MAC protocols which gave minimal delay so that we will compare LCX-MAC with synchronous MAC protocols in our future works. The energy consumption of X-MAC/BEB and LCX-MAC is decreased by 40% compared to the energy consumption of X-MAC.

This paper presents single hope network topology. In the future, we will extend to multihop network topology as the nodes in the IoT environments are multihop. We will also do detail simulation works to validate the analytical model results in network simulator version 2 (ns2) [24].

Data Availability

The data used to support the findings of this study are included within the manuscript.

Conflicts of Interest

Acknowledgments

References

[1] L. Fuhong, L. Qian, Z. Xianwei, C. Yueyun, and H. Daochao, "Cooperative differential game for model energy-bandwidth efficiency tradeoff in the internet of things," *China Communications*, vol. 11, no. 1, pp. 92–102, 2014.

[2] S. Huang, A. Liu, S. Zhang, T. Wang, and N. Xiong, "BD-VTE: a novel baseline data based verifiable trust evaluation scheme for smart network systems," *IEEE Transactions on Network Science and Engineering*, 2020.

[3] Y. Wei, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '02)*, pp. 1567–1576, New York, NY, USA, June 2002.

[4] T. Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 21st Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE INFOCOM '02)*, pp. 171–180, Los Angeles, CA, USA, November 2003.

[5] S. Du, K. Amit, and D. B. Johnson, "RMAC: a routing-enhanced duty-cycle MAC protocol for wireless sensor networks," in *Proceedings of the 26th Annual IEEE Conference on Computer Communications (INFOCOM 2007)*, pp. 1478–1486, Anchorage, AK, USA, May 2007.

[6] Y. Sun, S. Du, O. Gurewitz, and D. B. Johnson, "DW-MAC: a low latency, energy-efficient demand-wakeup MAC protocol for wireless sensor networks," in *Proceedings of the Ninth ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc 2008)*, pp. 53–62, Hong Kong, China, May 2008.

[7] W. Ye, F. Silva, and J. Heidemann, "Ultra-low duty cycle mac with scheduled channel polling," in *Proceedings of the 4th ACM International Conference on Embedded Networked Sensor Systems (SenSys '06)*, pp. 321–333, Boulder, CO, USA, November 2006.

[8] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 95–107, Baltimore, MD, USA, November 2004.

[9] Amre El-Hoiydi and J.-D. Decotignie, "WiseMAC: an ultra low power MAC protocol for multi-hop wireless sensor networks," in *Proceedings of the First International Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS 2004)*, pp. 18–31, Lecture Notes in Computer Science LNCS 3121, Turku, Finland, July 2004.

[10] M. Buettner, G. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems (SenSys '06)*, pp. 307–320, Boulder, CO, USA, November 2006.

[11] A. Bhatia and R. C. Hansdah, "A distributed TDMA slot scheduling algorithm for spatially correlated contention in WSNs," in *Proceedings of the IEEE International Conference on Advanced Information Networking and Applications Workshops (AINA)*, pp. 377–384, Barcelona, Spain, March 2013.

[12] A. Ullah, G. Kim, and J. S. Ahn, "Performance analysis of X-MAC protocol with collision avoidance algorithm," in *Proceedings of the 32nd Annual IEEE Conference on Computer Communications (INFOCOM 2013)*, pp. 164–212, Student Poster Session, Turin, Italy, April 2013.

[13] A. Ullah and J. S. Ahn, "Performance evaluation of X-MAC/ BEB protocol for wireless sensor networks," *Journal of Communications and Networks (JCN)*, vol. 18, no. 5, pp. 857–869, 2016.

[14] G. Kim, J.-G. Kang, and M. Rim, "Dynamic duty-cycle MAC protocol for IoT environments and wireless sensor networks," *MDPI Energies*, vol. 12, no. 21, pp. 1–13, 2019.

[15] A. B. Nacef, S. M. Senouci, Y. Ghamri-Doudane, and A. L. Beylot, "A cooperative low power mac protocol for WSNs," in *Proceedings of the IEEE International Conference on Communications ICC 2011*, Kyoto, Japan, June 2011.

[16] S.-h. Hong and H.-k. Kim, "A multi-hop reservation method for end-to-end latency performance improvement in asynchronous MAC-based wireless sensor networks," *IEEE Transactions on Consumer Electronics*, vol. 55, no. 3, pp. 1214–1220, 2009.

[17] J.-H. Lee, J.-W. Kim, and D.-S. Eom, "A delay-tolerant virtual tunnel scheme for asynchronous MAC protocols in WSN," *Wireless Personal Communications*, vol. 70, no. 2, pp. 657–675, 2013.

[18] I. Park, H. Lee, S. Kang, and D.-S. Eom, "RIX-MAC: an energy-efficient receiver-initiated wakeup MAC protocol for WSNs," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 5, pp. 1604–1617, 2014.

[19] I. Park, J. Yi, and H. Lee, "A receiver-initiated MAC protocol for wireless sensor networks based on tree topology," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 950656, 10 pages, 2015.

[20] G. Corbellini, E. C. Strinati, and A. Duda, "LA-MAC: low-latency asynchronous MAC for wireless sensor networks," in *Proceedings of the IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications*, pp. 380–386, Sydney, Australia, September 2012.

[21] Y. Liu, K. Ota, K. Zhang et al., "QTSAC: an energy-efficient MAC protocol for delay minimization in wireless sensor networks," *IEEE Access*, vol. 6, no. 1, pp. 8273–8291, 2018.

[22] X. Liu, A. Liu, T. Wang et al., "Adaptive data and verified message disjoint security routing for gathering big data in energy harvesting networks," *Journal of Parallel and Distributed Computing*, vol. 135, pp. 140–155, 2020.

[23] O. Yang and W. Heinzelman, "Modeling and performance analysis for duty-cycled MAC protocols with applications to S-MAC and X-MAC," *IEEE Transactions on Mobile Computing*, vol. 11, no. 6, pp. 905–921, 2012.

[24] Network Simulator, ns-2, http://www.isi.edu/nsnam/ns/.