Adaptive Duty Cycling MAC Protocols Using Closed-Loop Control for Wireless Sensor Networks

Jaehyun Kim[1], Seoggyu Kim[2] and Jaiyong Lee[1]
[1] Department of Electrical and Electronic Engineering, Yonsei University
Seoul, Korea
[e-mail: macross7@yonsei.ac.kr, jyl@yonsei.ac.kr]
[2] Department of Informaton Communication Engineering, Andong National University
Andong-si, Gyeongsangbuk-do, Korea
[e-mail: sgkion@andong.ac.kr]
*Corresponding author: Jaehyun Kim

Received October 17, 2010; revised November 30, 2010; accepted December 11, 2010; published January 31, 2011

Abstract

The fundamental design goal of wireless sensor MAC protocols is to minimize unnecessary power consumption of the sensor nodes, because of its stringent resource constraints and ultra-power limitation. In existing MAC protocols in wireless sensor networks (WSNs), *duty cycling*, in which each node periodically cycles between the active and sleep states, has been introduced to reduce unnecessary energy consumption. Existing MAC schemes, however, use a fixed duty cycling regardless of multi-hop communication and traffic fluctuations. On the other hand, there is a tradeoff between energy efficiency and delay caused by duty cycling mechanism in multi-hop communication and existing MAC approaches only tend to improve energy efficiency with sacrificing data delivery delay. In this paper, we propose two different MAC schemes (ADS-MAC and ELA-MAC) using closed-loop control in order to achieve both energy savings and minimal delay in wireless sensor networks. The two proposed MAC schemes, which are synchronous and asynchronous approaches, respectively, utilize an adaptive timer and a successive preload frame with closed-loop control for adaptive duty cycling. As a result, the analysis and the simulation results show that our schemes outperform existing schemes in terms of energy efficiency and delivery delay.

Keywords: MAC, wireless sensor network, duty cycling, energy efficiency, delivery delay

This research was supported by the MKE (Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2010-(C1090-1011-0006))

DOI: 10.3837/tiis.2011.01.006

1. Introduction

Ubiquitous computing and networking, often called pervasive computing, is an emerging field of research associated with developments in technologies such as wireless communications and networking, embedded systems, sensors, and radio frequency identification (RFID) systems, and has led to the evolution of the natural successors of mobile computing systems. The goal of ubiquitous computing is to create ambient intelligence, through which network devices embedded in the environment provide unobtrusive connectivity and services at all times, thus improving the human experience and quality of life without users having any explicit awareness of underlying communications and computing technologies. Wireless sensor networks (WSNs) have recently received a great deal of attention due to their wide range of applications, such as object tracking, environmental monitoring, military surveillance, and health care. Also, WSNs can use several different wireless technologies including IEEE 802.11 WLANs [1], IEEE 802.15.4 WPAN [2], RFID [3][4], body area networks (BAN) [5][6], and smart utility networks (SUN) [7].

The design of medium access control (MAC) protocols in WSNs is challenging due to the inherent characteristics that distinguish these networks from traditional wireless networks such as cellular networks or mobile *ad hoc* networks. These issues include asymmetric communication patterns (from many nodes to a sink node), limited energy resources, limited processing and memory capabilities, redundant data, and scalability [1][8]. Due to the ultra power and resource limitations of WSNs, efforts have been made to improve energy efficiency in MAC protocol design, and *duty cycling*, in which each node periodically cycles between the active and sleep states, has been introduced to reduce unnecessary energy applications over a wide variety of traffic conditions in multihop communication. Moreover, the end-to-end delivery delay should be decreased for efficient data delivery over a wide variety of traffic conditions in multihop communication networks.

Previous proposed MAC protocols in WSNs generally use a fixed duty cycle to improve energy savings. This fixed duty cycle mechanism has two drawbacks. One is that it increases delay in multihop data delivery and wastes unnecessary energy due to idle listening. The other is that it can not deal with the traffic fluctuations. In addition, there is a tradeoff between energy efficiency and delay caused by duty cycling mechanism in multi-hop communication and existing MAC approaches only tend to improve energy efficiency with sacrificing data delivery delay or otherwise. Therefore, previous proposed schemes are not actually efficient for WSNs.

In this paper, two different energy-efficient MAC schemes are proposed: adaptive duty cycling synchronous-MAC (ADS-MAC) and enhanced low power listening asynchronous-MAC (ELA-MAC). The proposed MAC protocols have several contributions. First, our proposed protocols decrease data delivery delay without sacrificing energy consumption by using the closed-loop control for adaptive duty cycling. Second, the enhanced control packet mechanisms such as a PRTS in ADS-MAC and preloads in ELA-MAC can reduced the unnecessary energy consumption. Third, our protocols can handle the traffic fluctuations by the adaptive duty cycling technique. Consequently, our proposed schemes adjust duty cycles for varying traffic conditions and perform adaptive duty cycling to accomplish high energy efficiency and low latency over traffic fluctuations in WSNs. Furthermore, we analyze the performance of MAC protocols in WSNs in terms of energy consumption and end-to-end delivery delay. In Section 2, we briefly review related works for existing MAC protocols in WSNs. In Section 3, the proposed schemes are described in detail. Section 4 and Section 5 present performance analyses and evaluations of proposed protocols in terms of energy consumption and end-to-end delivery delay. Finally, we conclude this work in Section 6.

2. Related Work

The existing contention-based MAC schemes in WSNs can be divided into two categories: synchronous and asynchronous. Synchronous MAC approaches, like S-MAC [8] and T-MAC [9], share schedule information that specifies the cycles of the active and sleep periods via synchronization packets. Nodes synchronize with neighboring nodes through the synchronization (SYNC) packet exchange, and govern communication during the common active and sleep periods.

In S-MAC, at the beginning of the active period, the node exchanges SYNC information with its neighbors to assure that the node and its neighbors wake up concurrently. This schedule is only adhered to locally, resulting in a virtual cluster, which mitigates the need for synchronization of all nodes in the networks. Nodes that lie on the borders of two clusters maintain the schedules of both clusters, which in turn maintains connectivity across the network. After the synchronization information is exchanged, the nodes send data packets using request-to-send (RTS)/clear-to-send (CTS) until the end of the active period, and then the nodes enter sleep mode. S-MAC achieves energy savings with periodic sleep intervals, and can decrease the energy wastage caused by idle listening. Although S-MAC improves the energy efficiency, it causes delay in multihop data delivery and wastes energy due to fixed duty cycling.

T-MAC, which is inspired by S-MAC, goes into the sleep state when no activation event has occurred for a certain time (*TA*), and enhances the power saving. In contrast to S-MAC, it operates with fixed length slots and uses a time-out mechanism to dynamically determine the end of the active period. The time-out value (*TA*) is set to span a small contention period and an RTS/CTS exchange. If a node does not detect any activity (an incoming message or a collision) within the time-out interval, it assumes that no neighbor wants to communicate with it and goes to sleep. On the other hand, if the node engages or overhears a communication, it simply starts a new time-out after that communication finishes. To save energy, the node turns off its radio while waiting for other communications to finish (overhearing avoidance). T-MAC adjusts to traffic fluctuations in the network and improves energy efficiency more than S-MAC. However, it still suffers from energy waste due to fixed duty cycling with the fixed time-out mechanism.

E^2-MAC [10] is similar to T-MAC and buffer based synchronous MAC schemes. Data is transferred in E^2-MAC if the buffer value exceeds the specified threshold value. Otherwise, the timer is set to a smaller value than in T-MAC. Since the protocol switches to sleep mode when the node does not transmit an RTS control packet or data during the timer operation, it improves energy efficiency compared with S-MAC and T-MAC. However, its buffer scheme may increase the end-to-end delay due to traffic conditions, and cannot guarantee fast data delivery in realtime applications.

On the other hand, asynchronous MAC protocols, like B-MAC (LPL) [11], WiseMAC [12], and X-MAC [13], do not exchange synchronization information to send or receive data. Alternatively, preamble sampling is utilized for data forwarding in asynchronous MAC protocols.

B-MAC utilizes a long preamble to achieve low power communication. If a node has data to send, it sends the data packet following a long preamble that is slightly longer than the sleep period of the receiver. During the active period, the node samples the channel, and if a preamble is detected, it remains awake to receive the data. With the inclusion of a long preamble, a sender is assured that at some point during the preamble, the receiver will wake up, detect the preamble, and remain awake in order to receive the data. This is called low power listening (LPL). The major advantage of LPL is that it minimizes the active period when there is no traffic. While B-MAC performs quite well, it suffers from an overhearing problem, in that receivers that are not the target of the sender also wake up during the long preamble and have to stay awake until the end of the preamble to find out if the packet is destined for them. This wastes energy at all non-target receivers within the transmission range of the sender, so the long preamble dominates energy usage and increases the per-hop latency.

WiseMAC uses a similar method to B-MAC, but the sender learns the schedules of the receiver wake-up periods by using time division multiple access (TDMA) as the control channel, and schedules its transmissions to reduce the length of the extended preamble. WiseMAC protocol uses nonpersistent carrier sense multiple access (CSMA) with preamble sampling to decrease idle listening. In the preamble sampling technique, a preamble precedes each data packet to alert the receiving node. All of the nodes in a network sample the medium with a common period, but their relative schedule offsets are independent. If a node finds the medium busy after it wakes up and samples the medium, it continues to listen until it receives a data packet or the medium becomes idle again. The size of the preamble is initially set to be equal to the sampling period. However, the receiver may not be ready at the end of the preamble, due to factors such as interference, which causes the possibility of overemitting-type energy waste. Moreover, overemitting is increased with the length of the preamble and the data packet, since no handshake occurs with the intended receiver.

In X-MAC, short preambles with target address information are used instead of a long preamble, unlike B-MAC, so the overhearing problem of B-MAC is solved. When a receiver wakes up and detects a short preamble, it looks at the target address that is included in the preamble. If the node is the intended recipient, it stays awake for the incoming data; otherwise, the node goes to sleep again immediately. This mechanism can decrease per-hop latency and the energy spent unnecessarily waiting and transmitting. This approach is simple to implement and achieves low power communication; however, it becomes less energy efficient and cannot guarantee on the worst-case delay as the traffic load increases.

There are several MAC protocols have also been proposed specifically fo BANs. Body sensor network MAC (BAN-MAC) [5] is a dedicated ultra-low-power MAC protocol designed for star topology BANs. BAN-MAC is compatible with IEEE 802.15.4, and accommodates unique requirements of the biosensors in BANs. By exploiting feedback information from distributed sensors in the network, BAN-MAC adjusts protocol parameters dynamically to achieve best energy conservation on energy critical sensors. BodyQoS [5] aims to provide QoS in body sensor networks with prioritized data stream service, asymmetric QoS framework, radio-agnostic QoS, and Adaptive Bandwidth Scheduling. It receives QoS and data transmission requests from the transport layer and uses the underlying MAC protocol to transmit data. BodyQoS adopts an asymmetric architecture. BodyQoS has an effective mechanism for prioritizing requests to maximize satisfaction. Medical Medium Access Control (MedMAC) [6] protocol for energy efficient and adaptable channel access in body area networks. The MedMAC incorporates a novel synchronisation mechanism, which facilitates contention free TDMA channels, and initial power efficiency. MedMAC outperforms IEEE 802.15.4 in terms of power efficiency in low and medium data rate medical applications.

3. Proposed Protocols

3.1 Network Model

WSN applications can be characterized by several traffic features, such as periodic traffic, event-triggered traffic, and quality of service (QoS)-required traffic. In QoS-required traffic conditions like high traffic congestion or mobility, a specific mechanism should be supported for managing those traffic conditions in order to achieve end users' satisfaction with the services that the system provides [14][15][16]. In this paper, we consider low/moderate traffic conditions and mobility, which are the most common traffic features in event-triggered traffic applications of WSNs. There is a tradeoff between energy efficiency and delivery delay, but only one of these factors has been considered in most previous research. Therefore, energy efficiency and delivery delay should be considered together for better performance gains.

There can be multiple sink nodes in WSNs, but each sensor node is assigned to only one sink node, since asymmetric communications generally occur in many-nodes-to-one-sink communication patterns. We assume that the network comprises randomly distributed nodes with a single sink node, and data aggregation is not considered here. We also consider that all of the sensor nodes have the same transmission range and initial battery power capability.

3.2 Adaptive Duty Cycling Synchronous MAC

As shown in Fig. 1, the active period ends when no activation event has occurred during the time-out period (T_A) in ADS-MAC. In other words, if there are no data to send or receive until timer expiration, the node goes to sleep to avoid unnecessary idle listening. Furthermore, the per-hop delay can be increased in the unidirectional multi-hop communication pattern from the sources to the destinations. This long delay problem can be solved by using a preoccupancy request-to-send (PRTS) packet with an adaptive timer T_A to adapt to the traffic conditions in ADS-MAC. This PRTS includes the current sender address, the next-hop node address, the duration of the transmission, and the address of the final destination of the current data flow. The data transmission is prescheduled to multi-hops by the PRTS and T_A so that data delivery delay can be reduced without sacrificing energy efficiency.

The adaptive timer T_A is employed in proportion to $H(i)$ with an average delay level in ADS-MAC, where $H(i)$ is the hop count (distance) of a node i from the destination node. The adaptive timer of each node i, $T_A(i)$, is expressed in ADS-MAC as follows:

$$T_A(i) = \begin{cases} T_{\min} + \gamma(T_{extension}) \times (1 - \frac{H(i)}{H_{total}}), \\ if\ E(i) > E_{\min}, H(i) \le H_{total} - 2 \\ T_{\min}, \qquad otherwise \end{cases} \tag{1}$$

where $T_{min} = Ct + R + T$, Ct is the contention interval, R is the length of an RTS packet, T is the short turnaround time $T_{extension} = T_{max} - T_{min}$, T_{max} is the active interval of the current complete cycle, γ is the delay parameter for traffic fluctuations, $H(i)$ is the hop count (distance) of a node i from the destination node, H_{total} indicates the total hops between the source and destination node, $E(i)$ is the residual energy level of a node i, and E_{min} is the lower bound on the residual energy threshold. $H(i)$ and H_{total} are obtained through the routing information.

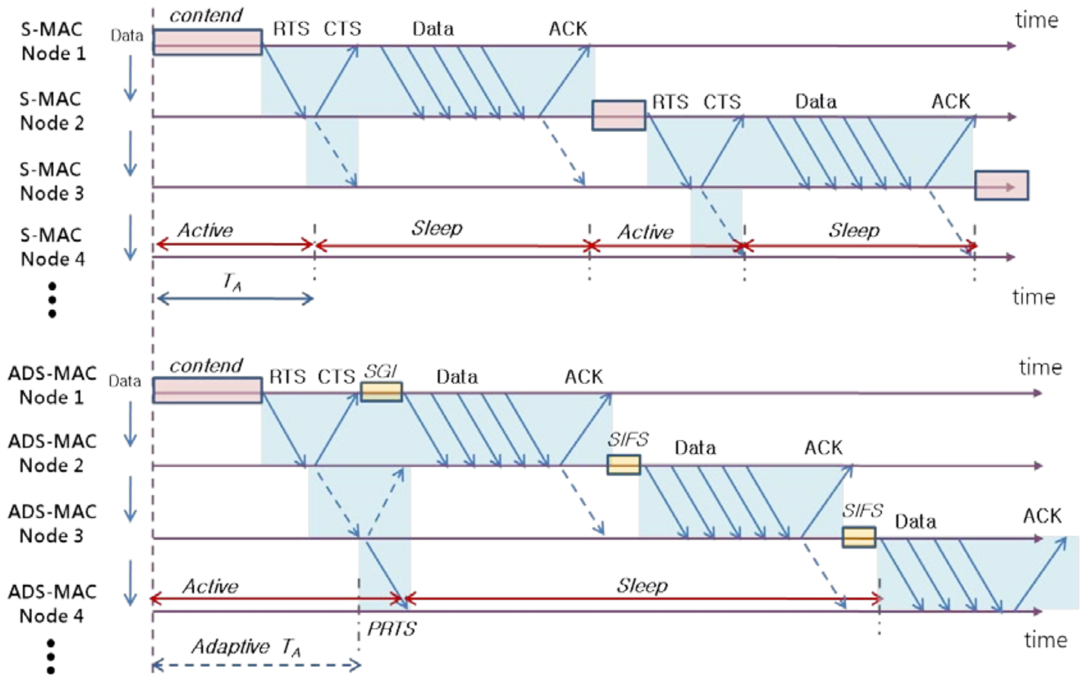

Fig. 1. Adaptive duty cycling scheme in ADS-MAC.

$$\gamma = \begin{cases} 0.5, & if\ D_{avg} < D_{\min} \\ 1.0, & if\ D_{\min} < D_{avg} < D_{\max} \\ 2.0, & if\ D_{\max} < D_{avg} \end{cases} \quad (2)$$

where D_{min} is the lower bound on the delivery delay threshold, D_{avg} is the measured average delay threshold, and D_{max} is the upper bound on the delivery delay threshold. This average delivery delay means an approximate estimation of the current traffic condition, and it is utilized for the delay parameter γ. The values of γ indicate a decrease in the current duty cycle, maintaining the current duty cycle, or an increase in the current duty cycle, respectively. In other words, the 0.5, 1.0, or 2.0 value of γ indicates halving the current duty cycle, maintaining the current duty cycle, or doubling the current duty cycle, respectively. These values are empirically chosen based on the SMAC implementation in [8]. The further study on the values of the delay parameter is our future work.

D_{avg} is the average level of all one-hop measured delay values collected by the sink node (final destination) in the current synchronization period, and is finally broadcasted by the SYNC packet. A one-hop delay is defined as the time difference between the time at which a packet enters the queue and the time at which it is successfully sent at a node. This delay value is recorded in the packet header by each node. Because the delay value is measured at each node, no synchronization problems are related to it. A node may maintain different schedules if it receives packets coming from different sources in a multi-hop network. In such cases, the nodes on the borders of different schedules will adopt those schedules together. However, this situation originates from routing problems, so schedule maintenance can be optimized through routing information or algorithms. In our implementation, each node is able to maintain three different schedules at most, so the adjustment procedure will not cause any synchronization problems for the nodes.

Recall that this adaptive timer is only activated if the residual energy level of a node i, $E(i)$, is larger than the residual energy threshold, E_{min}, which implies that the node does not have enough energy to take part in data transmission jobs with other nodes. If $E(i) \le E_{min}$, T_{min} is set to the node. The adaptive time-out scheme can efficiently reduce the end-to-end delivery delay and improve the energy savings, since the PRTS with T_A can quickly deliver data to multi-hops in a single complete cycle. Initially, synchronization is accomplished by synchronization packets from the network coordinator. Each node then sets up T_A through the SYNC packets. **Fig. 1** briefly shows the ADS-MAC operation compared with S-MAC. Node 1 operates as in typical contention-based MAC protocols. When node 3 overhears a CTS from node 2, it can go to sleep until the next data transmission. After this, the data are directly transferred from node 2 to node 3. Upon overhearing a CTS from node 2, node 3 immediately sends a PRTS to node 4 and goes to sleep until the next data communication. Node 4 can recognize that there are data destined for it and the duration of the data transmission through the PRTS, so it can go to sleep until the next data transmission like node 3. Note that when node 4 wakes up, it directly receives data from node 3 as in the previous procedure. Consequently, this mechanism improves energy efficiency and per-hop latency. If a PRTS is lost, there is no additional retry process in the current duty cycle. The node will simply start the new MAC operation with PRTS in the next duty cycle.

3.3 Enhanced Low Power Listening Asynchronous MAC

As shown in **Fig. 2**, ELA-MAC employs the successive preloads approach by transmitting a series of preload packets, each containing the address of the receiver node, the duration of the data transmission, and the request sampling period (*RSP*), instead of using one long preamble. A series of preloads prevents the non-receiver nodes from consuming energy for overhearing. In addition, the receiver node can go to sleep until the next data transmission, and then the receiver wakes up to receive data. This mechanism can reduce the unnecessary energy consumption for both the non-receiver nodes and the receiver node. After the data transmission, the receiver sends an acknowledge (ACK) packet back to the sender to guarantee the success of the data transmission. The *RSP* field informs the receiver and the neighbors about the channel sampling period that is to be changed. This change in the channel sampling period changes the length of the preload frame, and as a result, the per-hop delay can be decreased. The new value of the channel sampling period in the *RSP*, Δ_N, is defined in ELA-MAC as follows:

$$\Delta_N = \begin{cases} 0.5 \times \Delta_C, & if\ D_{avg} < D_{\min} \\ 1.0 \times \Delta_C, & if\ D_{\min} < D_{avg} < D_{\max} \\ 2.0 \times \Delta_C, & if\ D_{\max} < D_{avg} \end{cases} \quad (3)$$

where Δ_C is the current value of the channel sampling period. D_{min}, D_{avg}, and D_{max} have the same parameters as in ADS-MAC. The value of Δ_N indicates adjustment of the current duty cycle, like the value of γ in ADS-MAC. The average delivery delay is the average level of all one-hop measured delay values collected in the specific period, and this average delivery delay indicates an approximate estimation of the current traffic condition, like in ADS-MAC. This delay value is recorded in the packet header by each node and is collected by the sink node (final destination). The difference in D_{avg} between ELA-MAC and ADS-MAC is that in ELA-MAC, D_{avg} is obtained through ACK messages from the sink node (final destination) to the nodes, because ELA-MAC is an asynchronous scheme.

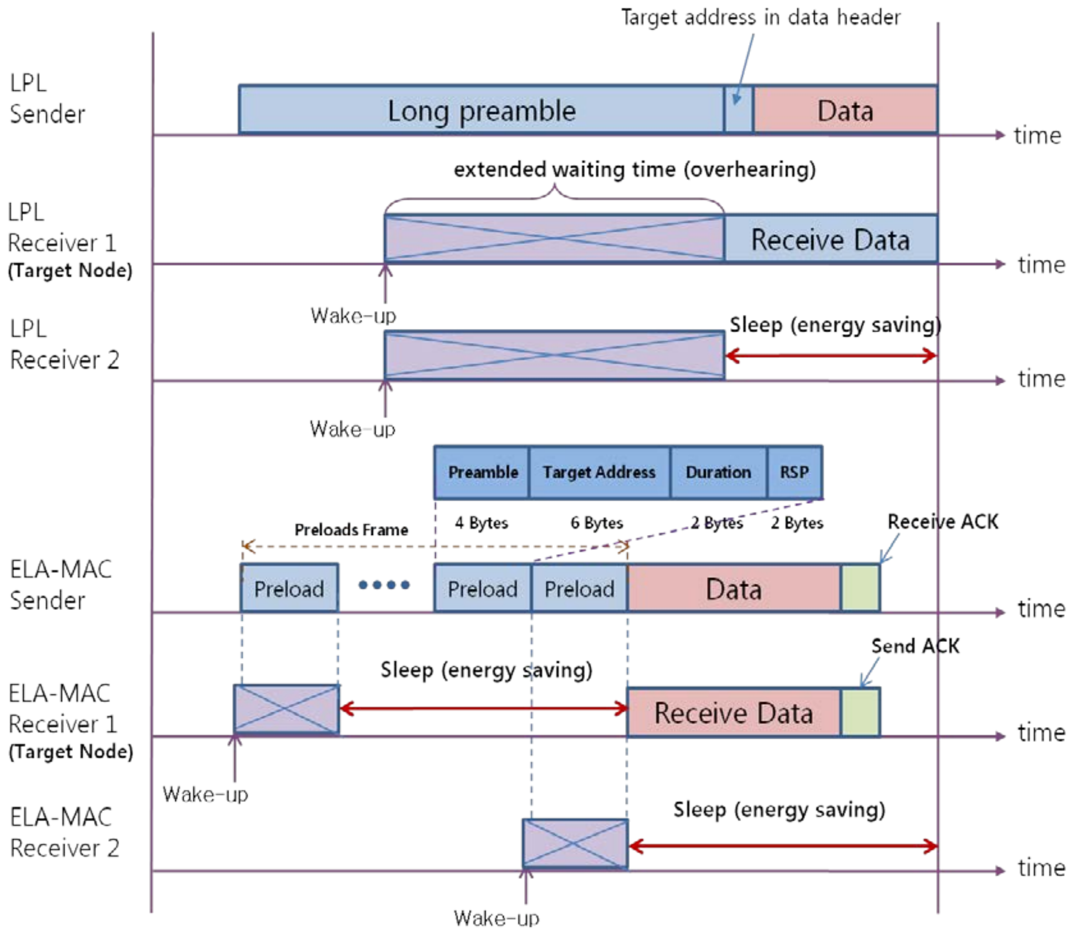

Fig. 2. ELA-MAC operation.

If a node has burst data to send and requests the receiver nodes to change the current channel sampling period, the node sends a series of preloads, including the new value of the channel sampling period, Δ_N, in the RSP field, before sending the data. When the receiver obtains Δ_N from the RSP, it changes the current channel sampling period Δ_C to Δ_N, and after receiving data from the sender, it sends an ACK to the sender. The ACK plays two important roles: it confirms the channel sampling period change and guarantees the data reception. After the sender receives the ACK, it changes the length of the preload frame by the new value of the channel sampling period for the next data communication. Also, the receiver nodes adopt the new channel sampling period. This mechanism can diminish the end-to-end delivery delay in the end.

In X-MAC [13], short strobed preambles are used for low power listening (LPL), but this technique is not efficient for LPL and energy savings. A sender and receiver can have perfect wake-up schedules that are different from each other in case the channel sampling period of the receiver is longer than the length of the short strobed preambles, so the performance of X-MAC may become seriously deteriorated. However, ELA-MAC can avoid this problem by using the preloads frame. Therefore, ELA-MAC achieves ultra LPL for better energy efficiency and accomplishes lower latency compared with other existing LPL schemes.

4. Performance Analysis

In this section, we analyze the average energy consumption and end-to-end delivery delay of ADS-MAC and ELA-MAC compared with other existing MAC protocols. We assume that all nodes can hear each other directly, and that each node has n neighbor nodes. Also, we suppose there are N hops from the source to the sink. In our analysis, energy consumption by the radio turn on/off and delivery delay by the packet transmission is only focused and we ignore energy consumption and delivery delay caused by the processing, queuing, or sensing data. The energy consumption of the radio is determined by how much time it spends transmitting (E_{tx}), receiving (E_{rx}), overhearing (E_{over}), idle listening (E_{idle}), channel sampling (E_{sample}), contending ($E_{contend}$), and sleeping (E_{sleep}). The delivery delay is decided by how long it spends carrier sensing (D_{cs}), contending ($D_{contend}$), channel sampling (D_{sample}), sleeping (D_{sleep}), sending control packets (D_{ctrl} and D_{ack}), preambles ($D_{preamble}$, $D_{spreamble}$, and $D_{preloads}$), and data for a single packet (D_{data}). For synchronous MAC protocols, the expected energy consumption is the sum of the expected energy spent in each state: $E = E_{contend} + E_{tx} + E_{rx} + E_{idle} + E_{sleep}$. The expected delivery delay is the sum of the expected time spent in each state: $D = D_{contend} + D_{ctrl} + D_{data} + D_{ack} + D_{sleep}$. For asynchronous MAC protocols, the expected energy consumption is the sum of the expected energy spent in each state: $E = E_{cs} + E_{tx} + E_{rx} + E_{over} + E_{sample} + E_{sleep}$. The expected delivery delay is the sum of the expected time spent in each state: $D = D_{cs} + D_{preamble} + D_{data} + D_{ack}$. **Table 1** summarizes all of the key parameters used in our analyses and simulations with typical values of CC1000 (Mica2) [17][18] and CC2420 (MicaZ) [17][18].

Table 1. Parameters for analysis and simulation in Chipcon CC1000 (Mica2) [17][18] and CC2420 (MicaZ) [17][18].

Symbol	Meaning	CC1000	CC2420
P_{tx}	Power in transmitting	31.2 mW	52.2 mW
P_{rx}	Power in receiving	22.2 mW	56.4 mW
P_{listen}	Power in listening (idle)	22.2 mW	56.4 mW
P_{sleep}	Power in sleeping	3 µW	3 µW
P_{sample}	Power in channel sampling	7.4 mW	12.3 mW
t_{spl}	Average time to sample channel	3 ms	2.5 ms
t_{csl}	Average carrier sense time	7 ms	2 ms
t_B	Time to Tx/Rx a byte	416 µs	32 µs
T_P	Channel sampling period	Varying	Varying
r_{data}	Data packet rate	Varying	Varying
L_{data}	Data packet length	50 bytes	50 bytes
L_{spr}	Short preamble length	10 bytes	10 bytes
t_{gap}	ACK listen interval in X-MAC	14 bytes	14 bytes
L_{pl}	Preload packet length	4.16 ms	0.32 ms
L_{pls}	Preloads frame length	Varying	Varying
L_{ack}	ACK packet length	10 bytes	10 bytes
L_{ctrl}	Control packet length	10 bytes	10 bytes
L_{prts}	PRTS packet length	14 bytes	14 bytes
t_{ct}	Average contention period time	9.15 ms	9.15 ms
t_A	Listen interval in S-MAC	115 ms	115 ms
t_O	Timeout interval in T-MAC	15 ms	15 ms
T_f	Length of a complete cycle	Varying	Varying
Duty Cycle	Duty cycle in synchronous MACs	10 %	10 %
T_{SYNC}	Period to send a SYNC	10 cycles	10 cycles

t_{SGI}	Average short guard interval	4.16 ms	0.32 ms
t_{SIFS}	Average SIFS interval	5 ms	192 μs
T_{min}	Minimum timeout interval in ADS-MAC	10 ms	10 ms

4.1 Energy Consumption

In analyses of energy consumption, all of the parameters are normalized to one second. In S-MAC, data can be transmitted for the fixed active period. The expected energy consumption of S-MAC without adaptive listen is

$$\begin{aligned} E_S &= E_{contend} + E_{tx} + E_{rx} + E_{idle} + E_{sleep} \\ &= P_{tx}(L_{data} + 2L_{ctrl} + L_{ack})t_B r_{data} \\ &+ P_{rx}\{t_{ct} + (n-1)t_A + (L_{data} + 2L_{ctrl} + L_{ack})t_B\}r_{data} \\ &+ P_{sleep}[1 - r_{data}\{t_{ct} + (n-1)t_A + 2(L_{data} + L_{ack} + 2L_{ctrl})t_B\}] \end{aligned} \tag{4}$$

where $P_{listen} = P_{rx}$. T-MAC goes to sleep if no activation event has happened for a certain time (t_O). The expected energy consumption of T-MAC is

$$\begin{aligned} E_T &= E_{contend} + E_{tx} + E_{rx} + E_{idle} + E_{sleep} \\ &= P_{tx}\{L_{data} + \frac{(6+k)}{2}L_{ctrl} + L_{ack}\}t_B r_{data} \\ &+ P_{rx}\{t_{ct} + (n-1)t_O + (L_{data} + L_{ack} + \frac{(6+k)}{2}L_{ctrl})t_B\}r_{data} \\ &+ P_{sleep}[1 - r_{data}\{t_{ct} + (n-1)t_O + 2(L_{data} + \frac{(6+k)}{2}L_{ctrl} + L_{ack})t_B\}] \end{aligned} \tag{5}$$

where k is the number of transmitting FRTSs in a complete frame of T-MAC. In ADS-MAC, data are sent with the adaptive timeout period with T_{ADS}. The expected energy consumption of ADS-MAC is

$$\begin{aligned} E_{ADS} &= E_{contend} + E_{tx} + E_{rx} + E_{idle} + E_{sleep} \\ &= P_{tx}\{L_{data} + L_{ack} + \frac{(4+kh)}{2h}L_{ctrl}\}t_B r_{data} + P_{rx} r_{data}\{t_{ct} + \frac{(n-1)}{r_{data}}T_{ADS} \\ &+ \frac{2}{h}t_{SGI} + \frac{(h-1)}{h}t_{SIFS} + t_B(L_{data} + L_{ack} + \frac{(4+kh)}{2h}L_{ctrl})\} \\ &+ P_{sleep}[1 - r_{data}\{t_{ct} + \frac{(n-1)}{r_{data}}T_{ADS} + \frac{2}{h}t_{SGI} + \frac{(h-1)}{h}t_{SIFS} \\ &+ 2(L_{data} + L_{ack} + \frac{(4+kh)}{2h}L_{ctrl})t_B\}] \end{aligned} \tag{6}$$

where h is the number of forwarding hops in a complete frame of ADS-MAC and T_{ADS} is the expected timeout interval in ADS-MAC.

LPL (B-MAC) sends a long preamble before each data packet. The duration of the preamble is at least the same as during the sampling period, T_P. The length of the preamble is $L_{preamble} = T_P / t_B$. A node will periodically receive n packets from n neighbors. Among them, only one packet is destined for the node and the rest are overhearing packets. The average time of the received preamble for each packet is $T_P / 2$. The expected time of the carrier sense is $t_{cs} = t_{csl} r_{data}$. Therefore, the expected energy consumption of LPL (B-MAC) is

$$\begin{aligned} E_{LPL} &= E_{cs} + E_{tx} + E_{rx} + E_{over} + E_{sample} + E_{sleep} \\ &= P_{listen} t_{cs} + P_{tx} t_{tx} + P_{rx} t_{rx} + P_{rx} t_{over} + P_{sample} t_{sample} + P_{sleep} t_{sleep} \\ &= \{P_{tx}(T_P + L_{data} t_B) + P_{rx}(\frac{n}{2} T_P + L_{data} t_B + t_{csl})\} r_{data} \\ &+ P_{sleep}\{1 - (t_{csl} + \frac{(n+2)}{2} T_P + 2L_{data} t_B) r_{data} - \frac{t_{spl}}{T_P}\} + \frac{P_{sample} t_{spl}}{T_P} \end{aligned} \quad (7)$$

In X-MAC, a series of short preambles is sent before each data packet. The expected short preamble length and the expected ACK listen interval at a sender are $(m+1)L_{spr} / 2$ and $mt_{gap} / 2$, respectively, where m is the expected number of iterations required for each data packet, L_{spr} is the length of a short preamble, and t_{gap} is the ACK listen interval between short preambles. The average length of the received short preambles for each data packet is $3L_{spr} / 2$, and the average time of the ACK listen interval at a receiver is $t_{gap} / 2$. The expected time of the transmitting state and the receiving state, respectively, are

$$t_{tx} = [\{\frac{(m+1)}{2} L_{spr} + L_{data}\} t_B + \frac{m}{2} t_{gap}] r_{data} \quad (8)$$

$$t_{rx} = \{(\frac{3}{2} L_{spr} + L_{data}) t_B + \frac{1}{2} t_{gap}\} r_{data} \quad (9)$$

where N_{itern} is the expected short preamble iteration required for each data packet. In addition, the expected time of the overhearing state and sleeping state, respectively, are

$$t_{over} = (n-1)(\frac{3}{2} L_{spr} t_B + \frac{1}{2} t_{gap}) r_{data} \quad (10)$$

$$t_{sleep} = 1 - t_{cs} - t_{tx} - t_{rx} - 2t_{ack} - t_{over} - t_{sample} \quad (11)$$

Moreover, the expected time in waiting for an ACK packet is $t_{sc} = t_{scl} r_{data}$, and the expected time in transmitting/receiving an ACK packet is $t_{ack} = L_{ack} t_B r_{data}$. The expected energy consumption of X-MAC is

$$\begin{aligned} E_X &= E_{cs} + E_{tx} + E_{rx} + E_{over} + E_{sample} + E_{sleep} \\ &= P_{rx}(t_{cs} + t_{ack} + t_{rx} + t_{over}) + P_{tx}(t_{tx} + t_{ack}) + P_{sample} t_{sample} + P_{sleep} t_{sleep} \\ &= P_{rx}\{t_{csl} + (\frac{3n}{2} L_{spr} + L_{data} + L_{ack}) t_B + \frac{(m+n)}{2} t_{gap}\} r_{data} + \frac{P_{sample} t_{spl}}{T_P} \\ &+ P_{tx}\{\frac{(m+1)}{2} L_{spr} + L_{data} + L_{ack}\} t_B r_{data} + P_{sleep}[1 - t_{csl} r_{data} - \frac{t_{spl}}{T_P} \\ &- \{(\frac{(m+3n+1)}{2} L_{spr} + 2L_{data} + 2L_{ack}) t_B + \frac{(m+n)}{2} t_{gap}\} r_{data}] \end{aligned} \quad (12)$$

In ELA-MAC, successive preload messages are sent before each data packet. The preload period is at least the same as the channel sampling period T_P. The length of the preload frame is $L_{pls} = T_P / t_B$. The average length of the received preload for each data packet is $3L_{pl} / 2$. The expected energy consumption is

$$\begin{aligned} E_{ELA} &= E_{cs} + E_{tx} + E_{rx} + E_{over} + E_{sample} + E_{sleep} \\ &= P_{rx}(t_{cs} + t_{ack} + t_{rx} + t_{over}) + P_{tx}(t_{tx} + t_{ack}) + P_{sample}t_{sample} + P_{sleep}t_{sleep} \\ &= P_{rx}\{t_{csl} + (\frac{3n}{2}L_{spr} + L_{data} + L_{ack})t_B\}r_{data} \\ &+ \frac{P_{sample}t_{spl}}{T_P} + P_{tx}\{(L_{data} + L_{ack})t_B + T_P\}r_{data} \\ &+ P_{sleep}\{1 - (\frac{3n}{2}L_{pl} + 2L_{data} + 2L_{ack})t_B r_{data} - (t_{csl} + T_P)r_{data} - \frac{t_{spl}}{T_P}\} \end{aligned} \tag{13}$$

To minimize the energy consumption, given a fixed n and r_{data}, we can obtain the optimal value by solving the following equation:

$$\frac{dE_{ELA}}{dT_P} = 0 \tag{14}$$

According to Eq. (13) and Eq. (14), the optimal T_P for ELA-MAC can be found as

$$T_P{}^* = \sqrt{\frac{(P_{sample} - P_{sleep})t_{spl}}{(P_{tx} - P_{sleep})r_{data}}} \tag{15}$$

4.2 Delivery Delay

To calculate delivery delay, unlike energy consumption analysis, we do not adopt normalized parameters. All of the parameters imply the average delays in each state. All nodes are synchronized duty cycles for synchronous MAC approaches, so data can be transmitted for a fixed active period in S-MAC. According to [2], the expected delivery delay of S-MAC without adaptive listen over N hops is

$$\begin{aligned} D_S(N) &= D_{contend} + D_{ctrl} + D_{data} + D_{ack} + D_{sleep} \\ &= NT_f + t_{ct} + t_{tx} - T_f / 2 \\ &= NT_f + t_{ct} + (L_{data} + 2L_{ctrl} + L_{ack})t_B - T_f / 2 \end{aligned} \tag{16}$$

where T_f is the length of a complete cycle of listen and sleep. In T-MAC, data can be sent with the timer TA and FRTS for multi-hop data transmission. For simplicity, we assume that the packet will travel over three hops in just one frame for T-MAC. The expected delivery delay of T-MAC over N hops is

$$\begin{aligned} D_T(N) &= D_{contend} + D_{ctrl} + D_{data} + D_{ack} + D_{sleep} \\ &= NT_f / 3 + 3t_{ct} + 3t_{tx} - T_f / 2 \\ &= NT_f / 3 + 3t_{ct} + 3(L_{data} + 3L_{ctrl} + L_{ack})t_B - T_f / 2 \end{aligned} \tag{17}$$

In ADS-MAC, data are sent for the adaptive active period with T_A and PRTS for multi-hop data transmission. For simplicity, we assume that the packet will travel over four hops in just one frame for ADS-MAC. The expected delivery delay of ADS-MAC over N hops is

$$
\begin{aligned}
D_{ADS}(N) &= D_{contend} + D_{ctrl} + D_{data} + D_{ack} + D_{sleep} \\
&= NT_f/4 + t_{ct} + 4t_{tx} - T_f/2 \\
&= NT_f/4 + t_{ct} + t_{SGI} + 3t_{SIFS} + (4L_{data} + 2L_{ctrl} + 4L_{ack})t_B - T_f/2
\end{aligned} \tag{18}
$$

Unlike synchronous MAC approaches, all nodes have their own duty cycles in asynchronous MAC approaches. In addition, if a node has data to send, it immediately wakes up and sends its data to the destination. LPL (B-MAC) sends a long preamble before each packet. The expected delivery delay of LPL (B-MAC) over *N* hops is

$$
\begin{aligned}
D_{LPL}(N) &= D_{cs} + D_{preamble} + D_{data} \\
&= t_{cs} + Nt_{tx} \\
&= t_{cs} + N(T_P + L_{data}t_B)
\end{aligned} \tag{19}
$$

X-MAC sends a series of short preambles before each data packet. The expected delivery delay of X-MAC over *N* hops is

$$
\begin{aligned}
D_X(N) &= D_{cs} + D_{spreamble} + D_{gap} + D_{data} \\
&= t_{cs} + Nt_{tx} \\
&= t_{cs} + N[\{(m+1)L_{spr}/2 + L_{data}\}t_B + mt_{gap}/2]
\end{aligned} \tag{20}
$$

In ELA-MAC, successive preload messages are sent before each data packet. The expected delivery delay of ELA-MAC over *N* hops is

$$
\begin{aligned}
D_{ELA}(N) &= D_{cs} + D_{preloads} + D_{data} + D_{ack} \\
&= t_{cs} + Nt_{tx} \\
&= t_{cs} + N\{T_P + (L_{data} + L_{ack})t_B\}
\end{aligned} \tag{21}
$$

5. Simulation Results

In this section, we evaluate the performance of ADS-MAC and ELA-MAC compared with other existing MAC protocols in terms of energy consumption and delivery delay. In our simulations of ADS-MAC and ELA-MAC, we used the ns-2 network simulator (version 2.29) [19], with the standard combined free space and two-ray ground reflection radio propagation model with a single omnidirectional antenna for each node. In the simulation, 1000 nodes are randomly distributed over a 1000m x 1000m area, and a packet is generated every 1 second by each source node and sent to the sink node (r_{data} = 1 for periodic model). The transmission range of each sensor node is 50m. The other parameters are shown in **Table 1**. Additionally, D_{max} = 3s, D_{min} = 1s, and E_{min} = 20% of the initial power capacity for ADS-MAC. D_{max} = 500ms, D_{min} = 100ms, and an average period of T_p = 100ms are set for ELA-MAC. To simplify our evaluation, we do not include the routing traffic in the simulations, and we assume that there is a routing protocol deployed to provide the shortest path between any two nodes. Furthermore, the simulation was repeated over 50 runs, and the average results were obtained and are shown here.

5.1 Energy Consumption

The cumulative energy consumption is represented as increases in the number of hops in **Fig. 3**, and the network lifespan is shown in **Fig. 4**. The network lifespan is determined by the time until the first sensor node depletes all of its energy. These results reveal that asynchronous MAC protocols provide greater energy savings than synchronous MAC protocols, since synchronous MACs expend additional energy on SYNC/RTS/CTS handshakes and idle listening caused by the listen intervals of the periodic duty cycles. Instead, asynchronous MACs achieve low power listening by using preambles to conserve energy consumption. T-MAC offers better energy efficiency than S-MAC due to its adaptivity, but through the adaptive timeout (T_A) and enhanced control packet mechanisms, data are efficiently forwarded via adaptive duty cycling in ADS-MAC. In addition, the energy efficiency of ELA-MAC is the best because of its ultra LPL and preload technique. Moreover, the adaptive channel sampling period can be supported by the RSP in ELA-MAC. Consequently, our schemes save more energy and provide better energy efficiency than existing schemes, resulting in prolonged network lifetimes.

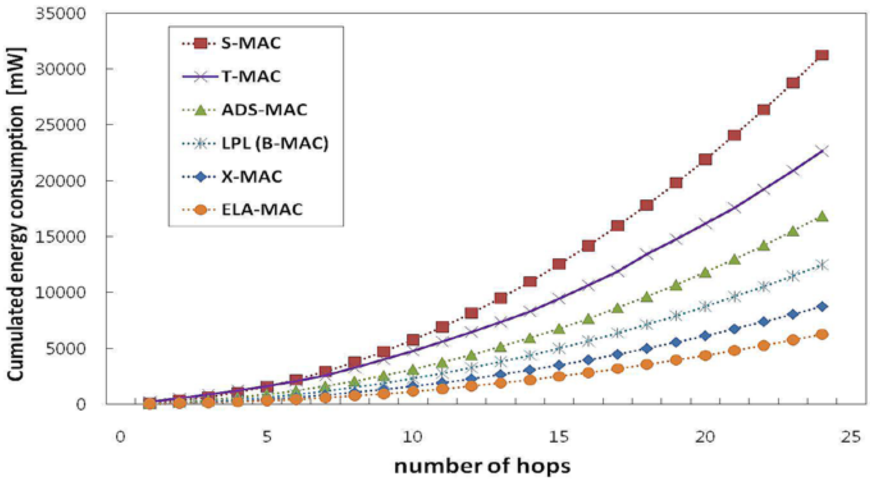

Fig. 3. Cumulated energy consumption in the periodic model.

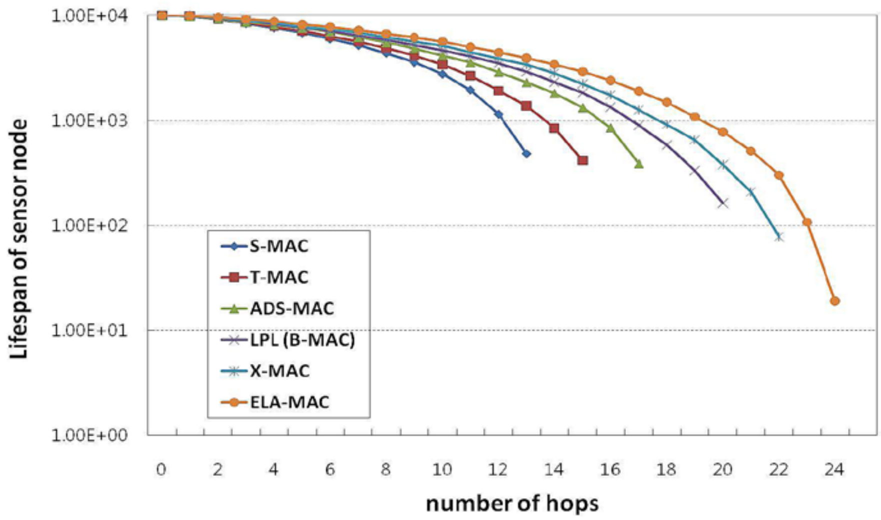

Fig. 4. Network lifespan.

For event-driven model, four source nodes generate data packets simultaneously and send them to the sink node in **Fig. 5**. The cumulative energy consumption is shown as more increases in the number of hops in **Fig. 5** because more traffic is generated in the network. These results show that asynchronous MAC protocols provide greater energy savings than synchronous MAC protocols, since asynchronous MAC protocols are more efficient than synchronous MAC protocols for event-driven model with light traffic conditions. ELA-MAC outperforms other MAC schemes for energy savings.

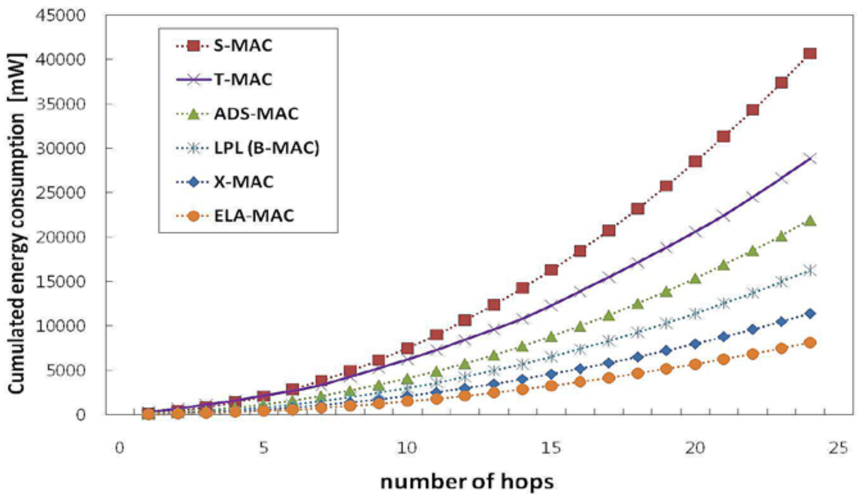

Fig. 5. Cumulated energy consumption in the event-driven model.

5.2 Delivery Delay

Fig. 6 shows the delivery delay in a multi-hop network. These results imply that asynchronous MAC protocols outperform synchronous MAC protocols, because the synchronous MACs achieve synchronization, long contention, and data forwarding with an RTS/CTS/DATA/ACK sequence. Asynchronous MACs, on the other hand, do not include RTS/CTS handshakes and synchronization, but send data immediately. Consequently, the delivery delays of the former are much greater than the latter as the number of hops increases. Our proposed schemes offer more significant improvements. Through the PRTS and closed-loop control with the adaptive timeout (T_A), data travel faster, and more hops and traffic fluctuations are handled in ADS-MAC. Due to the ACK listen periods required in X-MAC, the per-hop delay can be shortened, and the delivery delay is the lowest. However, ELA-MAC also takes advantage of the RSP scheme to alleviate long delays so for similar delivery delay performance to X-MAC. For event-driven model, four source nodes generate data packets simultaneously and send them to the sink node in **Fig. 7**. The delivery delay is shown as more increases in the number of hops in **Fig. 7** because more traffic is generated in the network. These results also show that asynchronous MAC protocols have less delay than synchronous MAC protocols, since there is no synchronization overhead and waste time in asynchronous MAC protocols for event-driven model with light traffic conditions. ELA-MAC has the best performance for data delivery delay compared with other MAC schemes.

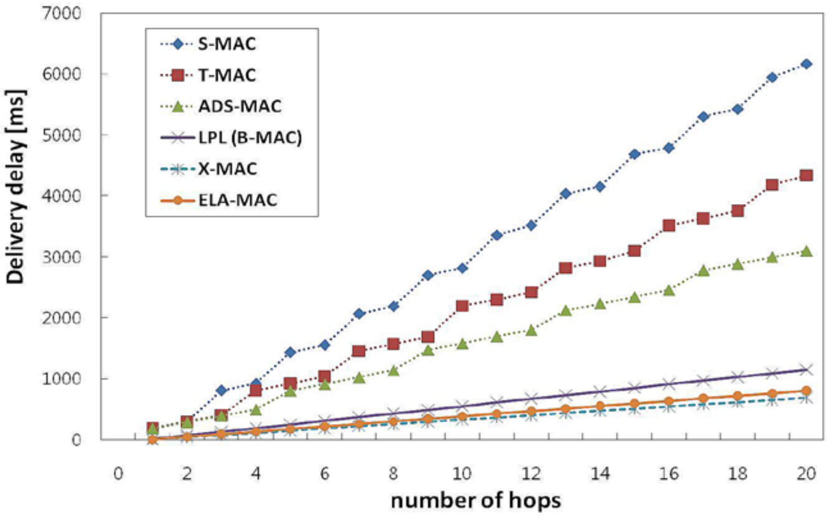

Fig. 6. Delivery delay in a multi-hop network in the periodic model.

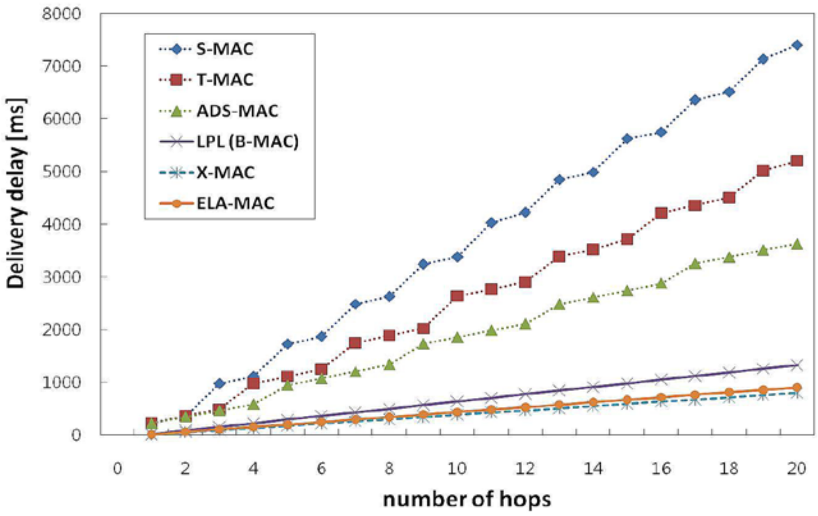

Fig. 7. Delivery delay in a multi-hop network in the event-driven model.

6. Conclusions

MAC protocols of WSNs are designed under the conditions of periodic traffic, light traffic, and delay-sensitive traffic. Usually, synchronous MACs are designed for periodic traffic, and asynchronous MACs are designed for light traffic. Therefore, MAC protocol design in WSNs should be tailored to different characteristics and application needs. In this letter, we propose two energy-efficient MAC protocols for WSNs: ADS-MAC and ELA-MAC. The proposed protocols are performed together in periodic and light traffic conditions. Our simulation results indicate that our algorithms provide significant performance improvements in terms of energy efficiency and delivery delay compared to existing schemes. We are currently implementing our algorithms with tiny-OS based sensor nodes to evaluate their performance in real environments, and intend to present validation of these implementations in future work.

References

[1] LAN MAN Standards Committee of the IEEE computer Society: IEEE Std 802.11-1999, Wireless LAN medium access control (MAC) and physical layer (PHY) specification, *IEEE* 1999.

[2] IEEE 802.15 working group for WPAN, http://ieee802.org/15/index.html

[3] M. Chen, S. Gonzalez, V. Leung, Q. Zhang and M. Li, "A 2G-RFID-based e-healthcare system," *IEEE Wireless Communications*, vol. 17, no. 1, pp. 37-43, Feb. 2010. Article (CrossRef Link).

[4] M. Chen, S. Gonzalez, Q. Zhang and V. C. M. Leung, "Software agent-based intelligence for code-centric RFID systems," *IEEE Intelligent Systems*, vol. 25, no. 2, pp. 12-19, Mar. 2010. Article (CrossRef Link).

[5] M. Chen, S. Gonzalez, A. Vasilakos, H. Cao and V. C. M. Leung, "Body area networks: a survey," *ACM/Springer Mobile Networks and Applications*, pp. 1-23, 2010. Article (CrossRef Link).

[6] N. F. Timmons and W.G. Scanlon, "An adaptive energy efficient MAC protocol for the medical body area networks," in *Proc. of IEEE Wireless VITAE*, pp. 587-593, May 2009. Article (CrossRef Link) .

[7] F. Kojima and H. Harada, "Long-lived smart utility network management using modified IEEE 802.15.4 MAC," in *Proc. of ICC 2010*, pp. 1-5, May 2010. Article (CrossRef Link).

[8] W. Ye, J. Heidemann and D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor network," *IEEE/ACM Transactions on Networking*, vol. 12, no. 3, pp. 493-506, Jun. 2004. Article (CrossRef Link).

[9] T. V. Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proc. of ACM Conference on Embedded Networked Sensor Systems*, pp. 171-180, Nov. 2003. Article (CrossRef Link).

[10] J. H. Kim, H. N. Kim, S. G. Kim, S. J. Choi and J. Y. Lee, "Advanced MAC protocol with energy-efficiency for wireless sensor networks," *Lecture Notes in Computer Science*, *Springer Berlin / Heidelberg*, vol. 3391, pp. 283-292, 2005. Article (CrossRef Link).

[11] J. Polastre, J. Hill and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proc. of ACM Conference on Embedded Networked Sensor System*, pp. 95-107, Nov. 2004. Article (CrossRef Link).

[12] A. El-Hoiydi and J. D. Decotignie, "Low power downlink MAC protocols for infrastructure wireless sensor networks," *ACM Mobile Networks and Applications*, vol. 10, no. 5, pp. 675-690, Oct. 2005**.** Article (CrossRef Link).

[13] M. Buettner, G. V. Yee, E. Anderson and R. Han, "X-MAC:a short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proc. of ACM Sensys'06*, pp. 307-320, Nov. 2006. Article (CrossRef Link).

[14] M. Chen, V.C. M. Leung, S. Mao, Y. Xiao and I. Chlamtac, "Hybrid geographic routing for flexible energy-delay tradeoff," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 9, pp. 4976-4988, Nov. 2009. Article (CrossRef Link).

[15] F. Xia, "QoS challenges and opportunities in wireless sensor/actuator networks," *MDPI Sensors*, vol. 8, no. 2, pp. 1099-1110, Feb. 2008. Article (CrossRef Link).

[16] Y. Jin, Z. Zhang and H. Liu, "Contention window based QoS DCC-MAC for wireless multimedia sensor networks," in *Proc. of IET CCWMC 2009*, pp. 201-204, Dec. 2009. Article (CrossRef Link).

[17] Texas Instruments Inc. cc1000 and cc2420 data sheet, http://www.ti.com

[18] Crossbow Technology mica2 and micaz data sheet, http://www.xbow.com

[19] Network simulator (ns-2 version 2.29). http://www.isi.edu/nsnam/ns

Jaehyun Kim received a B.S. in Electronic Engineering from Kwangwoon University, Korea in 1997 and an M.S. in Electrical and Electronic Engineering from Yonsei University, Korea in 2003. He was employed by SK Teletech as a research engineer from 2003–2004, and he is currently a Ph.D. candidate in Electrical and Electronic Engineering at Yonsei University, Korea. His research interests are MAC/routing protocol design in wireless sensor networks, cross-layer optimization, mobile multicast, and 3GPP LTE technology.

Seoggyu Kim received his B.S., M.S., and Ph.D. in Electrical and Electronic Engineering from Yonsei University, Korea in 1990, 1992, and 1997, respectively. He was employed by SK Telecom as a senior research engineer from 1997–2003, and by the IT research center of Yonsei University as a research professor from 2004-2005. Since 2006, he has been an assistant professor in the Department of Information and Computer Engineering, Andong National University, Korea. His research interests are ubiquitous sensor networks, QoS architecture in wired/wireless networks, and network design for IP-based converged networks.

Jaiyong Lee received a Ph.D. in Computer Engineering from Iowa State University, IA, USA in 1987. He was with employed by the Korean Agency for Defense Development (ADD) as a research engineer from 1977-1982, and by the Computer Science Department of POSTECH as an associate professor from 1987-1994. Since 1994, he has been a professor in the School of Electrical and Electronic Engineering, Yonsei University, Korea. He was the Executive Vice President of the Open Standards and Internet Association (OSIA) in 2005 and an IT professional of the Ministry of Information and Communication Republic of Korea (MIC) from 2003-2007. He is also actively serving in other IT areas, acting as the Executive Director of the Korea Institute of Communication Sciences (KICS), an Advisory Board Member of the Korean Association of RFID/USN, and the editor of the JCN and ETRI journals. Recently, he is doing research in the areas of QoS and mobility management for supporting 4G architecture, multicast protocol design for wireless access networks, and sensor MAC/routing protocol design.