Performance Study for S-MAC in Wireless Sensor Networks

Zhang Teng and Ki-Il Kim, *Member, KIMICS*

***Abstract*— Efficient energy management is a very important issue in wireless sensor network since wireless sensor nodes are usually battery-powered. Recently, S-MAC protocol based on low duty-cycle has been proposed to reduce energy consumption. Even though research effort has been made to evaluate performance of S-MAC by conducting various simulations, however, some important simulation parameters are not well evaluated yet. In this paper, we identify the performance of S-MAC under different amount of streams and different patterns such as data rate and traffic. Through analyzing the simulation results, we discover weakness of S-MAC as well as analyze impact of amount of streams and packet pattern.**

***Index Terms*— Sensor networks, S-MAC.**

I. INTRODUCTION

WSN(Wireless sensor network) [1] has become more and more popular recently because it made various network applications possible. A sensor node is a small device which is able to carry out environmental monitoring, mobile target tracking, smart space, and ubiquitous computing. However, the most noteworthy problem of the WSN is that each sensor node is operated by a limitative power, the battery. It is very hard to charge and replace battery. When the battery power is completely exhausted, the node will be dead. So, one of the most important challenges is how to reduce unnecessary energy consumption in order to prolong the lifetime of sensor network as well as prevent unexpected holes.

In order to suit to energy constrained environment, the design model to reduce energy consumption is often selected in MAC (Medium Access Control) protocol. So, typical protocols like IEEE 802.11 are not recommended because their primary goal is to maximize throughput. In order to save energy, it is the first step to identify where energy is consumed as well as wasted. First, collision can lead to frequent packet drop and corruption. Thus, discarded packet should be retransmitted but it contributes to increase energy consumption and causes additional delay. The second way is overhearing, which means a node picks up packets that are destined to other nodes. Idle listening to receive possible traffic which is not sent is another source to waste energy. In addition, sending and receiving control messages also consumes energy. IEEE 802.11 is a good MAC protocol for packet collision. CSMA/CA is used in IEEE 802.11. With this procedure, nodes can first listen to the channel for a predetermined amount of time and then send the RTS (request-to-send)/CTS (clear-to-send) control message to decide whether to send a data packet for avoiding collision. But in overhearing and idle listening condition, it has a bad representation.

To avoid problems mentioned above, Ye et al. proposed S (Sensor) – MAC [2], that is specifically designed to reduce energy waste on IEEE 802.11 when it runs on sensor node. S-MAC tries to reduce energy waste by avoiding all the problems mentioned above. The most outstanding technique in S-MAC is to use periodic sleep and wake up to reduce idle listening. In the sleep state, the sensor turns off the radio to save energy.

Sometimes, latency is also very important, but in S-MAC, it is a trade-off between energy consumption and latency. Due to the periodic sleep and wake up, the latency is longer than full active protocol like IEEE 802.11. To resolve this problem, Ye et al. improve a new technique which is called S-MAC with adaptive listen for the high latency [3]. Relative to S-MAC, it reduces the latency effectively without scarification of energy. In addition to [3], some mechanisms [4-10] have been proposed to improve the performance of S-MAC. Their schemes usually conduct simulation to identify the performance of S-MAC, though some scenarios and parameters are missing in the simulations. In order to compare them completely, we introduce different amount of source nodes, different amount of streams from one source and different packet pattern environments in this paper. According to that the new applications such as multimedia will be soon included in sensor networks, current trends and facts imply that the performance study is necessary to adapt these environments and to identify the performance of S-MAC.

The rest of this paper is organized as follows. Section II describes S-MAC and S-MAC with adaptive listen briefly. Section III shows the performance evaluation of S-MAC with adaptive listening different circumstances by simulations. Section IV concludes this paper and also includes the further work in this research fields.

Manuscript received January 13, 2010; revised January 20, 2010; accepted February 2, 2010.

The authors are with the Department of Informatics, Research Institute of Computer and Information Communication, Gyeongsang National University, Jinju, 660-701, Korea (Tel: +82-55-751-6282, Fax: +82-55-751-6280, Email: kikim@gnu.ac.kr). Ki-Il Kim is corresponding author.

II. OVERVIEW OF S-MAC

An important design consideration in S-MAC is to reduce energy consumption while providing collision avoidance. In addition, it can also decrease idle time and reduce impact by overhearing. In sensor network the key issue is lifetime of sensor. In S-MAC, node uses periodic sleep/wake up to prolong the lifetime of sensor network.

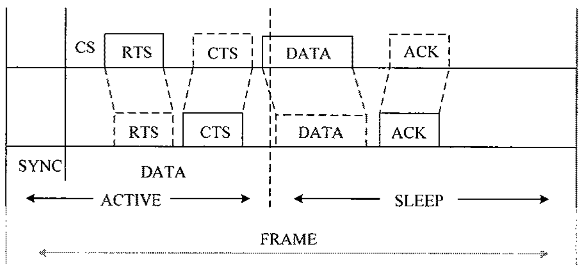


Fig. 1. S-MAC frame format, CS stands for carrier sense.

Fig. 1 shows the basic S-MAC operations. Each node becomes active for some duration to detect if there is any other node wishing to exchange messages with it. If there is one, the detecting node will make a transmission with that one otherwise, it will go to sleep and wake up in the next frame. During sleep, the node turns off its radio for saving energy and sets a schedule to wake up itself later. In low duty-cycle, it can save a lot of energy but lead to a long latency.

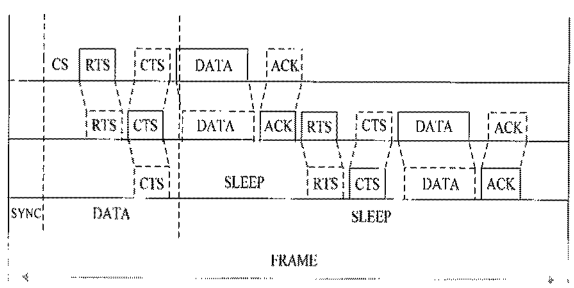


Fig. 2. S-MAC with adaptive listen frame format, CS stands for carrier sense.

To make neighboring nodes wake up at the same time for reducing latency and controlling overhead, S-MAC uses SYNC message to synchronize nodes in the same cluster to have the same sleep/wake up schedule. The cluster forms virtual cluster rather than real communication cluster like Bluetooth. At the first time of S-MAC frame, nodes utilize SYNC message to consist virtual cluster and maintain synchronization, it is called a synchronization period.

To avoid collision, S-MAC takes similar procedures introduced in IEEE 802.11, such as physical carrier sense, network allocation vector (NAV), and the RTS/CTS exchange for the hidden terminal problem as shown in Fig. 1. Before initiating a transmission, all senders perform carrier sense in order to check whether the medium is free or not. If the medium is free, a node can consequently utilize RTS/CTS/DATA/ACK to exchange packet as shown in Fig. 1. When the transmission is finished, the node goes to sleep. If no one detects event in active state, nodes should go to sleep and wake up again in the next frame according to predetermined schedule.

The follow contents introduce that how to choose schedule and establish schedule table.

- First, a node listens to the channel for a fixed amount of time as much as at least the synchronization period. If the node does not hear any schedule from other nodes, it will choose a sleep/wake up schedule by itself immediately and start to follow it. Then, the node tries to announce the schedule by broadcasting a SYNC packet to neighboring nodes. We call such a node a synchronizer since it chooses its schedule independently and other nodes will follow the schedule with it.
- If a node receives a SYNC message from a neighbor before choosing or announcing its own schedule, it will follow that schedule by setting its own one to be the same. Then the node will announce this schedule at its next scheduled active time. We call such a node a follower.
- In the protocol there are two cases if a node receives a different schedule after it chooses and announces its own schedule. If the node has no other neighbor, it will discard its current schedule and follow the new one. If the node has already followed a schedule with one or more neighbors, it will adopt both schedules by waking up at the active intervals of the two schedules.

In S-MAC protocol, all of the packets include the time duration together with current transmission. When a node is sending a control packet, its neighboring node which overhearing the packet will go to sleep for avoiding collision and saving energy. Therefore, all neighboring nodes around both the sender and receiver will have to go to sleep after they overhear the control message (RTS or CTS) until the current transmission is finished.

By Fig. 1, we can found that one node can transmit only one packet with another node at the low duty-cycle. This will lead to a rapid increase of delay. S-MAC with adaptive listen is proposed to improve it. Let's look at the diagram in Fig. 2. When node A wants to send data to node B, they will follow the sequence of RTS/CTS/DATA/ACK to transmit the date. When node B replies CTS to node A, if node C overhears the CTS, it will realize when the transmission is over and then go to sleep. When packet exchange is finished between node A and node B, node C will wake up to listen to the control message (RTS) sent by node B. If node C receives the RTS, it will reply one CTS immediately and the data packet sent from node B to node C will be contained in the same frame. Thus, in one frame, a packet can be sent via two hops. This technique will reduce about half the latency without regard to no more energy consumption.

S-MAC also has an approach that can fragment a large message into some small fragments and transmit the fragments in a burst while using only one RTS message and only one CTS message. When a data fragment is sent out, the sender will wait for the ACK packet from the receiver. If the sender fails to receive the ACK packet, it will extend the transmission time and retransmit the fragment immediately.

III. PERFORMANCE EVALUATION

To evaluate S-MAC with adaptive listen design, we evaluate it by version 2.29 of the ns-2 simulator. We compare the fundamental trade-offs between energy and latency with S-MAC with adaptive listen in different circumstances.

A. Simulation Environments

Table 1 shows the key parameters we used in our simulation. These values are the default settings with the ns-2.29 package and used in our simulation. Fig. 3 shows the original topology evaluated in our simulation, a chain scenario. It is a four-hop network with one source and one sink where node 0 can send packets to node 4 through node 1, node 2 and node 3. The distance between every two neighboring nodes is 250m in these scenarios because the transmission radius of the node is set as 250m. According to this scenario, we can ensure that nodes will not impact each other.

Sometimes the source nodes will send packets of large size which have to be divided into some fragments. These fragments are always transmitted in a burst time since S-MAC with adaptive listen uses the packet passing scheme. If a large packet is sent out, it will be divided into some fragments and sent in a burst, i.e., those fragments from one large packet are just used on one RTS and one CTS.

TABLE I
SIMULATION PARAMETERS

Bandwidth	20 Kbps	Sleep Power	0.01 W
Tx Range	250 m	Idle Power	0.5 W
Tx Power	1.0 W	Contention window (CW)	64 ms
Rx Power	1.0 W	Carrier Sensing Range	550 m

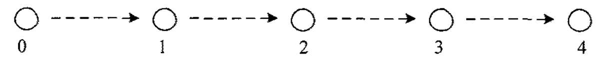


Fig. 3. Topology 1: original chain scenario.

In our simulation, the sizes of control messages (RTS and CTS) are 10 byte, the SYNC message is 9 byte in size. The source address and destination address are included in control messages, and the SYNC messages include the next sleep time from now when the SYNC message is sent from source node, etc.

In next subsection, we introduce our evaluation about the latency and energy consumption of S-MAC with adaptive listen in different amount of source nodes, different amount of CBR (Constants Bit Rate) streams, different packet patterns and S-MAC at different duty-cycle conditions. We use three kinds of scenarios and compare the performances by changing the parameters mentioned above. In the S-MAC with adaptive listen at periodic sleep, each node is set to run in 20% duty-cycle. In each test, the size of the packets from each source node is 512 bytes long. There is no fragmentation on all messages.

B. Simulation Results

The first simulation is to present the delay of each packet and average energy consumption of each node in different source nodes scenario. Fig. 4 shows the topology to us. In these scenarios, the amount of source nodes increase gradually on the original chain topology, in addition the distance of each hop is 250m to ensure not to impact each other. But maybe there are some impacts between source nodes.

S-MAC with adaptive listen is suit to a low traffic for saving energy. So in every simulation of this part, each source node generates messages periodically and the message inter-arrival period is 20s. We choose this interval of message inter-arrival period because it can lead to a light traffic.

Fig. 5 shows the mean energy consumption of each node in different amount of source nodes scenarios. In this case, IEEE 802.11 MAC consumes more energy than S-MAC with adaptive listen, especially there is only one source node. Because when the number of source nodes is one, the traffic is very light, the idle listening often happens. In S-MAC, if it does not receive task for transmitting packet by control message(RTS), the node will go to sleep followed its time schedule. But in IEEE 802.11 there is no method for avoiding idle listening, nodes will spend a lot of energy in the idle listening.

When the amount of source nodes is increasing, the traffic will be also increasing. In S-MAC nodes have to spend more time for message transmission, so the energy consumption is increasing gradually. However, in IEEE 802.11, although the nodes have to send more packets when the traffic is increasing, since the message transmission is very fast, the difference between those two network traffic states is not large in IEEE 802.11. It consumes energy mainly by wasting a lot of energies in idle listening. S-MAC can use the periodic sleep/wakeup on radios to avoid this problem.

Fig. 6 shows the average delay of each packet in different amount of source nodes scenarios. From the this figure, the delay of S-MAC with adaptive listen is much higher than IEEE 802.11. That is because S-MAC uses the periodic sleep/wake up schedule and in every frame S-MAC with adaptive listen sends one or two packets, other packets must be stored in the buffer first and then wait for the next frame. But in IEEE 802.11, nodes can transmit messages to another node immediately due to the nodes always stay in active state, they do not have to consider the state of neighbouring node, but just need to be careful to avoid message collision with other packets. Here is a trade-off between delay and energy consumptions in S-MAC since S-MAC will turn off radios for saving energy, and lead to a high latency.

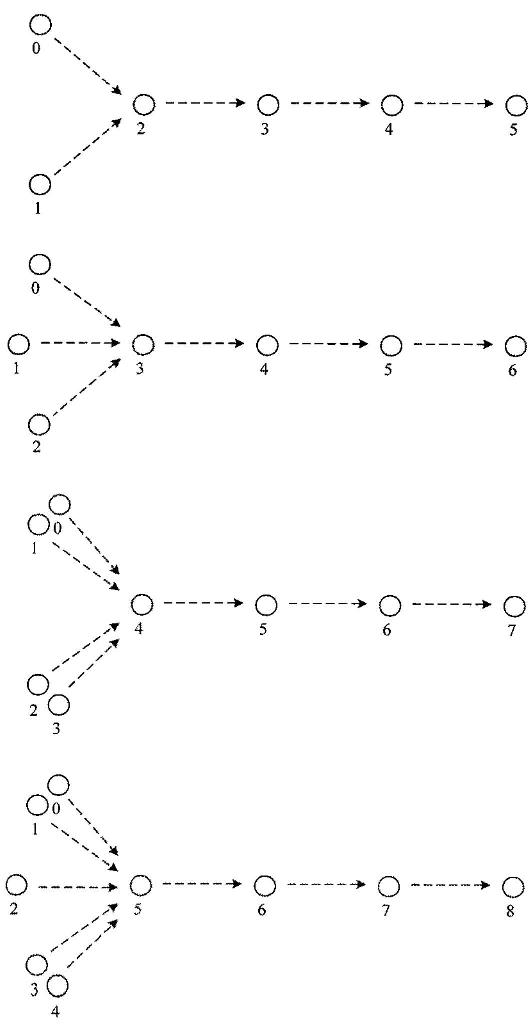


Fig. 4. Topology 2: increasing the number of source nodes gradually.

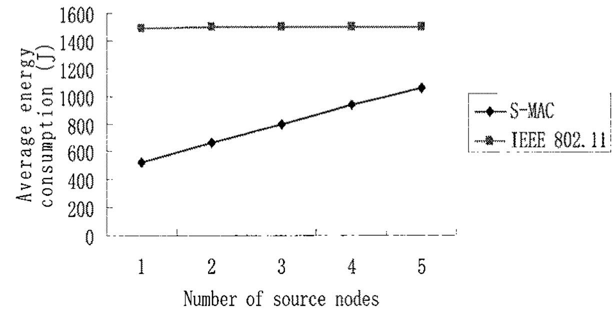


Fig. 5. Average energy consumption of each node in the different amount of source nodes scenario.

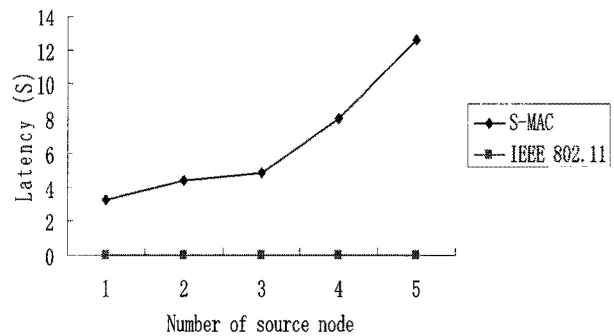


Fig. 6. Average latency of each packet in different amount of source nodes scenarios.

When the amount of source nodes is increasing, the traffic will be also increasing. In S-MAC nodes have to spend more time for message transmission, so the energy consumption is increasing gradually. However, in IEEE 802.11, although the nodes have to send more packets when the traffic is increasing, since the message transmission is very fast, the difference between those two network traffic states is not large in IEEE 802.11. It consumes energy mainly by wasting a lot of energies in idle listening. S-MAC can use the periodic sleep/wakeup on radios to avoid this problem.

Fig. 6 shows the average delay of each packet in different amount of source nodes scenarios. From the this figure, the delay of S-MAC with adaptive listen is much higher than IEEE 802.11. That is because S-MAC uses the periodic sleep/wake up schedule and in every frame S-MAC with adaptive listen sends one or two packets, other packets must be stored in the buffer first and then wait for the next frame. But in IEEE 802.11, nodes can transmit messages to another node immediately due to the nodes always stay in active state, they do not have to consider the state of neighbouring node, but just need to be careful to avoid message collision with other packets. Here is a trade-off between delay and energy consumptions in

S-MAC since S-MAC will turn off radios for saving energy, and lead to a high latency.

From the Fig. 6, the traffic is light when the number of source nodes is less than 3. In the case of one source node for each other, the delay increases slightly. That is because when the amount of source nodes is small, the impact of overhearing between source nodes is slight. When the number of source node is more than 3, the traffic load becomes heavy. In this case, when one source node is added, the delay increases greatly. That is because when the amount of source nodes is big, the impact of overhearing between source nodes is great. In one frame, although each source node generates packets periodically, only one source node can win the contention and send the message to the next hop. Other source nodes overhearing its control message should go to sleep and waiting for the next frame for transmitting messages. For example, in this five source nodes scenario, if one source node loses the contention in every frame, it will wait four time intervals of frame to transmit packets, therefore, the delay increases highly.

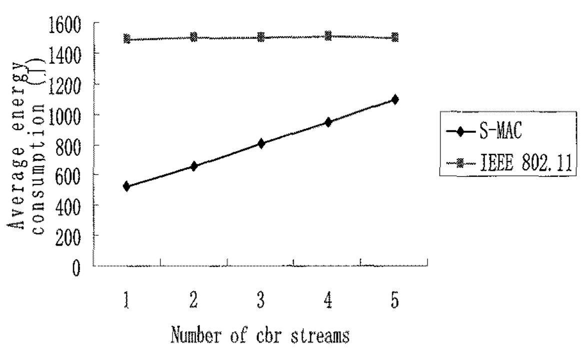


Fig. 7. Average energy consumption of each node by different amount of CBR streams in the original chain scenario.

Fig. 7 shows the average energy consumption of each node by different amount of CBR streams in the original chain scenario. Although in the last simulation the network also has different amount of streams from different amount of source nodes, this one is totally different. In this case there is only one source node and different amount of CBR streams is generated from this source node. From Fig. 7, the data of energy consumption is similar to Fig. 5. Because the two experiments have the same message inter-arrival period, same simulation time and same duty-cycle, both the number of sent packets and idle time between these two experiments are similar, that leads to a similarity of energy consumption.

Fig. 8 shows the average latency by each packet in different amount of CBR streams in the original chain scenario. In the same way, the latency of S-MAC with adaptive listen is higher than IEEE 802.11, and the reason is almost the same. The delay changes while we are varying the number of CBR stream in S-MAC. Comparing the two figures of Fig. 6 and Fig. 8, although all of them have the same amount of stream, the latency is not the same. In this case, the average delay is lower than experiment mentioned above, especially when the number of stream is more than 3. That is because in the last simulation, the impact of overhearing between source nodes is very high, and when the traffic becomes heavy, the latency is also affected by buffer. That means a latter packet must be stored in buffer until the former one has been sent out from node. Due to the periodic sleep/wake up, the nodes can change control messages for packet transmission by S-MAC just in the wake up state, this character adds the impact to buffer. As a trade-off, it may cause a high latency in return for gaining more energy conservation. In this case the source node is just affected by buffer but not by overhearing, so the delay is lower than the one above.

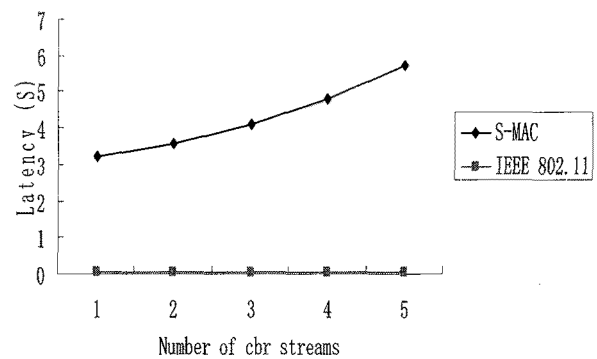


Fig. 8. Average latency of each packet by different amount of CBR streams in the original chain scenario.

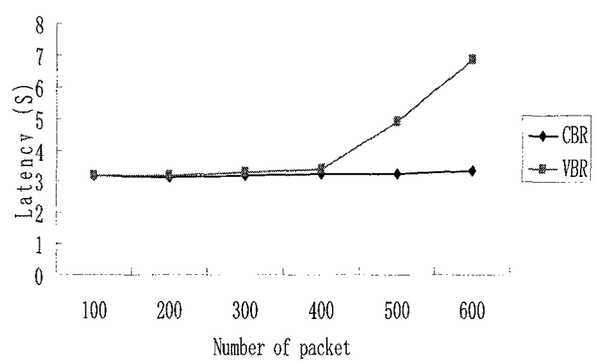


Fig. 9. Average latency of each message by different packet pattern in the original chain scenario.

Fig. 8 shows the average latency by each packet in different amount of CBR streams in the original chain scenario. In the same way, the latency of S-MAC with adaptive listen is higher than IEEE 802.11, and the reason is almost the same. The delay changes while we are varying the number of CBR stream in S-MAC. Comparing the two figures of Fig. 6 and Fig. 8, although all of them have the same amount of stream, the latency is not the same. In this case, the average delay is lower than experiment mentioned above, especially when the number of stream is more than 3. That is because in the last simulation, the impact of overhearing between source nodes is very high, and when the traffic becomes heavy, the latency is also affected by buffer. That means a latter packet must be stored in buffer until the former one has been sent out from node. Due to the periodic sleep/wake up, the nodes can change control messages for packet transmission by S-MAC just in the wake up state, this character adds the impact to buffer. As a trade-off, it may cause a high latency in return for gaining more energy conservation. In this case the source node is just affected by buffer but not by overhearing, so the delay is lower than the one above.

Fig. 9 shows the average latency of each message by different packet patterns in the original chain scenario. In general we use CBR packet for our simulation, but sometimes, adopting other packet pattern is also required. In this case we check the performance of VBR (Variable Bit Rate) stream and compare with CBR by S-MAC with adaptive listen. Different from CBR, VBR varies amount of messages which are sent out per fixed time interval, instead of transmitting packets in a steady rate from source node to sink node.

In Fig. 9, the x-axis denotes the amount of sent messages by two packet patterns and the y-axis denotes the average latency. In this case, the source node generates the same amount of messages at each simulation by the two packet patterns and in each time the simulation possesses the same run time, in addition, the size of each packet is as same as 512 bytes. As shown in the figure, when the number of packets sent out is less than 400, latencies of the two packet patterns are similar, since the traffic is light towards the two packet patterns in S-MAC with adaptive listen in the range of amount of sent packets. When nodes receive some messages, they are able to forward the packets in the active state immediately, instead of being stored in the buffer. When the number of packets sent out is more than 400, as shown in the figure, the latency of VBR stream is increasing but the data of CBR stream is not changed. Since the CBR stream has the constant bit rate, source node always keeps a fixed message inter-arrival period over time, and the traffic is still light for S-MAC with adaptive listen by CBR stream in the range of amount of sent packets, so the delay of CBR stream is not changed. About the VBR stream, due to the variable bit rate, it's very hard to maintain a fixed bit rate for packet transmission. Sometimes the source node would adopt a high bit rate for message transmission, or else there is not any packet generated in a long time. As shown in the Fig. 9, when the number of packets sent out is more than 400, due to the high bit rate, the latency of VBR stream is rose by S-MAC with adaptive listen.

Fig. 10 shows the average latency of each message on each hop by different duty-cycles in the 10-hop chain scenario. In this case we use 10-hop topology and the message inter-arrival period is 2.5s. In all three S-MAC duty-cycles, the latency increases linearly with the number of hops. However, S-MAC at 30% duty-cycle has much higher latency than the other ones. On the second hop, the delay of 30% duty-cycle is a little higher than the others, but after the second hop, the delay increases rapidly. The reason is that the traffic is high for S-MAC at 30% duty-cycle with adaptive listen, after the second hop, due to the sleep/wakeup schedule, a large amount of packets can't be transmitted immediately. They are stored in the buffer and wait to be transmitted sequentially. However, since S-MAC at 50% and 70% duty-cycle with adaptive listen possesses the long enough time for message transmission, the packets will not have to wait for a long time to be sent out. Therefore, S-MAC at 30% duty-cycle with adaptive listen has much more delay than the others.

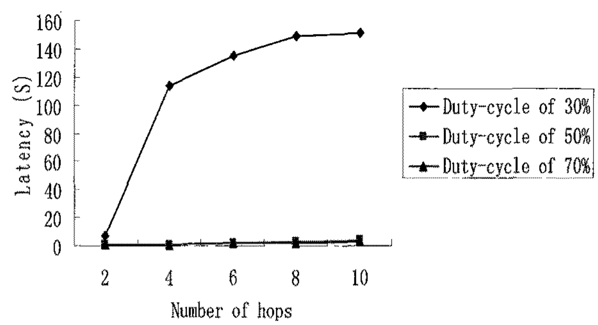


Fig. 10. Average latency of each message on each hop by different duty-cycle in the 10-hop chain scenario.

IV. CONCLUSION AND FURTHER WORKS

This paper presents a performance evaluation for energy-efficient Medium Access Control (MAC) protocol with a low duty-cycle called S-MAC, and the trade-off between energy consumption and latency. By making some simulations by S-MAC in the different circumstances and analyzing the simulation results, several impacts for S-MAC are discovered.

In S-MAC with adaptive listen, although the latency is decreased by utilizing overhearing, there is also long delay. We will continue searching the trade-off between energy and latency. The research to adapt multimedia data over S-MAC will be included in further works.

ACKNOWLEDGMENT

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-2010-C1090-1031-0007).

REFERENCES

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, "A survey on sensor networks," IEEE Communications Magazine, Vol. 40, No.8, 2002, pp. 102-114.

[2] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient mac protocol for wireless sensor networks," in Proc. of IEEE INFOCOM, Jun. 2002, pp. 1567-1576.

[3] W. Ye, J. Heidemann, D. Estrin, "Medium access control with coordinated adaptive sleeping for wireless sensor networks," IEEE/ACM Transactions on Networking, Vol. 12 No. 3, 2004, pp. 493-506.

[4] T.-H. Hsu, J.-S. Wu, "An application-specific duty cycle adjustment MAC protocol for energy conserving over wireless sensor networks," Computer Communications, Vol. 31, 2008, pp. 4081-4088.

[5] P. Lin, C. Qiao, X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in Proc. of the IEEE Wireless Communications and Networking Conference (WCNC), 2004, pp. 1534-1539.

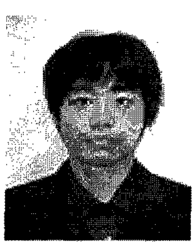
[6] R. Yadav , S. Varma, N. Malaviya "Optimized medium access control for wireless sensor network," International Journal of Computer Science and Network Security (IJCSNS), Vol. 8, No. 2, 2008, pp. 334-338.

[7] T. van Dam, K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in Proc. of the International Conference on Embedded Networked Sensor Systems (SenSys), 2003, pp. 171-180.

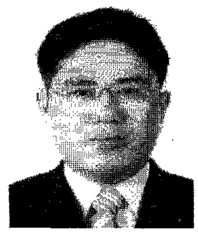
[8] S.-H. Yang, H.-W. Tseng, E.-K. Wu, G.-H. Chen, "Utilization based duty cycle tuning MAC protocol for wireless sensor networks," in Proc. of IEEE GLOBECOM, 2005, pp. 3258-3262.

[9] S. Du, A.K. Saha, D.B. Johnson, "RMAC: a routing-enhanced duty-cycle MAC protocol for wireless sensor networks," in Proc. of IEEE INFOCOM, 2007, pp. 1478-1486.

[10] J. Kim, K. H. Park, "An energy-efficient, transport-controlled MAC protocol for wireless sensor networks," Computer Networks, 2009.



Zhang Teng is currently working toward M.S. degree in the Gyeongsang National University, Jinju, Korea. His research interests include sensor networks, and MAC protocol.



Ki-Il Kim received the M.S. and Ph.D. degrees in computer science from the ChungNam National University, Daejeon, Korea, in 2002 and 2005, respectively. He is currently with the Department of Informatics, Gyeongsang National University as a faculty member. His research interests include routing for MANET, QoS in wireless network, multicast, and sensor networks.