EP-MAC: Early Preamble MAC To Achieve Low Delay And Energy Consumption In Duty Cycle Based Asynchronous Wireless Sensor Networks

Jeong-Yeob Oak[1], Young-June Choi[2] and Wooguil Pak[3]
[1] Access S/W Lab, Network Division, Samsung Electronics,
416, Maetan3-dong, Yeongtong-gu,
Suwon, 443-742, Korea
[2]School of Information and Computer Engineering, Ajou University,
San 5, Woncheon-dong, Yeongtong-gu, Suwon, 443-749, Korea
[3]Jangwee Research Institute for National Defence
San 5, Woncheon-dong, Yeongtong-gu, Suwon, 443-749, Korea
[e-mail: jyoak@netlab.snu.ac.kr, choiyj@ajou.ac.kr, wgpak@ajou.ac.kr]
*Corresponding author: Wooguil Pak

Received January 10, 2012; revised May 25, 2012; accepted October 17, 2012; published November 30, 2012

Abstract

Since wireless sensor networks are broadly used in various areas, there have been a number of protocols developed to satisfy specific constraints of each application. The most important and common requirements regardless of application types are to provide a long network lifetime and small end-to-end delay. In this paper, we propose Early Preamble MAC (EP-MAC) with improved energy conservation and low latency. It is based on CMAC but adopts a new preamble type called 'early preamble'. In EP-MAC, a transmitting node can find quickly when a next receiving node wakes up, so EP-MAC enables direct data forwarding in the next phase. From numerical analysis, we show that EP-MAC improves energy consumption and latency greatly compared to CMAC. We also implemented EP-MAC with NS-2, and through extensive simulation, we confirmed that EP-MAC outperforms CMAC.

Keywords: Wireless sensor networks, duty cycle, preamble, energy efficient, anycast

This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency" (NIPA-2012-(C1090-1221-0011)).

http://dx.doi.org/10.3837/tiis.2012.10.013

1. Introduction

Wireless sensor networks (WSNs) reduce the cost of managing industrial devices and home appliances. A WSN consists of many sensor nodes which are combinations of RF (Radio Frequency) transceivers and sensors. The most important and common requirements of WSN node are a long network lifetime and low end-to-end delay since nodes are battery-powered and the sensing data has a lifetime. Today, hundreds of MAC (Medium Access Control) protocols have been introduced to achieve these goals. However, it is very difficult to satisfy both of requirements simultaneously [7][9][10][11].

In this paper, we propose a new asynchronous MAC protocol called EP-MAC (Early Preamble MAC). It adopts strobed preamble and anycast multi-hop forwarding from X-MAC and CMAC, respectively. It also extends the concept of the strobed preamble to a data payload for reducing the energy consumption and latency of packet forwarding. The new preambles are called 'Early Preambles (EPs)' and they are transmitted during a data packet transmission period. Although EP-MAC does not adopt power control schemes for low energy consumption, it can achieve low delay and energy consumption, simultaneously using EPs.

If a WSN adopts unicast based routing protocols, the reliability of data transmission is significantly affected by node failures. However, EP-MAC is based on CMAC, which adopts an anycast protocol approach. Therefore, the failures of WSN nodes do not affect the reliable data transmission.

We formulated expressions for energy consumption and a end-to-end delay with total EP duration time. From numerical analysises, we showed that EP-MAC achieves a better performance compared to CMAC. We also confirmed that EP-MAC overcomes CMAC through extensive simulations.

The remainder of this paper is organized as follows. In Section II, we describe related work. In Section III, we develop our EP-MAC protocol based on CMAC to achieve very small end-to-end delay and low energy consumption, and present numerical analysis. In Section IV, we show simulation and analysis results. In Section V, we conclude this paper.

2. Related Work

In WSNs, many MAC protocols have been proposed to support a long network lifetime and a low end-to-end latency. In this section, we briefly introduce well-known asynchronous MAC protocols such as B-MAC, X-MAC and CMAC.

2.1 B-MAC

B-MAC (Berkeley MAC) is a contention-based asynchronous MAC protocol for WSNs [1]. B-MAC is simply implementable and provides flexible interfaces for ultra low power operations such as efficient collision avoidance. B-MAC does not use synchronization. Instead, each node sleeps periodically and independently. Due to this feature, B-MAC does not suffer from the synchronization overhead for adjusting the wakeup schedule of each node. Before forward a data packet, B-MAC transmits the preamble of which length is longer than the sleep period of a receiving (RX) node. When a RX node awakes, it detects the preamble by sampling the channel and then, it will receive the data packet from a transmitting (TX) node as shown in **Fig. 1**. The wakeup duration time is usually fixed and the sleep interval can be adjusted according to the total number of neighbor nodes and the amount of transmitted traffic.

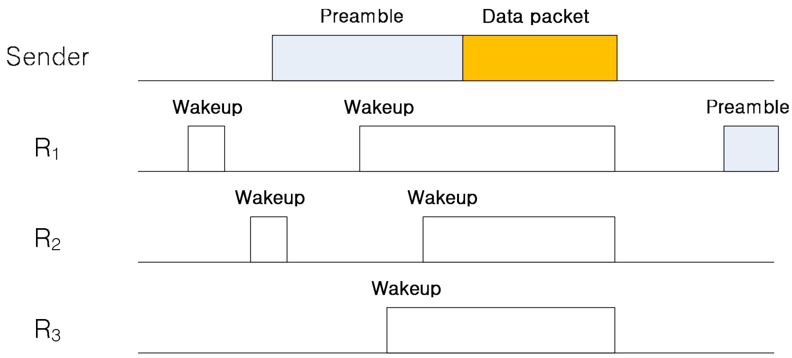

Fig. 1. Packet transmission in B-MAC. R_1 is a destination of data packet. Other nodes except R_1 give up forwarding the packet.

B-MAC can be implemented without control messages such as RTS (Request to Send) and CTS (Clear to Send) packets and a large memory. However, a RX node consumes much energy to listen to a long preamble until the TX node starts to transmit a preamble. Moreover, the preamble also incurs a long transmission delay. Due to the overhearing problem, if a data packet is delivered through multi-hops, its end-to-end delay is increased significantly and the throughput is degraded seriously. All nodes in the communication range of a TX node should listen to a preamble until the TX node finishes transmitting the preamble and sends a data packet.

2.2 X-MAC

X-MAC is a low power MAC protocol for WSNs. It achieves high energy efficiency by adopting a new low power communication called improved LPL (Low Power Listening) [2]. This MAC shares many features with B-MAC and belongs to asynchronous MAC protocols. X-MAC uses preamble sampling to notify the existence of data to forward to a RX node based on short preambles. The short preambles solve most of the problems caused by the long preamble of B-MAC. X-MAC shows a good performance for most of cases compared to B-MAC. The main characteristics of X-MAC are summarized as follows.

- *Short preambles with TX node ID*
  In previous LPL, a RX node is awake for a long time to receive a data packet. However, all other nodes should also listen to a whole preamble since they cannot identify the destination of a data packet from the preamble. To solve this overhearing problem, X-MAC uses multiple short preambles with an identifier (ID) of the RX node, i.e., the destination node. When other nodes except the RX node receive a preamble, they can sleep quickly to avoid unnecessary energy consumption.

- *Strobed preamble*
  When a node receives the preamble of which ID matches with the node ID, it does not receive following preambles prior to a data packet. There is a pause between two consecutive preambles and the node sends an 'early ACK (acknowledgement)' to notify the TX node that a RX node is ready to receive a data packet. Then, the TX node stops transmitting preambles and transmits the data packet.

2.3 CMAC

Convergent-MAC (CMAC) is an energy-effective and routing-integrated protocol for WSNs [3]. This protocol can forward a data packet to a sink node through multi-hop routes. All nodes that are closer to the sink node than a TX node are candidate RX nodes. At first, a TX node broadcasts a RTS packet to neighbor nodes. Then, some nodes which listened to the packet, i.e., candidate RX nodes try to send CTS packets back to the TX node in a contention manner. Finally, the winner of the contention becomes a RX node and it receives a data packet from the TX node. This procedure is repeated until the data packet is received by a sink node. This integrated MAC and routing protocol is called 'anycast' [4][5][8].

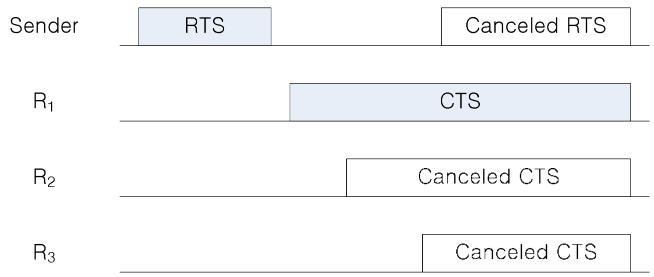

Fig. 2. Selection of a RX node among multiple candidate RX nodes by RTS-CTS exchanges. R_1 is closest to the sink node.

The RX node selection process is shown in **Fig. 2**. It is similar to the IEEE 802.11 MAC but uses a technique called prioritized CTS. Using the prioritized CTS, candidate nodes are able to avoid simultaneous CTS transmissions. The priority is defined by the distance between the node and the sink node, so a higher priority is given as the distance is shorter. If a candidate node receives a CTS packet with a higher priority, other candidate nodes give up transmitting their CTS packets. Therefore, the nearest node to the sink node becomes a RX node with the highest probability.

CMAC operates in two modes: anycast and unicast. Initially, CMAC operates in an anycast mode for low duty cycle asynchronous operation. As more packets are forwarded, the end-to–end path is converged to the shortest path, which is identical to a unicast mode for maximizing transmission efficiency.

3. Protocol Description

Our protocol, EP-MAC (Early Preamble-MAC) extends the concept of early preamble by adopting strobed preamble and anycast multi-hop forwarding from X-MAC and CMAC, respectively. EP-MAC also provides a unique technique to further reduce the length of preamble transmission period.

If a TX node has a data packet which is forwarded to the sink node, it broadcasts consecutive RTS packets to select a RX node. The RX node selection process is exactly the same with that of CMAC. As mentioned in Section 2, the prioritized CTS is used to avoid contention among RX candidate nodes. The priority is defined by the distance between the node and the sink node and the node with higher priority, i.e., closer to the sink node is selected.

In **Fig. 3**, a TX node finds a RX node, i.e., $\mathbf{R_1}$ by consecutive preambles. A RX node becomes the next TX node and will transmit consecutive preambles again to find a next RX node. When $\mathbf{R_2}$, a next candidate RX node wakes up during a data packet transmission of previous nodes, it will go to a sleep mode. This situation causes a long transmission delay as well as a high energy consumption of the TX node. The performance becomes worse, as the data packet size is longer or the duty cycle is lower.

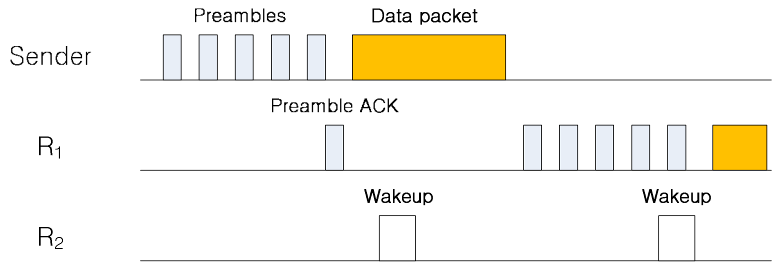

Fig. 3. Long transmission delay in X-MAC.

The packet transmission process of CMAC is also similar to X-MAC, so it suffers from the long transmission delay. EP-MAC extends the 'strobed preamble' concept to 'early preamble' for data packet. The EP is transmitted between divided data packets. It includes the length of the remaining data packet and the location information of the node that transmitted the packet to reduce a transmission delay. Due to EP, the RX node can transmit a data packet directly after the completion of receiving a data packet. EP is exchanged with the next RX node during the data transmission period. **Fig. 4** describes the packet transmission process of EP-MAC.

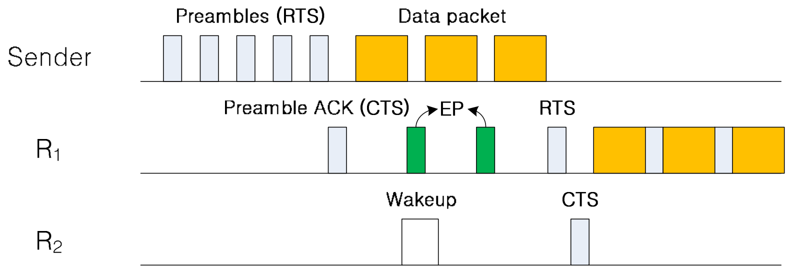

Fig. 4. Packet forwarding in EP-MAC.

When a TX node transmits a data packet after consecutive RTS packet transmission, there are pauses between transmissions of each divided part of the data packet. In each pause time, the RX node broadcasts an EP to neighbor nodes. Any neighbor node that received the EP, it checks if the node is closer to the sink node than the RX node, and then, it sleeps according to its wakeup-sleep schedule. Then, the node closer to the sink node wakes up when the RX node sends a RTS packet. Since the EP contains remaining data packet length, the node can know when it wakes up to received the RTS packet. By this way, one-hop transmission delay is decreased significantly and consequently, the energy consumption is also decreased.

Finally, if the packet transmission between TX and RX nodes is completed, the RX node becomes a next TX node to relay the received data. This process is repeated until the sink node receives the data packet.

As EP duration time increases, more EPs are transmitted, so the average number of candidate RX nodes increase. Owing to the increased candidate RX number, energy consumption and end-to-end delay are decreased simultaneously. We will formulate energy consumption and end-to-end delay according to EP duration, respectively in the following sub sections.

3.1 Average End-To-End Delay

We numerically analyze the performance of EP-MAC in terms of energy consumption and end-to-end delay with respect to EP duration (x), i.e., the total duration time of early preambles in **Fig. 5**.

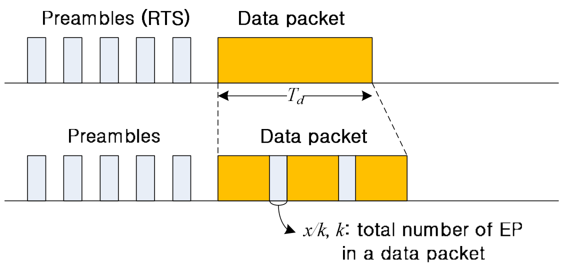

Fig. 5. MAC model for numerical analysis.

For the simplicity of our analysis, we define some variables as follows:

- x : total EP duration time
- Z : inter-packet arrival time
- G : CDF (cumulative distribution function) of Z
- λ : average packet arrival rate
- n : total number of forwarding nodes
- L : total number of hops
- L_D : inter-wakeup time
- T_d : data transmission time
- P_t : power consumption for transmission packet

If a neighbor node in the next hop of a RX node wakes up in the EP duration time, the RX node can forward a data packet to the neighbor node, i.e., the next RX node without waiting for the node to wake up. This case happens with the probability of $P\{Z < x\} = G(x)$. In this case, the average 1-hop transmission delay is $L_{a1} = x + T_d$. On the other hand, if the neighbor node wakes up out of EP duration time, the protocol uses a strobed preamble with the probability of $P\{Z \geq x\} = 1 - G(x)$. The average 1-hop transmitting delay is $L_{a2} = \frac{L_D}{n+1} e^{-\lambda x} + T_d$ because the average time to meet the wake up node is $\frac{L_D}{n+1}$.

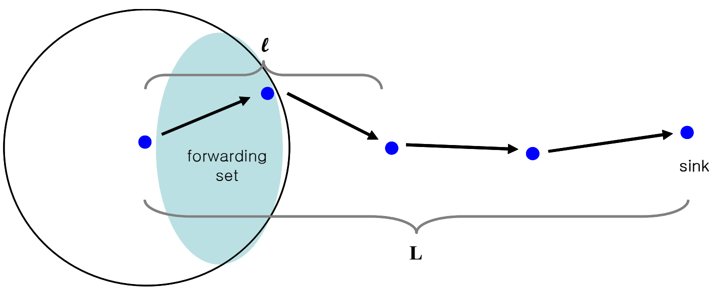

Fig. 6. Multi-hop model for numerical analysis.

As shown in **Fig. 6**, for L hop transmissions, the average transmission delay is given as follows.

$$L_a = \sum_{l=0}^{L} \left\{ L_{a1} \cdot l \cdot \binom{L}{l} G'(x)^l \{1 - G'(x)\}^{L-l} + L_{a2} \cdot (L-l) \cdot \binom{L}{l} \{1 - G'(x)\}^l G'(x)^{L-l} \right\}.$$

Assuming that $G(x) = 1 - e^{-\lambda x}$, the delay is given as follows [6].

$$L_a = \sum_{l=0}^{L} \binom{L}{l} \left\{ l \cdot (x + T_d)(\lambda e^{-\lambda x})^l (1 - \lambda e^{-\lambda x})^{L-l} + (L-l) \cdot \left(\frac{L_D}{n+1} e^{-\lambda x} + T_d \right) (1 - \lambda e^{-\lambda x})^l (\lambda e^{-\lambda x})^{L-l} \right\}.$$

Fig. 7 shows that the delay decreases exponentially as x, i.e., EP duration time increases.

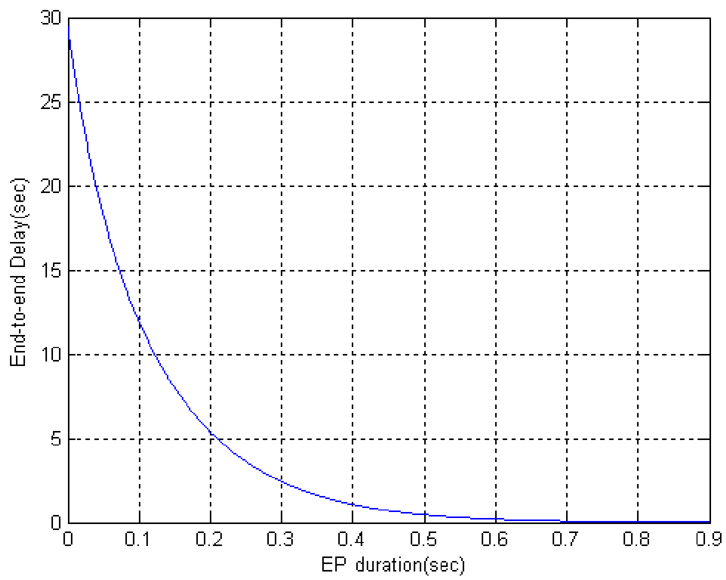

Fig. 7. End-to-end delay according to EP duration x, where λ =10, n=8, L=10, L_D=1sec, T_d=91.52msec and P_t=1.5W.

3.2 Average Energy Consumption

We calculate the average energy consumption in a similar manner to the average end-to-end delay. If a neighbor node in a next hop wakes up in the EP duration, a RX node can forward a data packet to the next node without delay. Therefore, the average energy consumption is given as

$$E_{a1} = G(x) \cdot p_t \cdot E[z \mid z < x] = G(x) \cdot p_t \frac{\int_0^x z \lambda e^{-\lambda z} dz}{\int_0^x \lambda e^{-\lambda z} dz} = G(x) \cdot p_t \frac{\frac{1}{\lambda} - \left(x + \frac{1}{\lambda} \right) e^{-\lambda x}}{1 - e^{-\lambda x}},$$

$$E_{a2} = \{1 - G(x)\} \cdot p_t \cdot \left\{ x + \frac{L_D}{n+1} \right\}.$$

From these equations, we calculate the average energy consumption as follows.

$$E_a = \sum_{l=0}^{L} \{ E_{a1} \cdot l \cdot a(l) + E_{a2} \cdot (L-l) b(l) \}$$

$$= \sum_{l=0}^{L} \left\{ E_{a1} \cdot l \cdot \binom{L}{l} G'(x)^l \{1 - G'(x)\}^{L-l} + E_{a2} \cdot (L-l) \cdot \binom{L}{l} \{1 - G'(x)\}^l G'(x)^{L-l} \right\}.$$

Assuming $G(x)=1-e^{-\lambda x}$, we obtain the average energy consumption for the L-hop case as follows:

$$E_a=\sum_{l=0}^{L}\binom{L}{l}\left\{p_t\left\{\frac{1}{\lambda}-\left(x+\frac{1}{\lambda}\right)e^{-\lambda x}\right\}l(\lambda e^{-\lambda x})^l(1-\lambda e^{-\lambda x})^{L-l}+p_t\cdot e^{-\lambda x}\left(x+\frac{L_D}{n+1}\right)(L-l)(1-\lambda e^{-\lambda x})^l(\lambda e^{-\lambda x})^{L-l}\right\}.$$

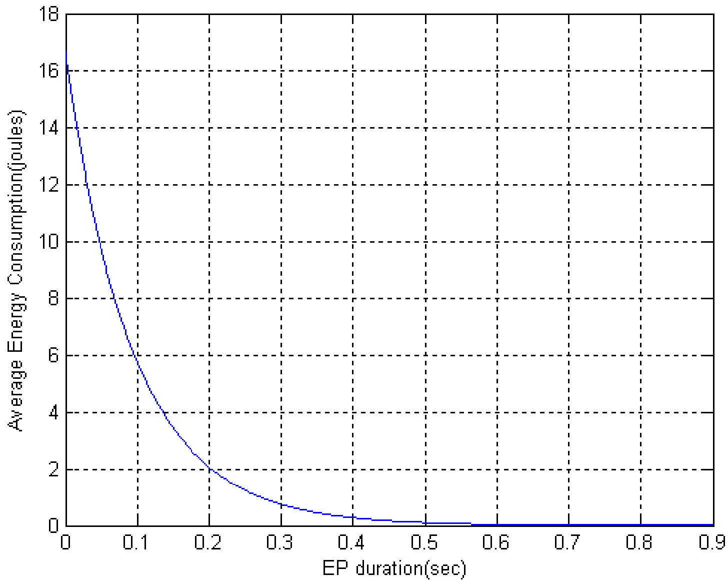

Fig. 8. Average energy consumption according to EP duration x, where λ =10, n=8, L=10, L_D=1sec, T_d=91.52msec and P_t=1.5W.

We can confirm that the average energy consumption also decreases as the EP duration increases from **Fig. 8**.

4. Simulation

We implemented EP-MAC in NS-2 to evaluate the performance by extending the CMAC simulator. We compared performances of EP-MAC and CMAC in terms of end-to-end delay and energy consumption. We used a two-ray ground model and the topology consists of 64 sensor nodes locating in 700 m × 700 m grid topology. Other simulation parameters are given as **Table 1**.

Table 1. Simulation parameters

Field	Configured value
TX power	27 mA
RX power	10 mA
Idle power	10 mA
Bandwidth	38.4 kbps
Transmission range	250 m
Interference range	500 m
Traffic interval	0.1 sec
RTS size	44 bytes
Data packet size	220 bytes

4.1 End-To-End Delay vs. Node Density

We measured the end-to-end delay, when the node density increases. We started simulation with 64 (=8×8) nodes in 700 m × 700 m area. We increased the total number of nodes to 81 (=9×9), 100 (=10×10), 121 (=11×11) , 144(=12×12) and 169 (13×13). The duty cycle of each sensor node is fixed at 40%.

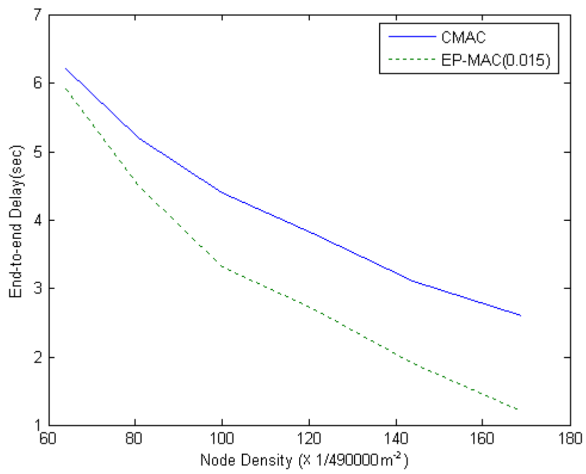

Fig. 9. End-to-end delay according to node density.

As shown in **Fig. 9**, EP-MAC outperforms CMAC in terms of an end-to-end delay. The gap of performance becomes bigger as the node density increases. When the node density is high, the probability of RX node to receive RTS by EP is high. Consequently, EP-MAC achieves a short end-to-end delay.

4.2 End-To-End Delay vs. Duty Cycle

We investigated the end-to-end delay as a function of the duty cycle which is increased from 10% to 70% when the node density is fixed. The simulation result is shown in **Fig. 10**.

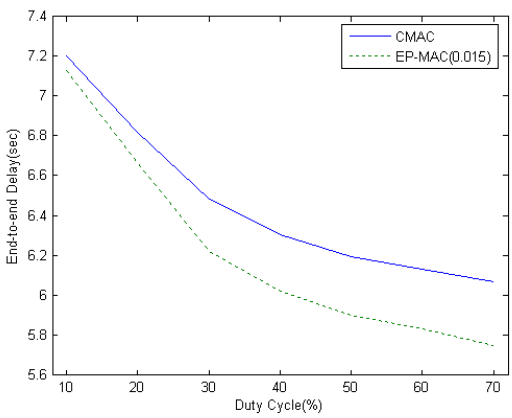

Fig. 10. End-to-end delay according to duty cycle

Fig. 10 shows that the network delay is decreased as the duty cycle increases. The performance gap between EP-MAC and CMAC is more salient as the duty cycle is increased since the probability to find a next hop node in EP duration is increased when the duty cycle increases. Therefore, the end-to-end delay is inverse proportional to the duty cycle.

4.3 Energy Consumption vs. Node Density

We measured the energy consumption of CMAC and EP-MAC while a node density increases when the duty cycle is fixed at 40%.

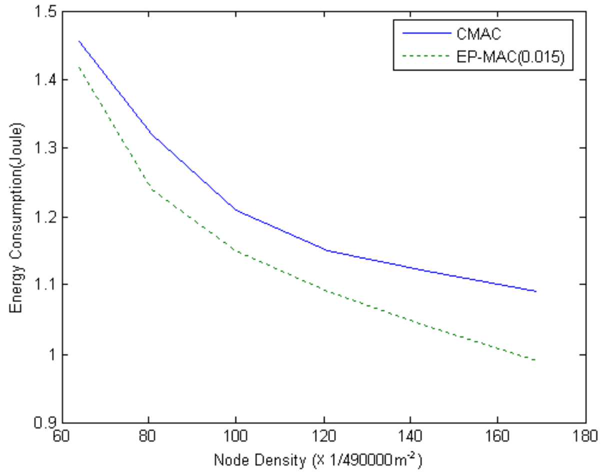

Fig. 11. Energy consumption according to node density.

From **Fig. 11**, we can see that the energy consumption of CMAC and EP-MAC is decreased because the probability of detecting an awake node close to a TX node becomes high as the node density increases. Due to EP, the performance gain between EP-MAC and CMAC grows bigger as the density increases.

4.4 Energy Consumption vs. Duty Cycle

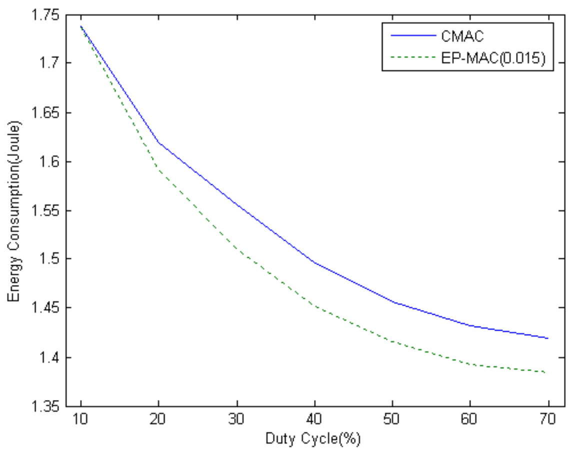

Fig. 12. Energy consumption according to duty cycle.

We measured the energy consumption of CMAC and EP-MAC while the duty cycle increases from 10% to 70% when other conditions are fixed.For a low duty cycle of less than 10% in **Fig. 12**, EP-MAC shows almost the same performance as CMAC. However, the performance gap between EP-MAC and CMA increases as the duty cycle is increased. From simulation results, we can confirm that EP-MAC outperforms CMAC for energy consumption as well as end-to-end delay.

5. Conclusion

This paper proposes a new asynchronous protocol named EP-MAC for small energy consumption and short end-to-end delay in wireless sensor networks (WSNs). EP-MAC achieves a better performance than conventional CMAC by broadcasting EP packets to neighbor nodes of the receiving node during a data transmitting period. Our numerical analysis and simulation results confirm that EP-MAC outperforms CMAC. EP-MAC improves energy consumption and delay simultaneously to find candidate RX nodes for current RX node in advance. Since EP-MAC does not rely on any power control schemes, we will extend EP-MAC to adopt other energy efficient techniques such as power control for better performance. We expect it will be widely used for various applications, e.g, large scaled environmental monitorings.

References

[1] Joseph Polastre, Jason Hill, and David Culler, "versatile low power media access for wireless sensor networks," in *Proc. of IEEE SenSys*, 2004. Article (CrossRef Link)

[2] Michael Buettner, Gary Yee, Eric Anderson, and Richard Han, "X-MAC: A Short Preamble MAC protocol for duty-cycled wireless sensor networks," in *Proc. of IEEE SenSys*, 2006. Article (CrossRef Link)

[3] Sha Liu, Kai-Wei Fan, and Prasun Sinha, "CMAC: An energy efficient MAC layer protocol using convergent packet forwarding for wireless sensor networks," *Fourth Annual IEEE Communications Society Conference on Sensor, Mesh, and Ad Hoc Communications and Networks*, San Diego, California, USA, Jun. 2007. Article (CrossRef Link)

[4] Kai-Wei Fan, Sha Liu, and Prasun Sinha, "Convergent anycast: a low duty-cycle mac layer for sensor networks," OSU-CISRC-4/05-TR24.

[5] Sunil Kulkari, Aravind Iyer, and Catherine Rosenberg, "An Address-light, Integrated MAC and routing protocol for wireless sensor networks," *IEEE/ACM Transactions on Networking*, Aug. 2006. Article (CrossRef Link)

[6] Michele Zorzi and Ramesh R. Rao, "Geographic random forwarding for ad hoc and sensor networks: energy and latency performance," *IEEE Transactions on Mobile Computing*, Oct. 2003. Article (CrossRef Link)

[7] Abdelmalik Bachir, Mischa Dohler, Thomas Watteyne, and Kin K. Leung, "MAC essentials for wireless sensor networks," *IEEE Communication surveys &Tutorials*, vol. 12, no. 2 2010. Article (CrossRef Link)

[8] Anycast, http://en.wikipedia.org/wiki/Anycast

[9] Xiaolei Shi; Stromberg, G., "SyncWUF: An ultra low-power mac protocol for wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol.6, no. 1, pp.115-125, Jan. 2007. Article (CrossRef Link)

[10] Z. Chen, C.-X. Wang, X. Hong, J. S. Thompson, S. A. Vorobyov, X. Ge, H. Xiao, and F. Zhao, "Aggregate interference modeling in cognitive radio networks with power and contention control," IEEE Transactions on Communications, vol. 60, no. 2, pp. 456-468, Feb. 2012. Article (CrossRef)

[11] I. Humar, X. Ge, L. Xiang, M. Jo and M. Chen, "rethinking energy-efficiency models of cellular networks with embodied energy," *IEEE Network Magazine*, vol.25, no.3, pp.40-49, Mar. 2011. Article (CrossRef Link)

Jeong-Yeob Oak received B.S. degree from the Department of Electrical Engineering, Yonsei University, in 2007, and M.S. degree from the Department of Electrical Engineering & Computer Science, Seoul National University, in 2009. From 2009, He is with Samsung Electronics Company as a software engineer, Suwon, Korea. His areas of interests include fourth-generation wireless networks, radio resource management, and wireless sensor networks.

Young-June Choi received B.S., M.S., and Ph.D. degrees from the Department of Electrical Engineering & Computer Science, Seoul National University, in 2000, 2002, and 2006, respectively. From Sept. 2006 through July 2007, he was a postdoctoral researcher at the University of Michigan, Ann Arbor, MI, USA. From 2007 through 2009, he was with NEC Laboratories America, Princeton, NJ, USA, as research staff member. He is currently an assistant professor at Ajou University, Suwon, Korea. His research interests include fourth-generation wireless networks, radio resource management, and cognitive radio networks.

Wooguil Pak received B.S. and M.S. degrees in Electrical Engineering from Seoul National University in 1999 and 2001, respectively, and the Ph.D. degree in Electrical Engineering and Computer Science from Seoul National University in 2009. From 2001 through 2009 he was with Samsung SECUi.COM as a software engineer. In 2010, he joined the Jangwee Research Institute for National Defence as a research professor. His current research interests include MAC and routing protocol design and QoS provisioning for wireless sensor network and wireless mesh network, and network security for high speed networks.