KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS VOL. 8, NO. 5, May. 2014

RIX-MAC: An Energy-Efficient Receiver-Initiated Wakeup MAC Protocol for WSNs

Inhye Park[1], Hyungkeun Lee[1] and Seokjoong Kang[2]
[1]Department of Computer Engineering
Kwangwoon University
Seoul, Korea
[2]Department of Management of Technology for Defense
Korea University
Seoul, Korea
[e-mail: hklee@kw.ac.kr, alwaysin@kw.ac.kr, sjkang64@korea.ac.kr]
*Corresponding author: Seokjoong Kang

Received December 10, 2013; revised February 13, 2014; accepted March 21, 2014; published May 29

Abstract

This paper proposes RIX-MAC (Receiver-Initiated X-MAC), a new energy-efficient MAC protocol based on an asynchronous duty cycling. RIX-MAC improves energy efficiency through utilizing short preambles and adopting the receiver-initiated approach, where RIX-MAC minimizes sender nodes' energy consumption by enabling transmitters to predict receiver nodes' wake-up times. It also reduces receiver nodes' energy consumption by decreasing the number of control frames. We use the network simulator to evaluate RIX-MAC's performance. Compared to the prior asynchronous duty cycling approaches of X-MAC and PW-MAC, the proposed protocol shows a remarkable improvement in energy-efficiency and end-to-end delay.

Keywords: wireless sensor network, medium access control, asynchronous duty cycling, energy-efficiency MAC

The research has been conducted by the research grant of Kwangwoon university in 2013.

http://dx.doi.org/10.3837/tiis.2014.05.006

1. Introduction

Unlike other wireless network such as WLAN and Bluetooth, sensor nodes in WSN (wireless sensor network) can't change or charge their own batteries. Therefore, energy-efficiency is a fundamental and important consideration when designing a MAC protocol for sensor nodes in WSN. To reduce energy consumption in WSN, many researches have been studied ever since. The most popular mechanism for reducing energy consumption in WSN is duty cycling. The duty cycling technique saves energy by switching sensor nodes between awake and sleeping states [1]. Existing duty cycling energy-efficient MAC protocols can be categorized into two types: synchronous and asynchronous protocols.

In the synchronous MAC protocols such as S-MAC [2] and T-MAC [3], the awake duration in duty cycle of each node is synchronized by exchanging periodical control frame such as SYNC. These additional control frames cause wasted energy usage. Also, overlapped common period for accessing medium causes collision between frames. In contrast, asynchronous MAC protocols, such as B-MAC [4], X-MAC [1], and RI-MAC [5], have higher energy efficiency than synchronous protocols. These protocols are using the preamble sampling, instead of additional synchronized control frames. Because of the small number of control frames and low likelihood of collision, asynchronous MAC protocols are more effective in terms of energy consumption. It is quite difficult to match awake state between the sender and receiver nodes, even though asynchronous MAC protocols are designed carefully to operate not to waste energy.

In this paper, we present an energy-efficient receiver-initiated MAC protocol called RIX-MAC (receiver-initiated X-MAC). The operation of RIX-MAC is substantially similar to one of X-MAC, where the proposed protocol utilizes the receiver-initiated wakeup scheme as described in RI-MAC. RIX-MAC reduces the number of overall control frames of the sender and receiver as well as it partially prevents the collision in some situation, such as the case of multiple senders and one-receiver. As a result, RIX-MAC can save the energy consumption rather than previous MAC protocols, such as X-MAC and PW-MAC [6].

This new MAC protocol, RIX-MAC, saves sender's energy consumption through receiver-initiated double wakeup scheduling to reduce the number of senders' control frames during a period between sender's and receiver's wakeups. It also saves receiver's energy because the sender and the receiver transmit control frames only when data exists in the sender. In the case of multiple senders, RIX-MAC achieves higher channel utilization by random backoff. We present the performance evaluation results of RIX-MAC by simulations in ns-2 compared with other energy-efficient MAC protocols.

The rest of this paper is organized as follow. Section 2 describes related work on duty cycling MAC protocols for WSN. Section 3 presents the design of the RIX-MAC protocol, and section 4 presents its performance evaluation on ns-2 [7] simulator. Finally, concluding remarks will be given in section 5.

2. Related Work

2.1 Synchronized MAC protocol using random backoff

S-MAC and T-MAC are RTS-CTS (ready-to-send, clear-to-send) based MAC protocols that make use of loose synchronization between nodes for duty cycling in WSN. The protocols use three techniques to achieve low power duty cycling: periodic sleep, virtual clustering and adaptive listening. The nodes in the network wakeup periodically, receive and transmit data, and return to sleep. At the beginning of awake periods, a node exchanges synchronization and schedule information with its neighbors to assure that the node and its neighbors wakeup concurrently. After the synchronization information is exchanged, the node transmits or receives frames using RTS-CTS until the end of awake periods and the node then enters the sleep mode. The biggest difference between S-MAC and T-MAC is that T-MAC improves the performance of S-MAC by shortening awake periods when the channel is idle.

Aside from these protocols, many synchronous MAC protocols have been studied. Among them, DPSMAC [8] proposes a differential probability selection MAC protocol which is efficient for medium access in real-time sensor networks, and DT-MAC [9] employs a dynamic threshold for the buffer in sensor nodes to maximize energy efficiency regardless of the network traffic condition. LO-MAC [10][11] proposes a protocol employing duty cycing and multiple fowarding in order to reduce idle listening and sleep latency, respectively.

Synchronous MAC protocols operate in a simple manner. However, maintaining the synchronous state of nodes in the network is not quite easy. Additional control frames in synchronous MAC protocols also cause the waste of energy at the nodes, and overlapped common periods for accessing medium increase the possibility of collisions in transmitting frames.

2.2 Asynchronous MAC protocols using preambles

B-MAC is a CSMA-based protocol that utilizes low power listening and an extended preamble to achieve low energy consumption. Nodes in the network have two periods, an awake and a sleep periods, and each node can have an independent schedule for duty cycling. If a node wishes to transmit, a data frame is preceeded by a preamble that is slightly longer than the sleep period of receiver. During an awake period, a node samples the medium and it remains the awake state to receive data if a preamble is detected. With the preamble ended, the sender transmits a data frame to the receiver. Even though B-MAC improves the energy efficiency over synchronous MAC protocols, it has the problem of wasting transmission energy due to long preambles.

On the other hand, X-MAC which is based on B-MAC uses the preamble sampling mechanism to achieve low power communication without synchronization. The key features are the short preamble sampling and the *early-ACK* mechanism. X-MAC transmits multiple short preambles to make receiver notice instead of a long preamble at B-MAC. If the receiver detects a short preamble during its awake period, it transmits an *early-ACK* frame to the sender before receiving the actual data frame, and the sender knows that the receiver stays in the awake period, and it transmits data frame immediately [11]. Consequently, X-MAC saves more energy than B-MAC due to the short preamble sampling and the *early-ACK* mechanism.

Besides these protocols, many asynchronous MAC protocols have been researched. WTE-MAC [12], EP-MAC [13], and BoostMAC [14] improve the end-to-end delay of X-MAC protocol through virtual tunneling procedure, strobed preamble concept, and dynamic channel polling time, respectively.

2.3 Asynchronous MAC protocol using receiver- Initiated features

RI-MAC employs a receiver-initiated wakeup approach from WiseMAC [15], in which wakeup beacons generated by the receiver are used to reduce energy waste of the sender. This protocol increases channel utilization and enables more efficient collision detection. In RI-MAC, each node announces its wakeup information with a beacon, and a sender starts data transmission upon receiving a beacon from its intended receiver. However, when a sender has a frame to send, it immediately wakes up to wait for the receiver leading to a large sender duty cycle due to its idle listening until the receiver wakes up.

Compared with RI-MAC, PW-MAC achieves near-optimal duty cycle at receivers and at senders. To enable a sender to accurately predict the wakeup times of a receiver, it requires every node in PW-MAC to compute its wakeup times using its pseudo-random wakeup schedule generator, and calculates the hardware-level delay, operating system delay and clock drift and so on. Therefore, PW-MAC reduces energy consumption of sensor nodes than RI-MAC by enabling sensors to predict receiver wakeup times.

In addition to these protocols, many receiver-initiated MAC protocols have been researched. EM-MAC [16] ehnanceds PW-MAC by exploiting an exponential chase algorithm for an efficient resynchronization over the clock drift. In addition, EM-MAC spreads the traffic load to different channels to minimize the collisions among multiple senders. In [17], τ-duration CCA is utilized to mitigate the collision problem in ACK-based LPL MAC protocols and short-preamble counter is used to conserve more energy by reducing unnecessary overhearing.

The key idea of these receiver-initiated MAC protocols is very effective and powerful in terms of energy consumption. But these protocols have critical disadvantage when there exist multiple senders and one-receiver. In the case of multiple senders which are ready to send data frames, collisions may occur with very high probability because senders wake up at the time that the receiver initiates. Our proposed new protocol partially solves its problem using random backoff procedure.

3. Receiver-initiated X-MAC (RIX-MAC) protocol

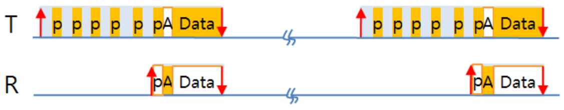
3.1 Overview

RIX-MAC is a contention-based MAC protocol with duty cycling, which employs clear channel assessment (CCA) [18] and low power listening (LPL) [19] for accessing channel. Every node in the network has the same operational cycle which is divided into two states: the wakeup and sleep states. The wakeup state is an active state for a node to receive or transmit data with the radio turned on, and the sleep state is for a node to save its power by turning off the radio. The wakeup state is divided into two periods: the Sched-wakeup (scheduled wakeup) and Synch-wakeup (synchronized wakeup) periods, where the Synch-wakeup period is optional when a node wants to transmit data frames.

Because RIX-MAC's operation is basically similar to one of X-MAC, a sender (TX-node) of RIX-MAC transmits short preambles during a Synch-wakeup period. In **Fig. 1**, the basic operations of X-MAC, PW-MAC and RIX-MAC are shown, where RIX-MAC transmits short preambles as X-MAC does and wakes up twice as PW-MAC does. The details of X-MAC and PW-MAC were described in previous sections. **Fig. 2** shows flow charts of internal operation of RIX-MAC at TX-node and RX-node in **Fig. 2**(a) and **Fig. 2**(b), respectively. RIX-MAC has following advantages.

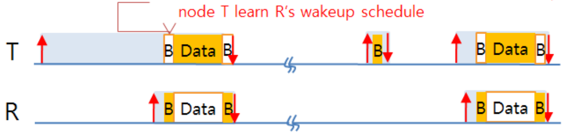
- X-MAC avoids the collision via other transmitter's frame. Under the situation of multiple senders, TX-nodes could prevent data collision by listening others' short preambles and the *early-ACK* frame of a receiver (RX-node).
- PW-MAC reduces the TX-node's power consumption through predicting the RX-node's wakeup schedule.

In the rest of this section, we will describe the proposed protocol's operational mechanism, the features for collision avoidance and additional advantages of our proposed protocol in detail.



(a) X-MAC

When the node T receive beacon frame from node R,
node T learn R's wakeup schedule



(b) PW-MAC

When the node T receive early-ACK frame from node R,
node T learn R's wakeup schedule

Insert the R's schedule

(c) RIX-MAC

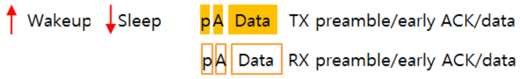


Fig. 1. Concepts of X-MAC, PW-MAC, and RIX-MAC

3.2 Additional fields for RIX-MAC

Two more fields are added in the control frames for operation of the proposed MAC protocol: wakeup-time and duration fields. The wakeup-time field is used to predict the next Sched-wakeup of RX-node. RX-node inserts the wakeup-time field into *early-ACK* frame. The field contains the time information which tells how much of time in *ms* to elapse up to RX-node's Sched-wakeup. With this field, TX-node could predict Sched-wakeup of RX-node because it is assumed that every node in the network has the same duty-cycle ratio.

The second field, duration, is used for virtual carrier sensing by setting the network allocation vector (NAV) of *involved neighbors*. When a TX-node wants to transmit data, another TX-node called an *involved neighbor* might be ready to send data, which is the situation of multiple senders. The duration field is inserted to control frames such as *short preamble* and *early-ACK*. This field indicates information about the transmission time of data frames. RX-node gets the value of the duration field in a short preamble and copies it to the corresponding *early-ACK* frame. The detailed exploitation of the fields with an *involved neighbor* and RX/TX-nodes will be described in the section 3.4.

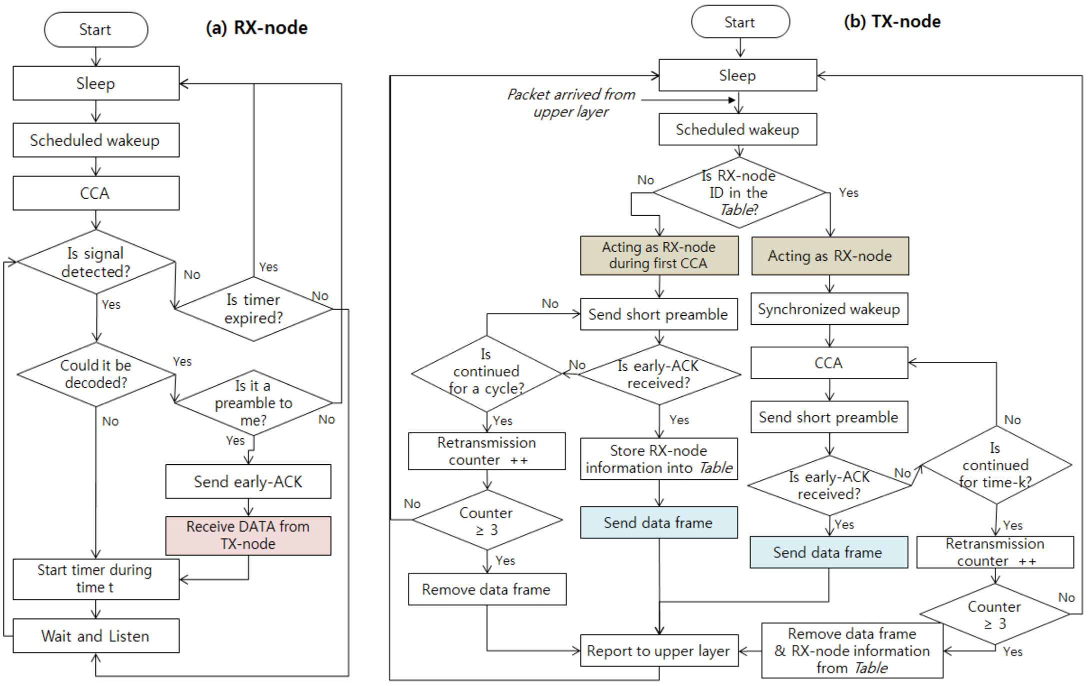


Fig. 2. Flow chars of internal operation at (a) RX-node and (b) TX-node

3.3 Receiver-Initiated operations of RIX-MAC

In RIX-MAC, all nodes have their own wakeup schedule independently. However, TX-node has not only its own schedule but also the temporary schedule of RX-node, where RX-node's schedule can be obtained by the wakeup-time field of *early-ACK* frames. TX-node in RIX-MAC wakes up twice in one cycle only when the TX-node knows the RX-node's wakeup schedule. In TX-node, the first wakeup occurs at its own schedule to receive data frames (Sched-wakeup) and the second one is to transmit data to RX-node (Synch-wakeup).

Since TX-node does not know the RX-node's schedule in the beginning of transmission, RIX-MAC operates just like X-MAC. When data to be transmitted arrive at the MAC layer, TX-node searches a schedule of designated RX-node in the wakeup-time table. If it cannot find the schedule, TX-node starts transmitting short preambles at its own schedule as X-MAC does. Upon receiving the *early-ACK* frame from RX-node, TX-node starts to transmit data frames immediately, obtains the exact timing information about RX-node's wakeup with the wakeup-time field from RX-node, and saves information on the wakeup-time table as a time gap between Sched-wakeup and Synch-wakeup.

Table 1 shows an example of the wakeup-time table with which TX-node is able to predict the wakeup time of RX-node. Then TX-node transmits short preambles to inform RX-node data transmission just before RX-node's wakeup.

Table 1. An example of wakeup-time table (one operational cycle: 1.5s)

Node ID	Time-gap (unit: ms)
3	40
8	1100
17	600

When TX-node predicts wakeup of designated RX-node, *data arrival time* means when data is ready to be transmitted at the MAC layer. Therefore, Synch-wakeup is determined by time-gap and *data arrival time*. An example of Sched-wakeup, Synch-wakeup and SLEEP periods with *data arrival time* is shown in **Fig. 3**. To explain it in detail, we denote T_{TD} as time that how much time has passed from last Sched-wakeup and T_{TR_r} as time-gap of designated RX-node, r, in *ms* unit. When data arrives at the MAC layer in TX-node, how much time remains to Synch-wakeup of r is denoted as T_{DR_r}. When data arrive from the upper layer, T_{TD} means time between TX-node's last Sched-wakeup and *data arrival time*. If data is ready after T_{TR_r} from the last Sched-wakeup, Synch-wakeup period would be started at the next cycle.

$$T_{DR_r} = \begin{cases} T_{TR_r} - T_{TD} & if\ \ T_{TR_r} - T_{TD} \geq 0, \\ T_{TR_r} - T_{TD} + C & else \end{cases}$$

(C is cycle)

For example, TX-node with the wake-up table of **Table 1** wants to transmit data to the node with 17 of ID as shown in **Fig. 3**, where time gap between Sched-wakeup and Synch-wakeup is 600*ms*. Since data arrive within 600ms after last Sched-wakeup, TX-node starts the Synch-wakeup period at the right away after $T_{TR_r} - T_{TD}$.

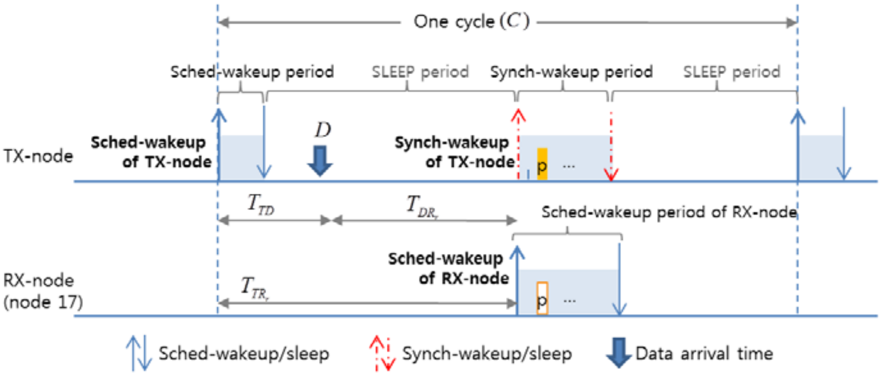


Fig. 3. Example using Time gap in wakeup-time table

RIX-MAC increases the energy-efficiency by reducing overhead, that is, the number of control frames compared to X-MAC and PW-MAC. In X-MAC, TX-node has to transmit a lot of short preambles to deliver a data frame. In PW-MAC, RX-node wastes control frames such as beacons because of unnecessary announcement even when there is no data to receive. In RIX-MAC, however, TX-node transmits control frames to RX-node only when required. Our proposed protocol, RIX-MAC, is an efficient protocol in terms of energy consumption.

3.4 Collision Avoidance and backoff in RIX-MAC

The operation of RIX-MAC is based on X-MAC's, and it adopts the feature of receiver-initiated wakeup from PW-MAC. In addition to energy efficiency, RIX-MAC has been designed to solve the problem caused by the case of multiple active senders which is frequently taking place in wireless sensor networks.

Because TX-node attempts to transmit on the RX-node's wakeup, receiver-initiated protocols might induce a collision problem when several nodes try to deliver data to a RX-node. RIX-MAC can solve this collision problem partially at some cases. Unlike X-MAC and PW-MAC, our proposed protocol, RIX-MAC, utilizes backoff procedure in order to avoid collision by multiple senders.

After Synch-wakeup, TX-node starts the random backoff procedure after clear CCA. Because the number of slots is randomly selected, the timer generates a range of backoff delay with *RD(i)*SlotTime*, where *i* is the power number of 2. In our simulation explained in the next section, we set *i* to 4 and the range of backoff is [0, 15]. If the medium is idle during CCA, RIX-MAC decreases the backoff timer by one *SlotTime* until the timer becomes to zero. On the other hand, if TX-node detects a signal during CCA or backoff, the backoff timer stops to count down. When the backoff timer pauses, TX-node set NAV during the period of other TX-node's transmission.

TX-node starts to decrease the backoff timer again after NAV. After the backoff timer becomes zero, TX-node then starts transmission of short preambles.

We suppose that there are several senders to transmit data to a RX-node. If one sender that selects the smallest backoff delay has a chance to transmit short preambles, other nodes become *involved neighbors* that set their own NAV to particular value in duration field of short preambles and defer their backoff procedure. Even though senders can't recognize each other which is the case of hidden node problem, *involved neighbors* can do the same operation with the value in duration field of the *early-ACK* frame. After their NAV expires, *involved neighbors* restart their backoff procedure, and finally another node of *involved neighbors* will occupy the medium. This cycle repeats until all of TX-nodes succeed to transmit data to a RX-node. The backoff and NAV scheme is shown in **Fig. 4** in which T1 and T2 are senders try to transmit data to a receiver, R, and know each other. In **Fig. 4**, T1 precedes T2 in occupying the medium. RIX-MAC reduces the probability of collision compared to PW-MAC and then increases the throughput of overall network.

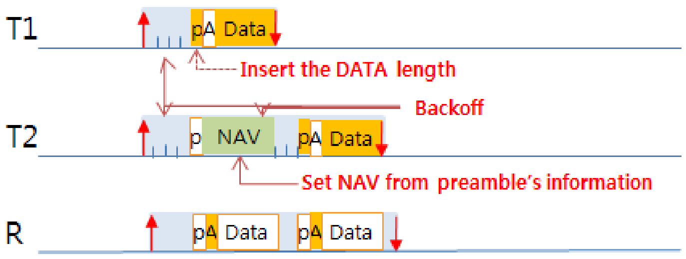


Fig. 4. Operation of multiple senders and a receiver

3.5 Receiving time-out and retransmission

Although it decreases the probability of collision with a random backoff in the case of multiple senders, RIX-MAC protocol uses a time-out approach when collision occurs. In the case of two or more TX-nodes that want to transmit data to a RX-node, they wake up at the same time due to the same information about the destination, try to transmit short preambles and set the timer *k* just before transmission. If two or more of them choose the same random backoff, their short preambles will collide. When short preambles collide, RX-node is not able to decode and reply to them, TX-node keeps transmitting short preambles until RX-node's *early-ACK* frame arrives or the timer *k* expires. When the timer expires, TX-nodes attempt retransmission with new random backoff values. After backoff, TX-nodes operate as described in previous sections.

Fig. 5 shows the described operation under the situation of collision,where T1 and T2 are transmitter and R is a common receiver. Because T1 and T2 select the same random number for backoff, R is not able to reply to short preambles. T1 and T2 choose the random backoff again to occupy the medium, T1 finally succeeds to transmit a data frame.

Consequentially, random backoff and time-out features decrease the likelihood of collision in the case of multiple senders. While our proposed protocol efficiently prevents the collision of data frames, other receiver initiated MAC protocols such as PW-MAC hardly prevent the collision of data frames.

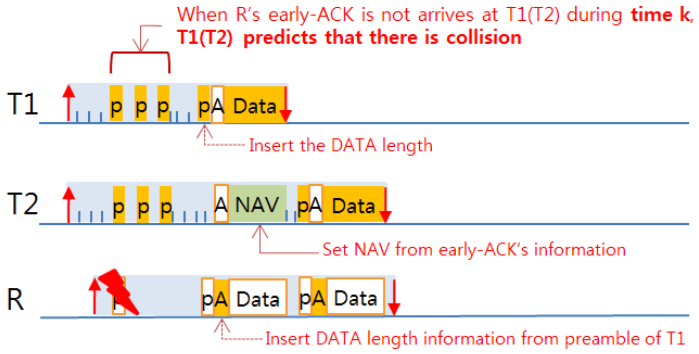


Fig. 5. Operation of multiple senders and a receiver when the collision be occurred.

4. Performance Evaluation

We used ns-2 (version2.32) as a network simulator in order to evaluate the performance of RIX-MAC. It is assumed that each node has a single omni-directional antenna where the radio propagation model of two-way ground reflection is employed. Key parameters of nodes and networks are shown in **Table 2** and **Table 3**, respectively.

Table 2. Node parameters

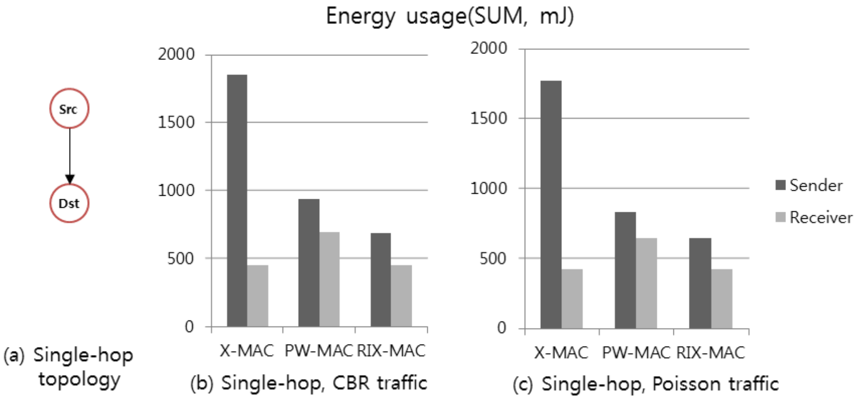
Parameters	**Value**
Routing protocol	Static
Duty cycle rate	5.9 %
Default sleep sustain time	1.395 second
Default Sched-wakeup period	0.088 second
Transport protocol	UDP/NULL
Agent (Traffic model)	CBR and Poisson
Frame size	128 bytes
Inter-node wakeup gap(node 0-1-2-3-4-5)	1.2 second

Table 3. Networking prarmeters

Parameters	**Value**
Bandwidth	250 Kbps
Rx power in mode	15.2 mA
Tx power in mode	28.9 mA
Sleep power in state	0.0004 mA
Idle(wakeup) power in state	0.0087 mA
RX/TX range	250 meter
Simulation running time	1,000 second

To evaluate the performance of our proposed protocol, we measure the energy consumption and the percentage of valid control frames. We also observed the end-to-end delay of frame delivery in the case of multiple senders. With the simulation results, we are going to show the performance and the robustness of protocols in various environments.

Fig. 6 shows the performance of RIX-MAC, X-MAC and PW-MAC protocols in a single-hop topology as shown in **Fig. 6**(a), where two types of traffic are applied. One is CBR static traffic and the other is Poisson random traffic. The simulation results are shown in **Fig. 6**(b) and **Fig. 6(**c). From the results, we observed RIX-MAC uses less energy than others regardless of the traffic model. Repeated short preambles and wasted beacon frames are the causes of higher energy consumption of X-MAC and PW-MAC.



(a) Single-hop topology

(b) Single-hop, CBR traffic

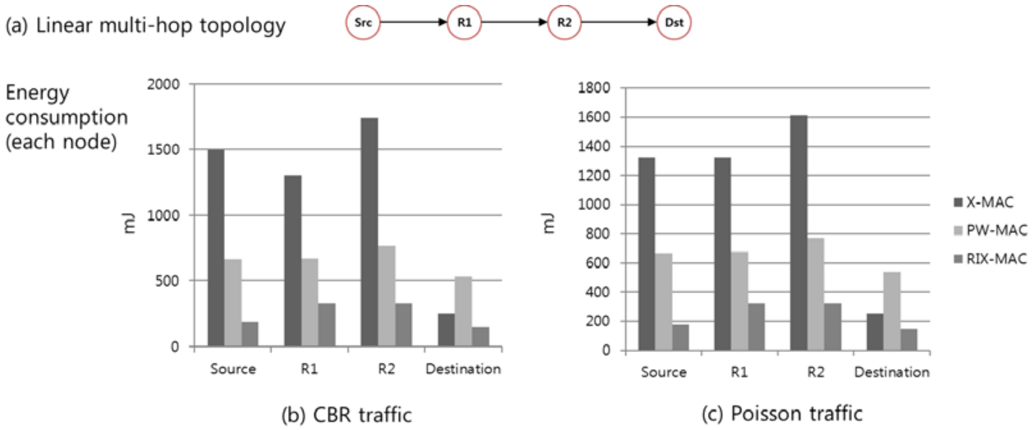
(c) Single-hop, Poisson traffic

Fig. 6. Single-hop topology (a) and simulation results with CBR traffic (b) and Poisson traffic (c)

Fig. 7(b) and (c) shows the performance of RIX-MAC, X-MAC and PW-MAC protocols in linear topology with multi hops as shown in **Fig. 7**(a). Because short preambles are transmitted until RX-node wakes up in X-MAC, the energy consumption of X-MAC is highest. Especially, a large amount of energy is wasted due to a number of short preambles in the case that wakeup time gap is high between TX-node and RX-node in X-MAC. To obtain the utilization of the channel, we denote the percentage of valid control frames as the ratio of transmitted data frames to all transmitted control frames.

$$\%\ of\ valid\ control\ frame = \frac{Transmitted\ data\ frames}{Transmitted\ control\ frames}$$

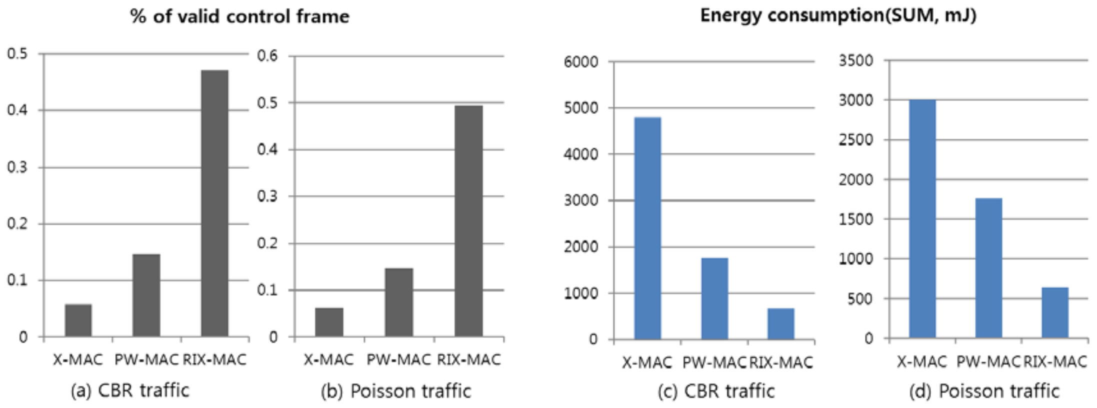
In **Fig. 8**(a) and (b), the percentage of valid control frames is shown, and **Fig. 8**(c) and (d) show the total energy consumption in the network. Our proposed protocol, RIX-MAC, is more highly utilizing the channel than others, since higher percentage of valid control frame implies higher utilization of the channel. X-MAC wastes energy because sender keeps transmitting short preambles until RX-node's *early-ACK* frame arrives, and PW-MAC wastes energy because receiver's beacons are broadcast at every Sched-wakeup. Since energy consumption in a node is highly related to lifetime of the network, RIX-MAC is expected to have a remarkably longer lifetime of than others such as X-MAC and PW-MAC.



(b) CBR traffic (c) Poisson traffic

Fig. 7. Linear multi-hop topology (a) and energy consumption with CBR traffic (b) and Poisson traffic (c)

Fig. 9(b) and (c) shows the performance of RIX-MAC in crossing topology with multiple senders and a receiver such as S1, S2 and R1 as shown in **Fig. 9**(a). In this experimental topology, there are six nodes and R1 is the relay node of two traffic streams in the network, where S1 and S2 generate Traffic 1 and Traffic 2, respectively and S1 generates more frames as twice as S2 (we set packet interval 10s and 5s). As shown in **Fig. 9**(b) and (c), energy consumption at nodes in X-MAC and PW-MAC is higher than in RIX-MAC. In particular, the energy consumption is highest in the link of S1-R1 and S2-R1, because collision of frames between S2 and S1 occurs at R1. The results show that our proposed protocol is effective in terms of energy consumption under the situation of multiple senders.



(a) CBR traffic (b) Poisson traffic (c) CBR traffic (d) Poisson traffic

Fig. 8. Percentage of valid control frame in multi-hop topology with CBR (a) and Poisson (b) and total energy consumption in multi-hop topology with CBR(c) and Poisson (d)

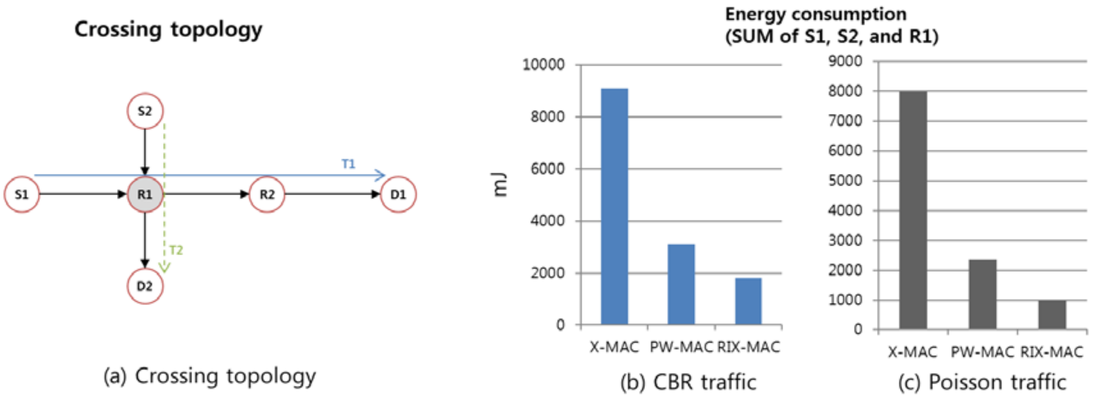


Fig. 9. Crossing topology (a) and total energy consumption at S1, S2 and R1 with CBR traffic (b) and Poisson traffic (c)

Fig. 10 shows the end-to-end delay of data Traffic 1 and Traffic 2 in the same scenario as shown in **Fig. 9**(a). X-MAC has the highest end-to-end delay on both of Traffic 1 and Traffic 2, because continuous transmission of short preambles interfere the other transmission. By the way, PW-MAC and RIX-MAC have low interference, since TX-nodes sleep until corresponding RX-node's wakeup time. In **Fig. 10**, RIX-MAC has lower end-to-end delay leading to higher throughput. Although PW-MAC has a fewer control frames for one data transmission, it is not able to prevent collision under the situation of multiple senders.

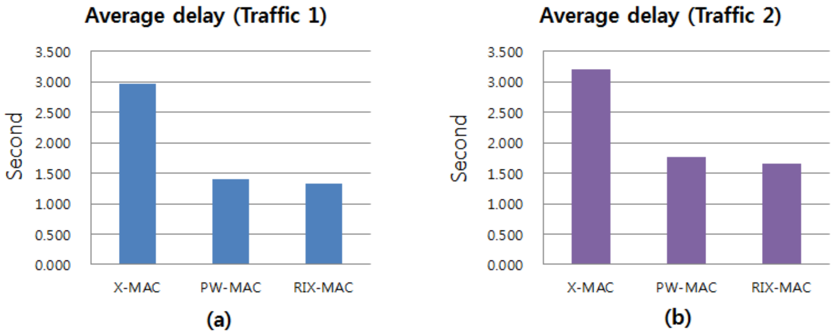


Fig. 10. End-to-end delay in crossing topology of Traffic 1 (a) and Traffic 2 (b) with CBR traffic

5. Conclusion

This paper has presented the design and evaluation of RIX-MAC, a new energy-efficient MAC protocol for sensor networks based on asynchronous duty cycling. Energy saving has been a research issue for wireless sensor networks since the lifetime of networks is critical. RIX-MAC is designed to minimize energy consumption by using short preambles and enabling senders to predict receiver's wakeups. Transmission collisions may occur over wireless channels, especially with bursty traffic that may be usual in a sensor network. The backoff mechanism of RIX-MAC also reduces the likelihood of transmission collisions in the case of multiple senders.

We conducted simulation on ns-2 simulator to evaluate the performance of RIX-MAC compared with X-MAC and PW-MAC. Our simulation includes scenarios in which nodes have a multi-hop linear topology and a crossing topology with multiple traffic flows. As a result, RIX-MAC reduces the energy consumption at sender and receiver nodes and increases the lifetime of sensor networks by lower collision and unnecessary wakeups. Simulation results also show that our proposed protocol outperforms the previous protocols in terms of end-to-end delay by avoiding collision and reducing overhead.

References

[1] Michael. Buettner, Gray V. Yee, Eric Anderson and Richard Han, "X-MAC: a short preamble mac protocol for duty-cycled wireless sensor networks," In *Proc. of the 4th ACM Conference on Embedded Networked Sensor System*, pp. 307-320, Nov 2006. Article (CrossRef Link)

[2] Wei Ye, John Heidemann, and Deborah Estrin, “An energy-efficient MAC protocol for wireless sensor networks,” In *21st International Annual Joint Conference of the IEEE Computer and Communications Societies*, pp. 1567-1576, June 2002. Article (CrossRef Link)

[3] Tijs Van Dam and Koen Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," In *Proc. of the first ACM Conference on Embedded Networked Sensor System*, pp. 171-180, Nov. 2003. Article (CrossRef Link)

[4] Joseph Polastre, Jason Hill, and David Culler, "Versatile low power media access for wireless sensor networks," In *Proc. of the second ACM Conference on Embedded Networked Sensor System*, pp. 95-107, 2004. Article (CrossRef Link)

[5] Yanjun Sun, Omer Gurewitz, and David B. Johnson, “RI-MAC: A receiver initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks,” In *Proc. Of the 6th ACM Conference on Embedded Networked Sensor System*, pp. 1-14, 2008. Article (CrossRef Link)

[6] Lei Tang, Yanjun Sun, Omer Gurewitz, and David B. Johnson, “PW-MAC: An energy-efficient predictive-wakeup MAC protocol for wireless sensor networks,” In *Proc. of the 30th IEEE International Conference on Computer Communications*, pp. 1305-1313, Apr. 2011. Article (CrossRef Link)

[7] ns-2, URL: http://www.isi.edu/nsnam/ns/, Last access: Sep. 2013.

[8] Myungsub Lee and, Chang-Hyeon Park, “A differential probability selection MAC protocol considering energy consumption in wireless sensor networks,” *International Journal of Distributed Sensor Networks 2013*, Vol. 2013, Article ID 217542, 8 pages, 2013. Article (CrossRef Link)

[9] Kyung-Tae Kim and, Hee-Yong Youn, “An energy-efficient MAC protocol employing dynamic threshold for wireless sensor networks,” *International Journal of Distributed Sensor Networks 2012*, Vol. 2012, Article ID 304329, 12 pages, 2012. Article (CrossRef Link)

[10] K. Nguyen, Y. Ji, S. Yamada, “Low overhead MAC protocol for low data rate wireless sensor networks,” *International Journal of Distributed Sensor Networks2013*, Vol. 2013 Article ID 217159, 8 pages, 2013, Article (CrossRef Link)

[11] Daqiang Zhang, ZhangBing Zhou, Qin Zou, Tianyi Zhan and Minho Jo, “Asynchronous event detection for context inconsistency in pervasive computing,” *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 4, pp. 195-205, 2012. Article (CrossRef Link)

[12] Jae-Ho Lee, Kyeong Hur, and Doo-Seop Eom, “WTE-MAC: wakeup time estimation MAC for improving end-to-end delay performance in WSN,” In *Proc. of Military Communications Conference 2011*, pp. 902-907, Nov., 2011. Article (CrossRef Link)

[13] Jeong-Yeob Oak, Young-June Choi, and Wooguil Park, “EP-MAC: early preamble MAC to achieve low delay and energy consumption in duty cycle based asynchronous wireless sensor networks,” *International Journal of KSII Transactions on Internet and Information System*, Vol. 6, N0. 11, pp. 2980-2990, Nov., 2012. Article (CrossRef Link)

[14] K. Stone and M. Colagrosso, "Efficient duty cycling through prediction and sampling in wireless sensor networks.", In *Proc. of International Journal of InterScience, Wireless Communications*

and Mobile Computing, pp. 1087-1102, May, 2007. Article (CrossRef Link)

[15] A. El-Hoiydi, and J.-D. Decotignie, "WiseMAC: An ultra low power MAC protocol for multi-hoip wireless sensor networks," In *Proc. of the first International Workshop in Algorithmic Aspects of Wireless Sensor Networks*, pp. 18-31, July, 2004. Article (CrossRef Link)

[16] L. Tang, Y. Sun, O. Gurewitz and D. B. Hohnson, "EM-MAC: A Dynamic Multi channel Energy-Efficient MAC Protocol for wireless sensor networks," In *Proc. of the 8th Wireless Communications and Mobile Computing International Conference (IWCMC*), pp. 1159 - 1164, 2012. Article (CrossRef Link)

[17] S. Lim, Y. Kang, J. Jeong, and C. Kim, "Design, Analysis and Evaluation of A New Energy Conserving MAC protocol for Wireless Sensor Networks," *International Journal of KSII Transactions on Internet and Information System*, vol. 6, N0. 12, Dec., 2012. Article (CrossRef Link)

[18] IEEE, Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low Rate Wireless Personal Area Networks (LR-WPANS), IEEE 802.15.4-2011, 2011. Article (CrossRef Link)

[19] A. El-Hoiydi, "Aloha with Preamble Sampling for Sporadic Traffic in Ad Hoc Wireless Sensor Networks," In *Proc. of IEEE International Conference on Communications*, Vol. 5, pp. 3418-3423- Apr., 2002. Article (CrossRef Link)

Inhye Park received the BS and MS degree in computer engineering from Kwangwoon university in 2008 and 2010, respectively. She is currently a candidate of PhD degree in computer engineering from Kwangwoon university, Seoul, Korea. Her current research interests are in the area of wireless networks including mobile ad hoc networks, wireless LAN, wireless sensor networks and tactical communications.

Hyungkeun Lee received the B.S. degree in electronic engineering from Yeonsei university, Seoul, Korea, in 1987, and M.S. and Ph.D. degree in computer engineering from Syracuse university, NY, USA in 1998 and 2002, respectively. He is an associate professor in the department of computer engineering at Kwangwoon university, Seoul, Korea. He was with Samsung Electronics as a senior research engineer for six years. His current research interests are in the area of wireless networks such as mobile ad hoc networks, wireless LAN, wireless sensor networks and wireless multimedia communication. He is also interested in design and analysis of tactical data links in defense systems.

Seokjoong Kang received his BS and MS degrees in Computer Science from Indiana University in 1988 and 1991 respectively and his Ph.D degree in Electrical Engineering and Computer Science from University of California, Irvine in 2003. After the completion of his Ph.D program, he worked as a lecturer and research staff at UCI. During 1991-1998, he worked as a senior researcher at Korea Institute of Defense Analyses in Republic of Korea. He also worked as a principle researcher at Samsung Electronics Co. during 2004-2006. Currently, he is a associate professor in the department of Management of Technology for Defense at Korea University, Seoul, Korea. His research interest includes Embedded System, Software engineering, Real-time system and Distributed systems.