2011 Third International Conference on Communications and Mobile Computing

LC-MAC : An Efficient MAC Protocol for the Long-Chain Wireless Sensor Networks

Chen Fang
frank_fc@seu.edu.cn

Hao Liu
nicky_lh@seu.edu.cn

LiLi Qian
lillian_qll@seu.edu.cn

National ASIC System
Engineering Research Center
Southeast University
Nanjing 210096, China

***Abstract* — In this paper, we present a new duty-cycle media access control(MAC) protocol, called Long-chain MAC protocol (LC-MAC), that exploits a mechanism for relay nodes booking in advance and transmitting in a burst manner in order to reduce the end-to-end delivery delay in a long-chain sensor network scenario without sacrificing energy efficiency. In LC-MAC, a super SYNC frame can travel across multiple hops and schedule the upcoming data packet delivery along the long-chain route. Each relay node sleeps and intelligently wakes up at a scheduled time, so that the upstream node can send the data packet to it and then it can immediately forwards the data packet to the downstream node. Our simulation results in ns-2 show that LC-MAC achieves significant improvement in end-to-end delivery delay over S-MAC without sacrificing energy efficiency or network throughput.**

Keywords-LC-MAC; long-chain; WSN; MAC; multi-hop

I. INTRODUCTION

Inexpensive integrated system-on-a-chip devices comprising a radio transceiver and a microcontroller have been provided by many IC manufacture companies [1][2]. It has made various wireless sensor network applications possible. In many wireless sensor network applications the traffic pattern consists of data collected from many source nodes to a sink through a unidirectional backbone network. Especially, the backbone composed of ten or more relay nodes seems like a long-chain in some of the applications. These applications include monitor systems for electric cable, boats in a watercourse, traffic on a road and so on. Similar to other applications, nodes except the sink have limited power resources, poor capability and a little amount of storage.

In a long-chain network, events are detected by relay nodes first. After that, relay nodes are responsible for relaying these events as data packages to the sink. Assuming events happening follow an average probability distribution along the long-chain, the data packages will be relayed hop-by-hop in a chain-to-one traffic pattern toward a sink. It will lead to increased transit traffic intensity and latency as relay nodes move closer towards the sink, resulting in significant packet collision, congestion, and loss; at best this leads to limited application fidelity measured at the sink, and at worst the congestion collapse of the sensor network[3].

It is well known, Ye et al. proposed S-MAC[4] which is specifically designed to reduce energy wastage on IEEE 802.11 based sensor nodes. In order to save more energy and reduce end-to-end delivery delay, addressing on the fixed duty cycle of S-MAC, U-MAC[5], T-MAC[6] and DS-MAC[7] are proposed to provide various duty cycle scheme to assign different duty cycles for different tasked nodes and different traffic load. U-MAC balances the tradeoff by utilization based tuning of duty cycle and selective sleeping after transmission. Similar to U-MAC, DS-MAC tunes duty cycle by doubling or halving according to latency and the emptiness of packet queue. T-MAC introduces an adaptive duty cycle in a novel way: by dynamically ending the active part of it. This reduces the amount of energy wasted on idle listening, in which nodes wait for potentially incoming messages, while still maintaining a reasonable throughput. But the MAC protocols utilizing active/sleep duty cycles mentioned suffer from a data forwarding interruption problem, where nodes on a multi-hop path to the sink are not all notified of data delivery in progress, resulting in significant sleep delay. DMAC[8] is designed to solve the interruption problem by giving the active/sleep schedule of a node an offset depends upon its depth on the tree topology. This scheme allows continuous packet forwarding because all nodes on the multi-hop path can be notified of the data delivery in progress. But DMAC is designed for unidirectional tree topology and not special for the long-chain topology, so it needs time to notice the nodes on the tree for more data transmission. It is inefficient for long-chain topology and will take more time. Other extensive MAC solutions for achieving energy-efficient based on TDMA(Time Division Multiple Address) technique[9]-[11] and multi-channel [12]-[14] are also not suitable for this situation. All the MAC protocols above have done well in the wireless sensor network for reducing power consumption and end-to-end delivery delay. But for the long-chain topology, these protocols are not suitable and we have to design a new MAC protocol for it.

In order to reduce the latency and prolong the lifetime of the nodes in a long-chain topology, this paper proposes LC-MAC, a low latency and energy efficient MAC protocol. The contribution of this paper is the proposal of LC-MAC, a novel protocol for the long-chain topology wireless network. LC-MAC is based on Location

978-0-7695-4357-4/11 $26.00 © 2011 IEEE
DOI 10.1109/CMC.2011.65

Detection technique to help relay nodes find its location. Super SYNC Message Passing technique is applied to relay the super SYNC message in a short time. Burst Transmission technique is the most important point to realize data packages transmitting in a quick way.

Rest of the paper is organized as follows: Section II proposes the problem we met and section III is protocol design. Section IV is performance evaluation. Section V is conclusion.

II. PROBLEM STATEMENT

Unlike other researches, we do not consider a universal MAC protocol. Our research is focus on a real application called electric cable monitor system. In this real scene, wireless relay nodes (noted as R_n) integrating temperature and vibrating sensors are attached to cable connectors deployed in a line along the electric cable as illustrated in Fig.1.a. The aim of this system is to relay the monitoring data including temperature and vibrating from every relay node to a server so that the software can analyze the information and give an alarm before damage happens. All the relay nodes are powered by battery. Our goal for this application is to reduce end-to-end delivery delay the more the better and save nodes' energy as much as possible.

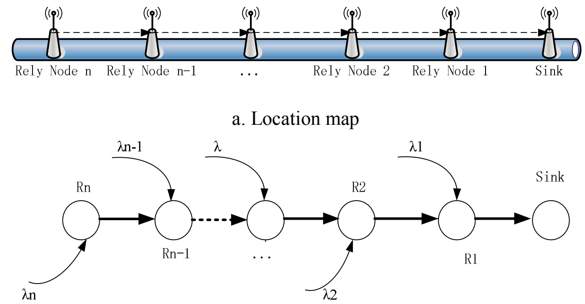
This scene can be abstracted as a long-chain transmission problem, as illustrated in Fig.1.b. Every relay node except the sink has to detect the events from the sensors and compose data packages that are going to be sent to the sink. Besides, a relay node except R_n has to relay the data packages from other relay nodes. Assuming R_1 gets a traffic load of λ_1 packages/s from sensors and R_2 gets λ_2 packages/s. The traffic load prepared for node i is

$$TL_i = \sum_{i}^{n} \lambda_i \tag{1}$$

TL_i is denoted as the traffic load of node i. The frames required for transmission by the MAC protocols utilizing active/sleep working manner is

$$N_{total} = \sum_{i=1}^{n} (N_{ip} \times i) \tag{2}$$

N_{total} is denoted as the total frame need for transmission. N_{ip} is denoted as the number of data packages collected by node i in T. We can find out that the minor i is, the heavier traffic load is, the more frames are. That is to say data packages gotten by relay nodes increase traffic load intensity and latency at the relay nodes closer toward the sink. Typically within a small number of hops, it will lead to a significant packet collision, congestion, and loss; and even lead to congestion collapse of the whole network as mentioned in section I.



a. Location map

b. Long-chain topological graph

Figure 1. Topological graph.

III. PROTOCOL DESIGN

A. Location Detection

Assuming all the relay nodes are equally spaced in a line and every relay node can adjust its radio emission power to make ensure its neighboring relay nodes can hear it. At the beginning of initialization, relay nodes detect neighboring relay nodes. The relay node which only has one neighboring relay node will set itself as an end point of the long-chain noted as R_n. And then, R_n will send a Location Detect Package(LDP) including its address to its neighboring relay node. The neighboring relay node gets the LDP will add its address into this package and send it to another neighboring relay node. The LDP will be relayed one by one and finally reach the sink. The sink sends the LDP with address table back along the route as it came. As a result, relay nodes can get their location information. The end point of the long-chain is noted as R_n and the nearest relay node to the sink is noted as R_1. Other relay nodes are noted from R_{n-1} to R_2 as illustrated in Fig. 2.

B. Super SYNC Message Passing

After step A, every relay node gets location information of the long-chain clearly and will create a Staggered Wakeup Schedule (SWS) for relaying Super SYNC Message (SSYNC). There is a relay action every m(m⩾1) frames and the relay action happens in the listen duration. When the relay action happens, every relay node will follow a Staggered Wakeup Schedule to relay SSYNC package as illustrated in Fig. 3. The Staggered Wakeup Schedule for a relay node is calculated according to the node's location. For a relay action, it doesn't include RTS(Request To Send) or CTS (Clear To Send), because every relay node follows the Staggered Wakeup Schedule to transmit SSYNC, so collision can be avoided. The total duration for the relay action is T_{relay} as expressed in equation (3).

$$T_{relay} = Ts \times n \tag{3}$$

T_s is denoted as the duration for SSYNC transmitting once.

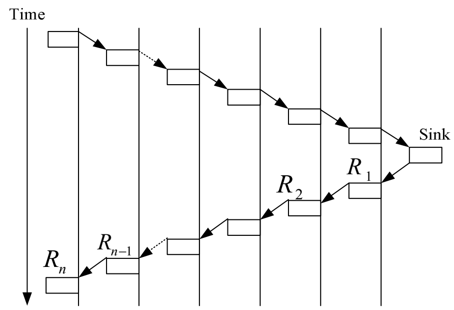


Figure 2. Distance detection.

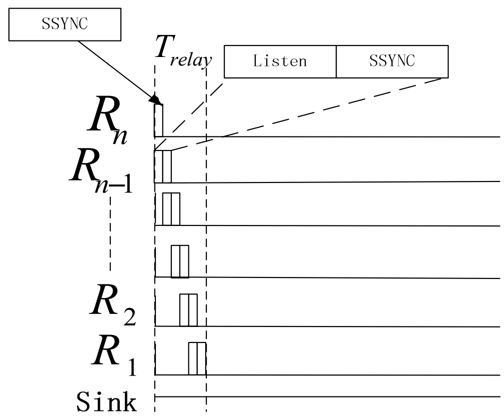


Figure 3. SSYNC relay action.

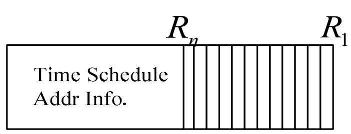


Figure 4. SSYNC framework.

The framework of a SSYNC is as illustrated in Fig. 4. The SSYNC is composed of a transmition information part and a registration form part. The transmition information part includes a sleep schedule and address information. The registration form part is divided into n fractions for n relay nodes. Every fraction has a space (p bits (p⩾1)) to register the number of packages which are going to send. Once the endpoint confirmed, the length of SSYNC is fixed. Any relay node getting SSYNC will update it. The SSYNC package is created by R_n and sent to R_{n-1}. The package will be relayed one by one until it reaches the sink finally. Every relay node registers the number of packages which are going to send. The range of the number is 0 to (2^p-1). The duration of the SSYNC is

$$T_s = T_{ss} + T_r \times n \tag{4}$$

The length of the fixed transmition information is denoted by T_{ss}. The fixed length of a fraction is denoted by T_r. The length of registration form is $T_r \times$n. Equation (4) shows that, T_s is determined by the number of relay node in the long-chain. For an example, registration form takes 5 Bytes if there are 20 relay nodes in the long-chain and p=2. Its length increases a little compared to a single time schedule package used by S-MAC.

C. Burst Transmission

After step B, Every relay node gets the information about the number of data packages belonging to different relay nodes which are going to be relayed. Every relay node will calculate (and find) the time point to wake up and relay the data packages. It is also a staggered wakeup. The burst transmission will take T_{burst} expressed as equation (5). For easier understanding, we omit Short InterFrame Spacing (SIFS) duration for application data processing here.

$$\begin{aligned} T_{burst} &= (T_{data}+T_{ack})\times N_n \times n + (T_{data}+T_{ack})\times N_{n-1}\times(n-1) \\ &+ (T_{data}+T_{ack})\times N_{n-2}\times(n-2) ... + (T_{data}+T_{ack})\times N_1 \times 1 \end{aligned} \tag{5}$$

The number of data packages need to be sent by R_n is denoted by N_n and for R_{n-1} is N_{n-1}, and so on. The duration of sending a data package is denoted by T_{data} (assuming the length of all the transmission durations are equal) and the duration of receiving a ACK(Acknowledgement) package is denoted by T_{ack}. For an example, R_n has two data packages need to be sent to the sink and R_{n-1} has one, n is 10. As illustrated in Fig. 5. We can get equation (6).

$$\begin{aligned} T_{burst} &= (T_{data}+T_{ack})\times 2 \times 10 + (T_{data}+T_{ack})\times 1 \times (10-1) \\ &= (T_{data}+T_{ack})\times 29 \end{aligned} \tag{6}$$

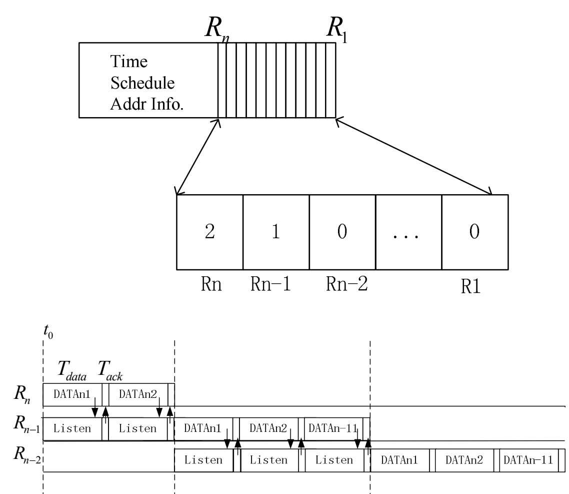


Figure 5. Staggered wakeup.

IV. PERFORMANCE EVALUATION

To evaluate our LC-MAC design, we evaluated it using version 2.29 of the ns-2 simulator. We simulate the Two RayGround radio propagation model and a single omni-directional antenna at each sensor node and a NOAH(No Ad Hoc Routing) routing protocol. Table I shows the key parameters we used in our simulations. Traffic loads are generated by CBR(Constant Bit Rate) flows, and all data packets are 50 bytes in size. Intermediate relaying nodes do not aggregate or compress data. We also assume that the application data processing at any node can be finished within a SIFS period. Thus, data processing introduces no extra latency. The transmission latencies for different types of packets are also shown in Table I. We keep the same duty cycle (10%) for both LC-MAC and S-MAC, although it makes the whole cycle in LC-MAC longer than in S-MAC due to LC-MAC's Burst transmission. The duty cycle related settings are shown in Table I. In our simulations, we assume all the nodes in the network have already been synchronized to use a single wake-up and sleep schedule. We compared LC-MAC with S-MAC and S-MAC with adaptive listen mode [15].

A. Overview of Scenarios

Our simulation scenario is a chain scenario. All nodes are equally spaced in a striaight line, and relay nodes are placed 200m apart. For the first simulation as shown in Fig.6, the length of the chain is 9 hops. We choose three relay nodes randomly and three CBR(constant bit rate) flows and then send packets from these relay nodes to sink. For the second simulation as shown in Fig.7, the length of the chains varies from 4 hops to 9 hops. We choose the last two relay nodes and tow CBR flows and then send packets from these relay nodes to sink. The chain scenario helps us to study the protocols for multi-hop delivery.

B. Latency Evaluation

In this subsection, we evaluate the performance of latency using a typical light traffic load for sensor networks. For the first simulation, each CBR flow generates traffic loads varies from 4s to 10s interval. For the second simulation, each CBR flow generates a traffic load in 10s interval.The entire simulation both runs for 5050.0 seconds.

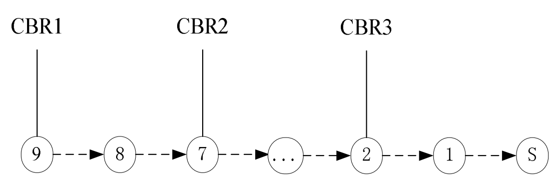


Figure 6. Chain scenario a.

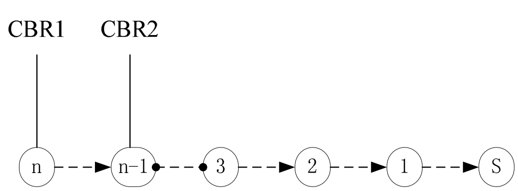


Figure 7. Chain scenario b.

Fig.8(a) shows, how the average packet latency varies with the package interval. For LC-MAC protocol, it keeps latency at a very low level from 4s to 10s interval. Latency in S-MAC and S-MAC with adaptive listen mode take several tens times to the hundreds times of the latency in LC-MAC at 4s intervl and decreases as the package interval increases. This shows the advantage of LC-MAC's capability of multi-hop delivery. It is because we used location detection to fix the route, the SSYNC message passing mechanism to reserve channel time and data transmission time, and the burst transmission mechanism to relay data packages at a very fast manner without collision. This capability can be better presented in Table II. Fig.8(b) shows, how the average packet latency varies with the path length. Latency in S-MAC and S-MAC with adaptive listen mode increases rapidly as the path length increases.But LC-MAC also keeps the latency at a very low level. This shows LC-MAC can keep capability of multi-hop delivery from 4 to 9 hops.

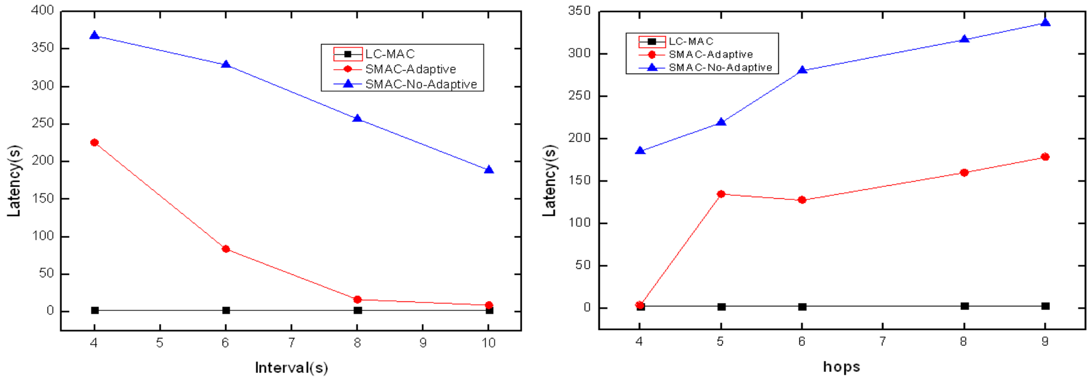
TABLE I. SIMULATION PARAMETERS

NETWORKING PARAMETERS			
Bandwidth	20Kbps	Carrier Sensing Range	550 m
Receive Power	14.4 mW	DIFS*	10 ms
Transmit Range	250 m	SIFS	5 ms
Transmit Power	36 mW	Slot size	1ms
Idle Power	0.45W	CW_{DATA}	64ms
Sleep Power	0.015 mW	CW_{SYNC} (LC-MAC)	4ms
CW(Contention Window)		CW_{SYNC} (S-MAC)	32ms
*DIFS(Distributed Inter-Frame Spacing)			
TRANSMISSION DURATION PARAMETERS			
	Frame Size (bytes)	Transmit Latency (ms)	
RTS/CTS	10	11.0	
ACK	10	11.0	
SSYNC	14	14.2	
DATA	50	43.0	
CYCLE DURATION PARAMETERS			
	Tsync (ms)	Tdata (ms)	Tsleep (ms)
S-MAC	55.2	88.0	1288.8
LC-MAC	31.2	88.0	1072.8

C. Energy Consumption Evaluation

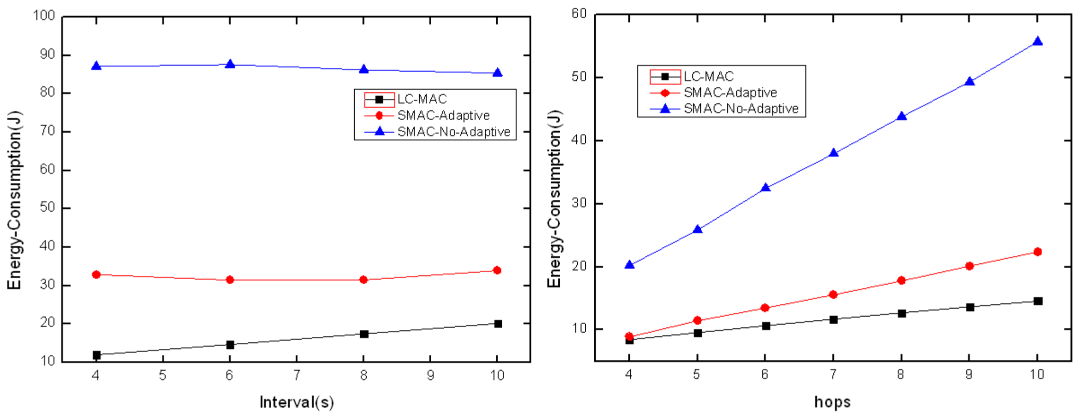
In this subsection, we evaluate the energy effciency of LC-MAC. We use a typical light traffic load for sensor networks. For the first simulation, each CBR flow generates traffic loads varies from 4s to 10s interval. For the second simulation, each CBR flow generates a traffic load in 10s interval.The entire simulation both runs for 5050.0 seconds.

Fig.9 shows the average energy-consumption over all the sensors in the chain scenario. Average energy-consumption is calculated by dividing the total energy consumed (by the sensors) by the total simulated time. Fig.9(a) shows when the path length is 9, relay nodes in LC-MAC consume less energy as those in S-MAC and S-MAC with adaptive listen mode. Fig.9(b) shows when the path length varies from 4 to 9, LC-MAC also consumes less energy than S-MAC and S-MAC with adaptive listen mode. But the three protocols consume nearly same energy when the path length is short. This is because they use the same duty cycle(10%), thus having the same power effciency. As the traffic load increases, both LC-MAC and S-MAC increase their energy consumption, but LC-MAC has a smaller rate of increase than does S-MAC and S-MAC with adaptive listen mode.



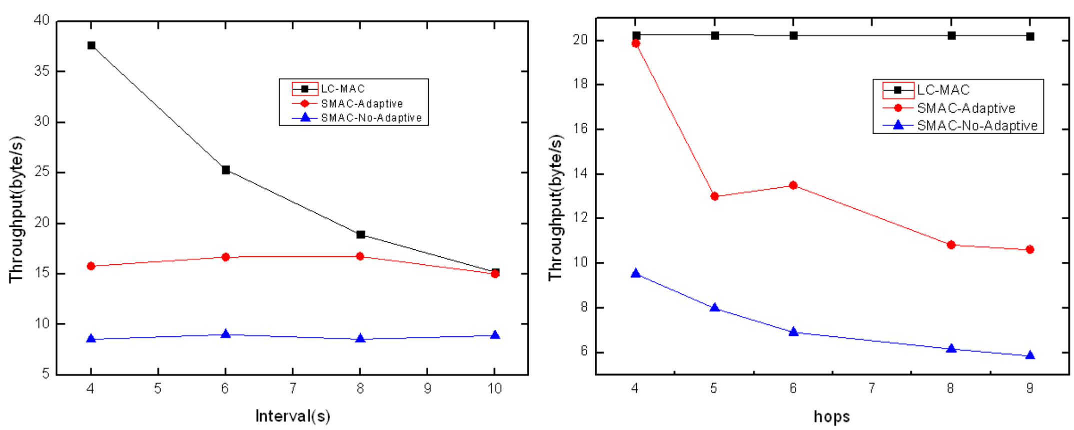
a. Interval-Latency　　　　b.Hops-Latency

Figure 8.　Latency in the chain scenario



a.Interval - Energy Consumption　　　　b. Hops - Energy Consumption

Figure 9.　Average energy-consumption of sensors in the chain scenario



a. Interval - Throughput　　　　b.Interval - Throughput

Figure 10. Throughput in the chain scenario

TABLE II. RESULTS OF 9-HOP NETWORK

	Latency (seconds)	*Tcycle* (seconds)	$\frac{Latency}{Tcycle}$ (cycles)	$\frac{PathLength \times Tcycle}{Latency}$ (hops/cycle)
S-MAC	336.9	1.432	235.26	0.039
S-MAC (Adaptive)	178.4	1.602	111.36	0.081
LC-MAC	2.81	1.192	2.36	3.82

D. Throughput Evaluation

In this subsection, we evaluate the network throughput using LC-MAC. Although network throughput is not a crucial metric in typical sensor networks, it is important when the traffic can potentially come in a burst. For the first simulation, we again use the 9-hop chain. We varied our traffic load in terms of the packet generation rate, from 2.5 packets every 10 seconds to 1 packet every 10 seconds. The simulation runs for 5050.0 seconds. Fig.10 shows our simulation result. Fig.10(a) shows, throughput in LC-MAC take several times to the serveral tens times of the throughput in S-MAC and S-MAC with adaptive listen mode at 4s intervl and decreases as the package interval increases. For the second simulation as shown in Fig.10(b), LC-MAC shows its better throughput in mutihop delivery than S-MAC and S-MAC with adaptive listen mode. For LC-MAC and S-MAC with adaptive listen mode, their throughput is same at 4 hops. It is because S-MAC with adaptive listen mode can relay a data package with 2 hops without sleep delay so that it increases throughput when the length of the chain is short. For LC-MAC, it can keep the level of throughput from 4 to 9 hops.This shows the advantage of LC-MAC's capability of heavier traffic delivery than S-MAC and S-MAC with adaptive listen mode.

V. CONCLUSION

The contribution of this article on a long-chain wireless sensor network is to create a more efficient protocol LC-MAC to reduce delivery latency substantially. The results are promising. LC-MAC performs much better than S-MAC with more than 99% decrease in delivery latency at best, a small amount decrease in energy consumption and a better throughput in heavier traffic and mutihop scenario as shown in Section IV. These came at the situation of a fixed long-chain topology. For applications which are time sensitive and the network topology is a long-chain, LC-MAC offers a solution for promoting the efficience of a wireless sensor network. In our simulation, LC-MAC outperforms S-MAC.

REFERENCES

[1] MC13211/212/213 ZigBee™- Compliant Platform - 2.4 GHz Low Power Transceiver for the IEEE® 802.15.4 Standard plus Microcontroller. Datasheet, Freescale Semiconductor , August 2009.

[2] System-on-chip for 2.4 GHz zigbee®/IEEE 802.15.4 with location engine. Datasheet, Texas Instruments,Dec 2009.

[3] Gahng-Seop Ahn,Se Gi Hong,Emiliano Miluzzo,Andrew T. Campbell,Francesca Cuomo, “Funneling-MAC: a localized, sink-oriented MAC for boosting fidelity in sensor networks,” in Proceedings of the 4th international conference on Embedded networked sensor systems, June 2006, pp. 293 - 306.

[4] W. Ye, J. Heidemann, and D. Estrin, “An energy-efficient MAC protocol for wireless sensor networks,” in Proceedings of IEEE INFOCOM, June 2002, pp. 1567–1576.

[5] Shih-Hsien Yang,Hung-Wei Tseng,Wu E.H.-K,Gen-Huey Chen, “Utilization based duty cycle tuning MAC protocol for wireless sensor networks,” in GLOBECOM, Dec.2005, 5 pp. – 3262.

[6] Tijs van Dam,Koen Langendoen, “An adaptive energy-efficient MAC protocol for wireless sensor networks,” in Proceedings of the 1st international conference on Embedded networked sensor systems, 2003, pp. 171–180.

[7] Lin P,Qiao C,Wang X, “Medium access control with a dynamic duty cycle for sensor networks,” Wireless Communications and Networking Conference, 2004, pp. 1534 - 1539 Vol.3.

[8] Lu G, Krishnamachari, B. Raghavendra, C.S., “An adaptive energy-efficient and low-latency MAC for data gathering in wireless sensor networks,” Parallel and Distributed Processing Symposium.18th International, April 2004, pp. 224.

[9] L. van Hoesel and P. Havinga, “A lightweight medium access protocol (LMAC) for wireless sensor networks: reducing preamble transmissions and transceiver state switches,” in: Proceedings of the International Conference on Networked Sensing Systems (INSS), 2004.

[10] W. L. Lee, A Datta, and R. Cardell-Oliver, “FlexiTP: A flexible-schedule-based TDMA protocol for aault-tolerant and energy-efficient wireless sensor networks,”, IEEE Trans. Parallel Distrib. Syst, vol 19, no. 6, pp. 851-864, Jun. 2008.

[11] F. Yu, T. Wu, and S. Biswas, “Toward in-band self-organization in energy-efficient MAC protocols for sensor networks,” IEEE Trans. Mobile Comput., vol. 7, no.2, pp.156-170, Feb. 2008.

[12] M. J. Miller and N. H. Vaidya, “A MAC protocol to reduce sensor network energy consumption using a wakeup radio,” IEEE Trans. Mobile Comput, vol 4, no 3, pp 228-242, May. 2005.

[13] G. Zhou, C. Huang, T. Yan, T. He, J. A. Stankovic, and T. F. Abdelzaher, “MMSN: Multi-frequency media access control for wireless sensor networks,” IEEE Infocom, pp. 1-13, Apr. 2006.

[14] Y. Wu, J. A. Stankovic, T. He, and S. Lin, “Realistic and Efficient Multi-Channel Communications in Wireless Sensor Networks,” IEEE Infocom, pp. 1867-1875, Apr. 2008.

[15] [15] Wei Ye,Heidemann, J, Estrin D, “Medium access control with coordinated adaptive sleeping for wireless sensor networks,” IEEE/ACM TRANSACTIONS ON NETWORKING, JUNE 2004, pp. 493 – 506 VOL. 12.