KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS VOL. 11, NO. 8, Aug. 2017

# CCDC: A Congestion Control Technique for Duty Cycling WSN MAC Protocols

**Beakcheol Jang[1] and Wonyong Yoon[2]**
[1]Department of Computer Science, Sangmyung University
Seoul, Republic of Korea
[2]Department of Electronic Engineering, Dong-A University
Busan, Republic of Korea
[e-mail: wyyoon@dau.ac.kr]
*Corresponding author: Wonyong Yoon

*Received December 20, 2016; revised March 15, 2017; accepted April 9, 2017;
published August 31, 2017*

## Abstract

Wireless Sensor Networks hold the limelight because of significant potential for distributed sensing of large geographical areas. The radio duty cycling mechanism that turns off the radio periodically is necessary for the energy conservation, but it deteriorates the network congestion when the traffic load is high, which increases the packet loss and the delay too. Although many papers for WSNs have tried to mitigate network congestion, none of them has mentioned the congestion problem caused by the radio duty cycling of MAC protocols. In this paper, we present a simple and efficient congestion control technique that operates on the radio duty cycling MAC protocol. It detects the congestion by checking the current queue size. If it detects the congestion, it extends the network capacity by adding supplementary wakeup times. Simulation results show that our proposed scheme highly reduces the packet loss and the delay.

**Keywords:** Congestion Control, Duty Cycling, Medium Access Control, Wireless Sensor Network

---

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2016R1D1A1B03930815).

**https://doi.org/10.3837/tiis.2017.08.003** **ISSN : 1976-7277**

## 1. Introduction

Wireless Sensor Networks (WSNs) comprise many autonomous sensors and one or more sinks and cooperatively monitor physical or environmental conditions [1]. They can be deployed easily and fastly because of their wireless and multi-hop routing functionalities. In result, they have been exploited in various applications that monitor plants and animals, natural phenomena, military battlefield surveillance, etc. Sensors are often deployed in large numbers in fastidious or unreachable areas, and it is difficult to put on new batteries; therefore the energy efficiency is considered to one of the most important design factors. Because WSNs are used to monitor something, the size of packets to be transmitted is small in many scenarios, and the idle listening time of the radio is long relatively. Medium Access Control (MAC) protocol is often designed to minimize the idle listening time based on the radio duty cycling mechanism that periodically turns off the radio. The energy consumption of the radio sleep state is much smaller than those of the other states (for example, In MICAz with CC2420 radio, the energy consumption in the sleep mode is only 0.003mW, but those in the others are bigger than 52.2mW [2, 6].), so the radio duty cycling mechanism is useful to save the energy in many scenarios of WSNs.

The main task of networking is to transmit packets successfully from the source to the destination while it is very important to reduce the energy consumption. Although the radio duty cycling mechanism is effective and often necessary to avoid the energy waste, it may cause the network congestion [2], when the traffic load is high relatively. The network congestion occurs when the offered load exceeds the network capacity. Typical problems are queuing delay and packet loss. In the duty cycling mechanism, the sensor turns off the radio based on its wakeup interval, the cycle that each sensor node turns off its radio; therefore the network capacity is limited by the wakeup interval directly. In most of duty cycling MAC protocols [2-8], wakeup intervals of all sensor nodes in the WSN are same. However, WSNs deliver various types of traffics, from simple periodic reports to unpredictable burst of packets. Even under periodic traffic patterns, offered loads of sensors are very different according to their locations (that is, the sensor near to the sink transmits many packets because it has many descendants, receives packets from them, and needs to forward them). Therefore, wakeup times of sensor nodes should be dynamically adaptable to the traffic load to avoid the network congestion.

The network congestion has received great attentions for last few of decades in the area of computer networks. Although many papers [9, 10, 11, 17] for WSNs have made every endeavor to solve it in their problem domains, none of them has mentioned the network congestion caused by the radio duty cycling of MAC protocols. The congestion control contains two essential steps, the congestion detection and mitigation. To detect the congestion, some mechanisms check the queue size, others sense the channel loading [9] and the others check the observed event reliability [10]. To mitigate the network congestion, some mechanisms control the data rate in the application [10, 17] and transport layers [9, 11], and others exploit the hop-by-hop flow control.

In this paper, we present a simple but effective Congestion Control mechanism for Duty Cycling WSN MAC protocols called to CCDC. To detect the network congestion, each sensor checks its current queue size. If the queue size is bigger than the preset threshold, the sensor regards a congestion occurs and informs it to its parent in the routing path. To avoid the congestion, the parent increases its network capacity by adding supplementary wakeup times until the congestion is extinct. The sensor transmits congested queued packets at additional wakeup times of the parent. To ensure the reliable packet transmission, we also present a collision detection and retransmission mechanism called to CCDC-ACK. We evaluate CCDC and CCDC-ACK through the simulation. It shows that our schemes highly improve the packet loss and the delay compared to the pure radio duty cycling MAC protocol. It also shows that CCDC is more efficient than the pure radio duty cycling MAC protocol in terms of the energy consumption when considering the energy consumption per successfully transmitted packet.

The remainder of this paper is organized as follows. In Section II, we review related works and define the problem handled in this paper. Section III describes the proposed schemes in detail. In Section IV, we evaluate our mechanisms. Finally we conclude the paper in Section V.

## 2. Related Work

In this section, we present related works and define the problem handled in this paper.

### 2.1 Related Works

Many papers have proposed novel and efficient mechanisms to control network congestions in wireless sensor networks.

[9] mentions that transport of event leads to various degrees of congestion in the network depending on sensing applications. It presents an energy-efficient congestion control mechanism called to CODA, which consists of three sub-mechanisms, receiver-based congestion detection, open-loop hop-by-hop backpressure, and closed-loop multi-source regulation. CODA detects congestions based on the present and past channel loading conditions and the current buffer occupancy. As long as each node detects congestion in CODA, it broadcasts backpressure message, which is propagated to the source. When the source event rate is bigger than the maximum theoretical throughput of the channel, a source is more likely to contribute to congestion and therefore closed-loop congestion control is triggered. The reception of ACKs at sources serves as a self-clocking mechanism allowing sources to maintain their current event rates. In contrast, failure to receive ACKs forces a source to reduce its own rate. Simulation results indicate that CODA mitigates congestions in various feasible congestion scenarios.

[10] also presents a novel energy-efficient reliable transport scheme for WSN, called to event-to-sink reliable transport (ESRT). If the event-to-sink reliability is lower than the threshold, ESRT reduces the data reporting period of source nodes to raise the reliability. If the reliability is higher than the threshold, then ESRT increases the data reporting period in order to conserve energy while maintaining reliability. Analytical evaluation and simulation result show that ESRT improves the reliability of WSNs with minimum energy expenses.

[17] argues that high priority traffic such as event reporting is generated only for a short period of time while low priority traffic such as periodic data reporting usually exists in the network. For such environment, service differentiation in wireless multimedia sensor networks (WMSNs) is important. It presents a priority based congestion control protocol for WMSNs which adjusts the source traffic rates based on current congestion and the priority of each traffic source. Simulation results show that the proposed mechanism achieves low packet loss probability and it provides low queuing delay and guarantees bandwidth for high priority real time traffic.

[11] argues that media access control of WSNs must allow fair bandwidth allocations to all nodes of WSNs while maintaining the energy efficiency. They present an adaptive rate control

scheme with the new CSMA mechanism to achieve the fairness and the energy efficiency without explicit control packets. Simulation results show that the proposed scheme is effective in achieving fairness with reasonable energy efficiency.

[19] presents a congestion-aware routing algorithm based on traffic priority for WSNs that reduces contestion by considering the priority of data. It classifies packets according to the data priority, in which traffic is redirected to control congestion in the network. Simulation result shows that the proposed mechanism reduces packet loss, energy consumption and buffer size.

Although they have proposed efficient techniques to avoid the network congestion for wireless sensor networks, none of them has considered the network congestion caused by the radio duty cycling of MAC protocols.

Some WSN MAC protocols [5, 6, 8, 18] vary their active times with the traffic load. The node that transmitted a packet keeps transmitting packets if it still has packets in its queue. The node that received a packet has an additional active time; therefore nodes can receive and transmit a series of packets at one time. Although their active times adapt to the traffic load, they cannot be considered to the congestion control mechanism because nodes always have additional active times regardless of the network congestion. Besides, their additional active times cause the energy waste when the traffic load is lower than the network capacity. [15] presents an efficient and unified receiver-driven MAC protocol integrated with several clever MAC-based rate control techniques, called to LET-MAC. Simulation results shows that LET-MAC provides higher energy efficiency at low traffic scenarios and higher medium utilization at high traffic situations than existing duty cycling MAC protocols. LET-MAC works based on control packets such as the Beacon, but our proposed mechanism does not need any control packet. [16] mentions the problem of the nonuniform traffic distribution of sensors in WSNs, analyzes the fairness of the tree-based WSN, and finally presents a fair data collection protocol. The simulation results show that the proposed scheme guarantees the fair delivery of packets and reduces the delay.

## 2.2 Problem Definition

Lots of WSN MAC protocols [2-8] run based on the radio duty cycling mechanism to save energy, in which the sensor can transmit or receive a packet only when it turns on its radio (i.e., the sensor can transmit or receive only one packet every wakeup interval), its network capacity, $N_{cap}$, cannot be bigger than the data rate and it is limited by the interval of the periodic wakeup, $I_w$, and the length of the data packet, $L_d$, as in:

$$N_{cap} = min(\frac{Ld}{Iw}\ ,\ R_d) \tag{1}$$

where $R_d$ is the data rate. Many of the MAC protocols [2-8] use a globally fixed wakeup interval in terms of both the time and the location. If the wakeup interval improper to the traffic load is set (that is, the wakeup time, that is, the network capacity is either small or big compared to the traffic load), the network congestion occurs, or the system is not energy-efficient any more. Unfortunately, it is difficult to set one optimal wakeup interval for the WSN because we cannot foresee its traffic load beforehand. WSN applications can be categorized into two types, the target tracking application and the periodic data sampling application. In case of the former, it is not possible to predict the movement of the target, and we cannot forecast the traffic load in advance. Even in case of the latter, traffic loads are different according to the location of the sensor, and we cannot choose one optimal wakeup interval for whole sensors. As the wakeup interval increases, the energy consumption decreases, but the network capacity decreases and the network congestion occurs frequently. It causes the packet drop by the queue overflow and increases the delay too. Therefore wakeup times of sensors should be dynamically adaptable to the traffic load to avoid the network congestion.

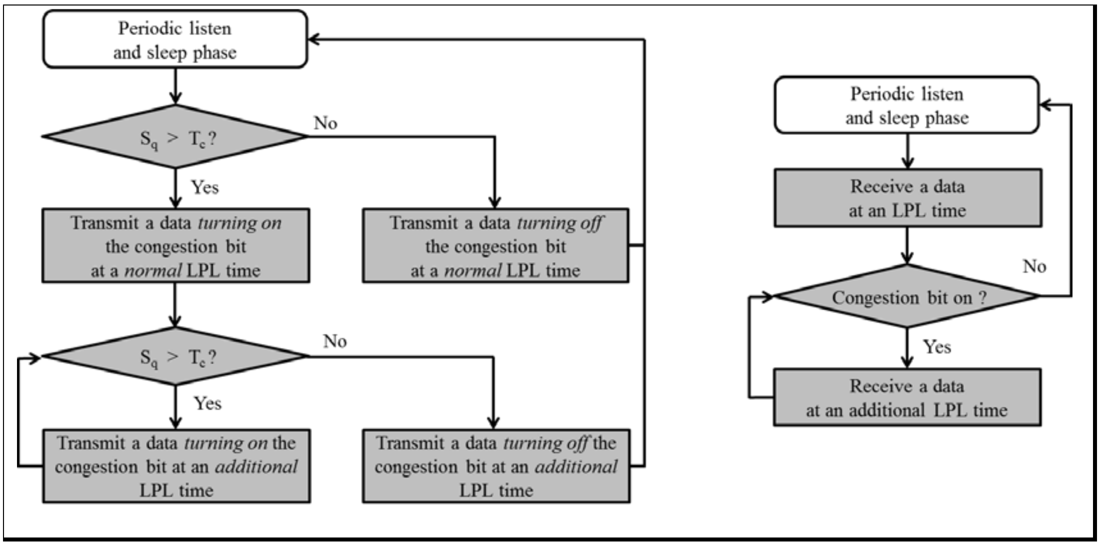
## 3. CCDC Protocol Design

The radio duty cycling mechanism is effective to avoid the energy waste in WSN MAC protocols, but it causes the network congestion [2]. Our Congestion Control mechanism for Duty Cycling WSN MAC protocols called to CCDC, dynamically enhances the network capacity by injecting additional wakeup times when the network congestion occurs. In this section, we present the design of CCDC.

### 3.1 CCDC Operation

CCDC has following two essential steps:

- Congestion Detection: To detect the congestion, each sensor checks the current size of its queue, $S_q$. If $S_q$ is bigger than the preset congestion threshold, $T_c$, the sensor regards a congestion occurs.
- Congestion Mitigation: To mitigate the congestion, if $S_q$ is bigger than $T_c$, the sensor informs the fact that the congestion occurs to its parent by turning on the congestion bit, $B_c$, of its outgoing packets to the parent. To CCDC, each packet should include a $B_c$ additionally. The parent enhances its network capacity by adding a supplementary wakeup time every preset short interval, $I_s$. The sensor transmits queued packets every $I_s$ during the time that $S_q$ is bigger than $T_c$. $I_s$ should be much smaller than the normal wakeup interval, $I_w$, the cycle that a sensor turns off its radio periodically in duty cycling MAC protocols.



(a) Sender (b) Receiver

**Fig. 1.** Flow chart of CCDC operation

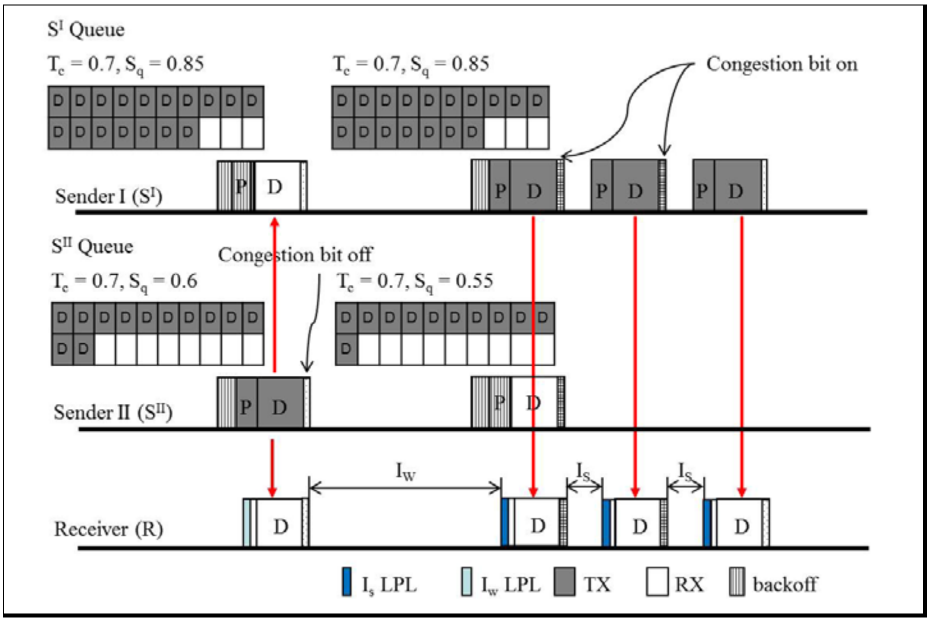


**Fig. 2.** Example of CCDC operation

**Fig. 1** presents the flow chart for CCDC operation. Each sensor performs the periodic listen and sleep, where the sensor periodically turns on its radio perform Low Power Listening (LPL) every wakeup interval, $I_w$. **Fig. 1**(a) describes CCDC operation that a sensor node acts as a sender. If a sender has a packet to send in its queue, it wakes up at its receiver's periodic wakeup time. It checks $S_q$ to detect the congestion. If $S_q$ is bigger than $T_c$, the sender regards the congestion occurs and transmits the packet turning on $B_c$. If not, it transmits the packet turning off $B_c$ and goes back to the periodic listen and sleep phase. If it transmitted the packet whose $B_c$ was turned on, it sleeps for $I_s$, wakes up and checks $S_q$ again. If $S_q$ is bigger than $T_c$, it turns on $B_c$ of an additional packet and transmits it. During the time that the $S_q$ is bigger than $T_c$, it repeats sleeping for $I_s$ and transmitting an additional packet turning on $B_c$. Once $S_q$ is smaller than or equal to $T_c$, it transmits an additional packet turning off $B_c$ and then goes back to the periodic listen and sleep phase. **Fig. 2**(b) describes CCDC operation that a sensor node acts as a receiver. A receiver wakes up every $I_w$ and receives a packet. If it receives a packet whose $B_c$ is turned on, the receiver sleeps during $I_s$, wakes up and receives an additional packet. If not, it goes back to the periodic listen and sleep phase. During the time that the receiver gets additional packets whose $B_c$ is turned on, it repeats sleeping for $I_s$ and receiving an additional packet. If not, it goes back to the periodic listen and sleep phase.

**Fig. 2** shows an example of CCDC operation with AS-MAC [2]. In AS-MAC, the node periodically wakes up at asynchronously scheduled time from its neighbors and performs LPL every $I_w$ to receive the packet. It sends Hello packet every Hello interval to publish its scheduling information too. Hello interval is the multiple of $I_w$. The node stores its neighbors' scheduling information in its own neighbor table. Based on it, a sender wakes up at the unique wakeup time of the receiver and sends the packet. **Fig. 3** and **5** presents examples of CCDC operation with AS-MAC too. In **Fig. 2**, sender I, $S^I$, sender II, $S^{II}$, and receiver, R, are in the same communication range. R wakes up every $I_w$ and performs LPL. $T_c$ is set to 0.7. At the first LPL time of R, $S^I$ and $S^{II}$ have packets in their queues. They perform a random back-off to avoid the collision. $S^{II}$ chooses a smaller back-off than the back-off of $S^I$, and $S^I$ postpones its packet transmission to the next regular wakeup time of R. $S^{II}$ transmits a packet, but its $S_q$ is smaller than $T_c$ (e.g., 0.6 is smaller than 0.7). $S^{II}$ regards that the congestion does not occur transmits a packet turning off $B_c$ and goes back to the periodic listen and sleep state. R gets a packet. It sees that $B_c$ is turned off, so it goes back to the periodic listen and sleep state too. At the second LPL time of R, $S^I$ and $S^{II}$ have packets in their queues too. They perform random back-off respectively. $S^I$ chooses a smaller back-off than that of $S^{II}$ in this time, and $S^I$ transmits a packet, in which $S_q$ is bigger than $T_c$ (e.g., 0.85 is bigger than 0.7). $S^I$ transmits a packet turning on $B_c$ and it repeats transmitting an additional packet every $I_s$ until $S_q$ becomes smaller than or equal to $T_c$ (e.g., two times). R gets the packet whose $B_c$ is turned on and it wakes up every $I_s$ and receives additional packets until it receives the packet whose $B_c$ is turned off (e.g., two times). The random back-off is required when a packet is sent at the regular wakeup time to avoid that senders transmit packets simultaneously as in [2]. However, it is not required when a packet is sent at the additional wakeup time because only one sender that choses the smallest back-off in the normal wakeup time transmits additional packets in supplementary wakeup times. When several senders choose the same smallest back-off, a collision occurs. To solve the problem, we suggest the collision detection and retransmission mechanism in Section 3.2.

The reason why we choose AS-MAC as the reference MAC protocol for CCDC is that AS-MAC is considered to one of very energy efficient scheduled duty cycling MAC protocols for WSNs [12, 13]. CCDC can be applied to many duty cycling MAC protocols without big modifications, not only scheduled MAC protocols [4, 6] but also preamble based MAC protocols [3, 5]. For example, CCDC can be applied to preamble based MAC protocols [3, 5] as follows. When the congestion occurs, senders transmit additional packets with preambles whose size is larger than $I_s$ every $I_s$. Receivers add supplementary wakeup times (that is, LPLs) during the congestion.

### 3.2 Collision Detection and Retransmission Mechanism

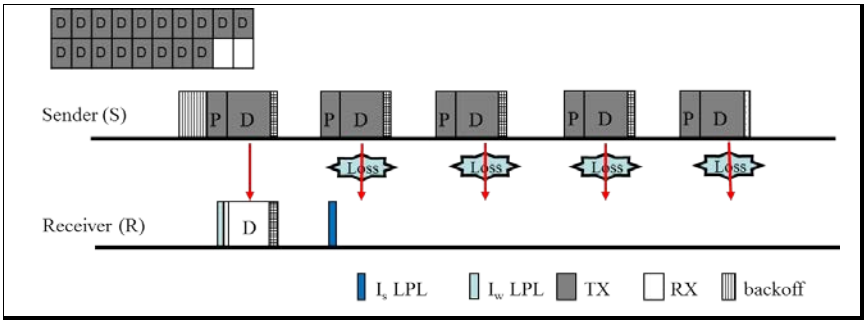
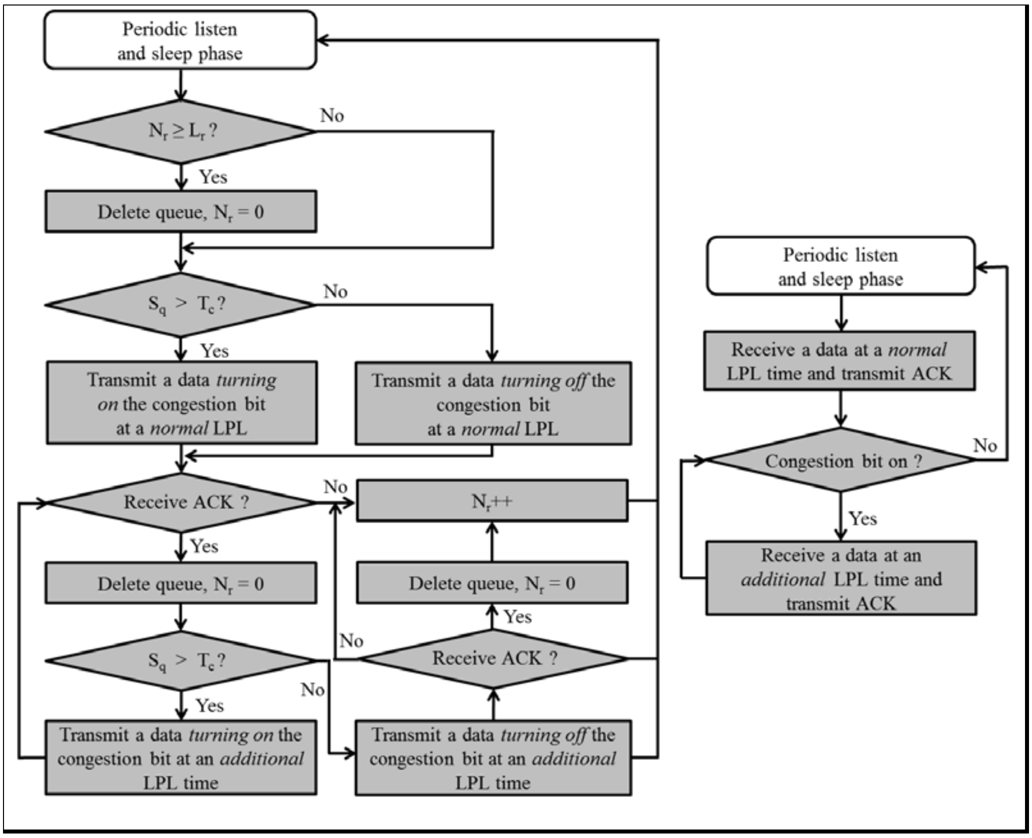


**Fig. 3.** Problem of CCDC operation

**Fig. 3** presents the problem of CCDC operation in terms of the reliable packet transmission. Sender, S, and receiver, R, are in the same communication range. At the normal LPL time of R, S has packets in its queue. It performs the random back-off and checks $S_q$. $S_q$ is bigger than $T_c$ (e.g., 0.9 is bigger than 0.7), and S transmits consecutive packets until $S_q$ becomes smaller than or equal to $T_c$. In the second packet transmission, the packet was lost because of a collision, channel noise or hidden terminal problem [14].

R could not get a correct packet and goes back to the periodic listen and sleep state. S keeps transmitting additional packets because $S_q$ is bigger than $T_c$, and consecutive packet losses occur (e.g., four consecutive packet losses in **Fig. 3**). As the channel condition becomes worse, more consecutive packet losses occur. To avoid the fatal problem, we introduce the collision detection and retransmission mechanism for CCDC, called to CCDC-ACK. The goal of CCDC-ACK is to guarantee the reliable packet transmission rather than the congestion control.



(a) Sender (b) Receiver

**Fig. 4.** Flow chart of CCDC-ACK operation

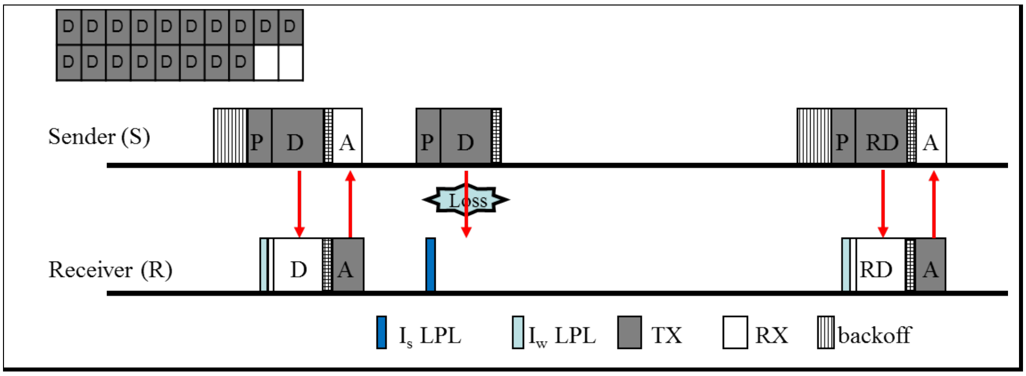


**Fig. 5.** Example of CCDC-ACK operation

**Fig. 4** presents the flow chart of CCDC-ACK. As shown in **Fig. 4**(b), a receiver transmits an acknowledgement packet, ACK, whenever it receives a packet. As shown in **Fig. 4** (a), a sender transmits a packet, and then it waits ACK from the receiver. If it receives ACK, it tries to transmit a next queued packet every $I_s$ according to CCDC operation. If not, it increments the number of retransmissions, $N_r$, and goes back to the periodic listen and sleep state. Whenever the sender has packets in its queue, it checks if $N_r$ is equal to the preset retransmission limit, $L_r$. If so, it deletes a packet from its queue and sets $N_r$ to zero. If not, it transmits a packet according to CCDC-ACK operation. If the sender does not receive ACK, it goes back to the periodic listen and sleep state at once; therefore the problem of consecutive packet losses shown in **Fig. 3** is avoidable. **Fig. 5** presents an example of CCDC-ACK. In the second packet transmission, the sender, S, does not receive ACK and goes back to the periodic listen and sleep state at once. At the next LPL time of the receiver, R, S retransmits the packet and receives ACK from R. The purpose of CCDC-ACK is to ensure the reliable packet transmission. If the channel condition is good, CCDC-ACK may not be effective. ACK of CCDC-ACK deteriorates the system throughput. In many WSN scenarios, the packet size is small, and CCDC-ACK may be a burden especially when the channel condition is good.

## 4. Evaluation

The network congestion is caused by the radio duty cycling mechanism. CCDC reduces the network congestion by injecting additional wakeup times when it detects the network congestion. To evaluate the performance of CCDC, we implemented simulation codes in NS2. We implemented a pure duty cycling MAC protocol (i.e., AS-MAC), CCDC and CCDC-ACK operations respectively. We use the multi-hop WSN topology that consists of ten sensor nodes, with tenth node as the sink at one end of the network. All sensors except the sink generate a packet every data generation interval. Each node receives packets from its child (for example, sixth node receives packets from fifth node) and transmits packets that it generates and receives to its parent (for example, sixth node transmits packets to seventh node). All sensors including the sink perform the radio duty cycling. We evaluate the performance of CCDC in terms of three traditional evaluation metrics: packet loss, delay and energy consumption. As the parameter, we use the wakeup interval, $I_w$. $I_w$ is very important as the parameter in WSN duty cycling MAC protocols. As $I_w$ increases, the energy consumption and the network capacity decrease. In the experiment, we set the data generation interval to one second, the queue size to thirty and the retransmission limit of CCDC-ACK to five. Sensors generate a packet every data generation interval thirty times. Each simulation lasts for 200 seconds, and it is performed ten times. In last, we calculate average values for each case. The reason why we choose the periodic data sampling application and the simple chain multi-hop topology rather than the target tracking application that traffic loads change and large complex topologies such as grid or random networks is that the effect of CCDC is obvious in the latter scenario more than in the former scenario.

### 4.1 Packet Loss

**Fig. 6** shows the average end-to-end packet loss rate as a function of $I_w$ for a pure radio duty cycling MAC protocol (AS-MAC), CCDC and CCDC-ACK. If a packet had been generated from the source but was not received by the sink during the running time of the experiment, we regarded that the packet was lost. The packet loss rate of AS-MAC is much larger than those of both CCDC and CCDC-ACK. The large packet loss rate in AS-MAC is because AS-MAC sends only one packet per $I_w$. On the other hands, both CCDC and CCDC-ACK may transmit a series of packets per $I_w$ if they recognize the congestion (that is, the queue size, $S_{q,}$ is bigger than the congestion threshold, $T_c$). The packet loss rate of CCDC-ACK is larger than that of CCDC. It is because the simulation considers the situation the congestion occurs and the channel noise does not exist. As the channel noise increases or the congestion reduces, the packet loss rate of CCDC-ACK will be smaller than CCDC. The performance of AS-MAC is barely affected by the fixed allocation of the offset of $I_w$ because AS-MAC sends only one packet per $I_w$; therefore we performed the simulation only one time for each $I_w$ for AS-MAC. To decrease the packet loss rate in AS-MAC, small $I_w$ can be chosen. However, the system will not be energy efficient especially in edge nodes far from the sink that have low traffic loads.

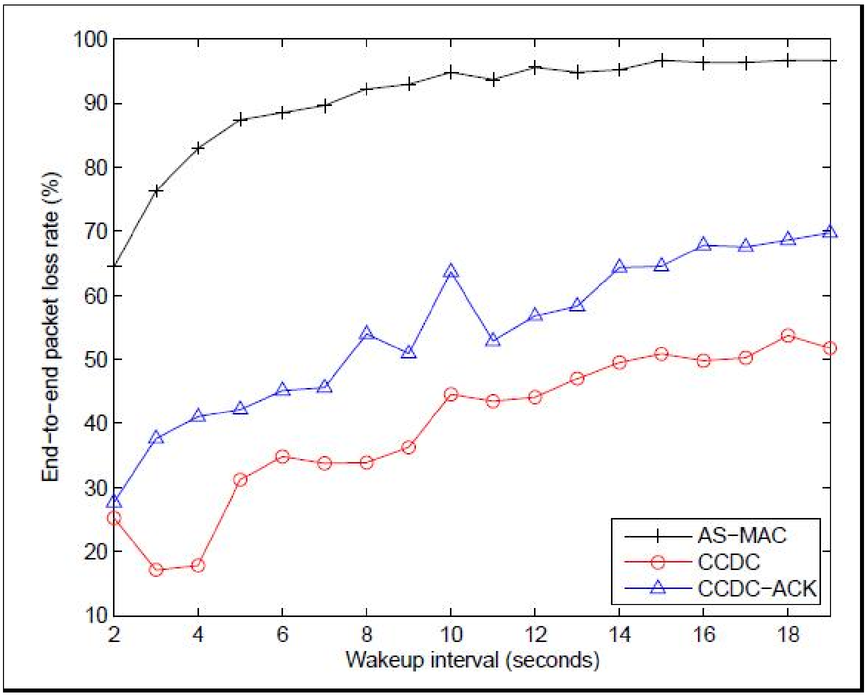


**Fig. 6.** End-to-end packet loss rate

### 4.2 Delay

**Fig. 7** shows the average end-to-end delay as a function of $I_w$ for AS-MAC, CCDC and CCDC-ACK. We excluded delays of lost packets in the calculation of the average delay. The reason why delays in AS-MAC are excessively long (that is, they are much larger than the product of $I_w$ and the hop count) is that queuing delays occur by its limited transmission capacity (i.e., only one data packet transmission per $I_w$). Delays of both CCDC and CCDC-ACK are much smaller than those of AS-MAC. It is because both CCDC and CCDC-ACK allow a sender to transmit a series of queued packets per $I_w$ if the congestion occurs.

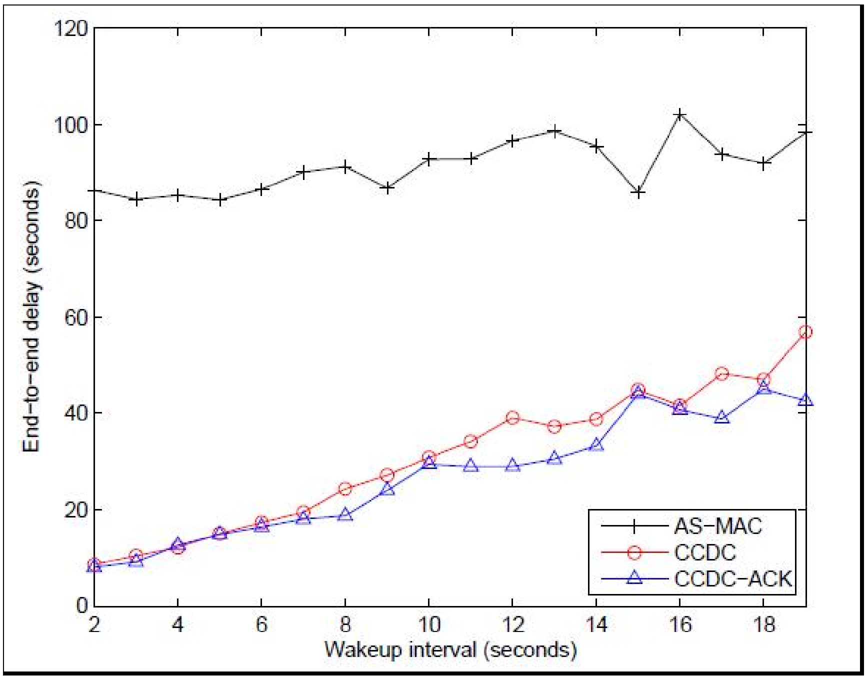


**Fig. 7.** End-to-end delay

### 4.3 Energy Consumption

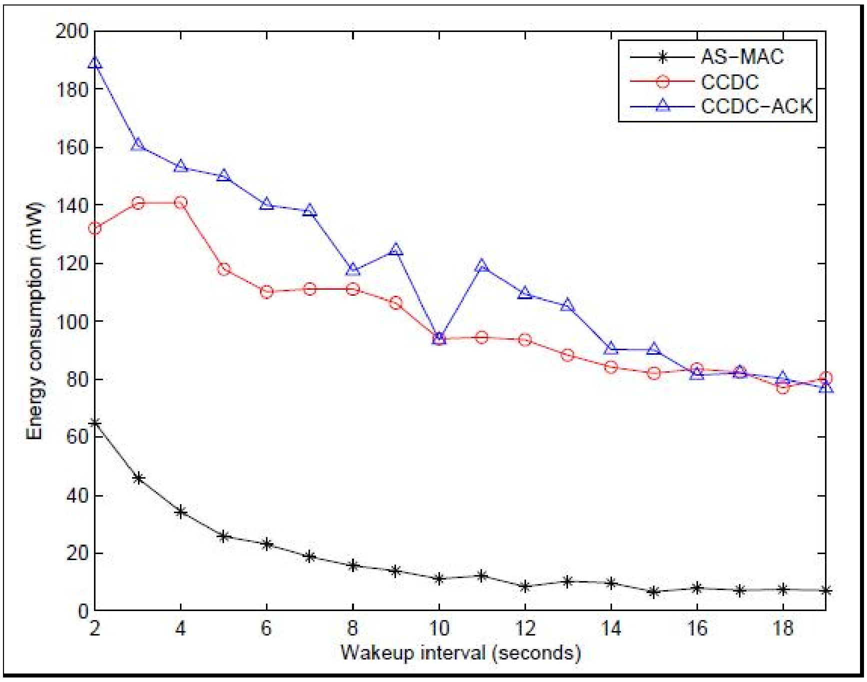


**Fig. 8.** Energy consumption

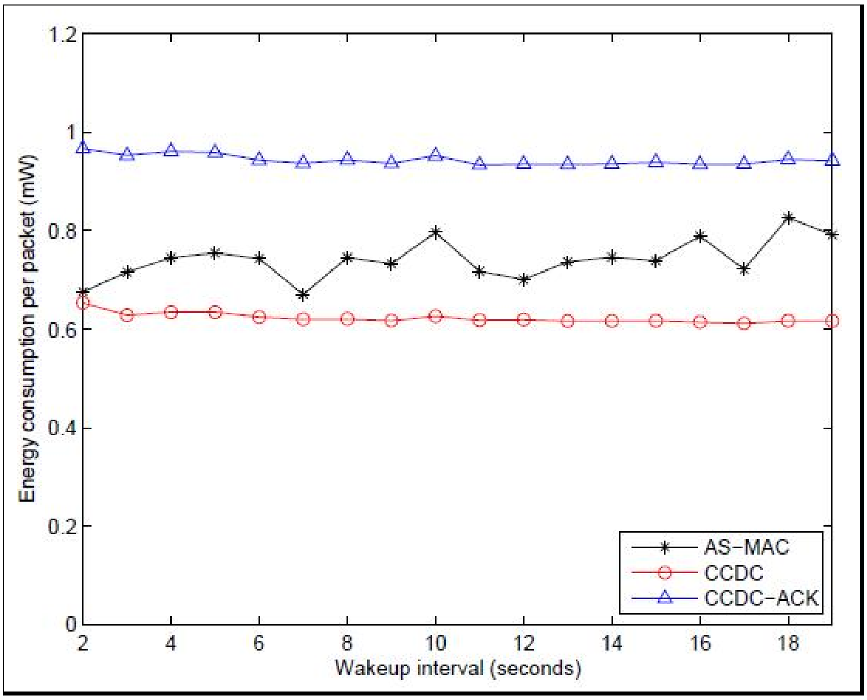


**Fig. 9.** Energy consumption per transmitted packet

**Fig. 8** shows the average energy consumption of the tenth node in **Fig. 6** as a function of $I_w$ for AS-MAC, CCDC and CCDC-ACK. To measure the energy consumption, we accumulated time in each state of the radio considering the energy consumption in each state of MICAz [2] and computed the total energy consumption at the end of the experiment. Energy consumptions of CCDC and CCDC-ACK look greater than that of AS-MAC. It is because numbers of transmitted and received packets in CCDC and CCDC-ACK are much greater than those in AS-MAC as shown in **Fig. 7**; therefore they must not be considered to energy wastes. Energy consumptions of CCDC-ACK are slightly greater than that of CCDC. It is because of transmissions and receptions of acknowledgement packets, ACKs, in CCDC-ACK.

**Fig. 9** shows the average energy consumption of the tenth node per successfully transmitted packet as a function of $I_w$ for AS-MAC, CCDC and CCDC-ACK. To measure the energy consumption per successfully transmitted packet, we divided total energy consumptions in **Fig. 8** to corresponding numbers of packets that the tenth node successfully received respectively. The energy consumption per successfully transmitted packet of CCDC is smaller than that of AS-MAC. It obviously shows that the energy efficiency of CCDC is better than that of AS-MAC. The reason why energy consumptions per successfully transmitted packet of CCDC-ACK are greater than those of AS-MAC is transmissions and receptions of ACKs in CCDC-ACK.

## 5. Conclusion

The radio duty cycling mechanism is good for the energy conservation in the wireless sensor network, but it may cause the network congestion especially when the traffic load is high. In this paper, we present a simple and efficient congestion control mechanism for duty cycling wireless sensor network MAC protocols called to CCDC, in which nodes additionally transmit and receive packets increasing the network capacity during they recognize the network congestion. Simulation results show that our proposed mechanism considerably reduces the packet loss and the delay. CCDC consumes more energy than the pure duty cycling MAC protocol because it requires additional wakeup times. However, CCDC is more energy efficient as shown in the simulation result when the energy consumption per successfully transmitted packet is considered.

## References

[1] B. Jang, "Wireless mac protocol design and analysis," *PhD Thesis, Computer Science*, North Carolina State University, 2009. Article (CrossRef Link).

[2] B. Jang, J. Lim, and M. L. Sichitiu, "An asynchronous scheduled MAC protocol for wireless sensor networks," *Elsevier Computer Networks*, vol. 57, no 1, 2013, pp. 85-98, also in *Proc. of IEEE Conference on Mobile Ad Hoc and Sensor Systems (MASS)*, pp. 434-441, 2008. Article (CrossRef Link).

[3] 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 Systems (SenSys)*, pp. 95-107, 2004. Article (CrossRef Link).

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

[5] 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 Conference on Embedded Networked Sensor Systems (SenSys)*, pp. 307-320, 2006. Article (CrossRef Link).

[6] W. Ye, F. Silva, and J. Heidemann, "Ultra-low duty cycle MAC with scheduled channel polling," in *Proc. of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pp. 321-334, 2006. Article (CrossRef Link).

[7] P. Dutta, S. Dawson-Haggerty, Y. Chen, C. M. Liang, and A. Terzis, "Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless," in *Proc. of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pp. 1-14, 2010. Article (CrossRef Link).

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

[9] C. Y. Wanm, S.B. Eisenman, and A. T. Campbell, "CODA: Congestion detection and avoidance in sensor networks," in *Proc. of ACM Conference on Embedded Networked Sensor Systems (SenSys)*, pp. 266-279, 2003. Article (CrossRef Link).

[10] O. B. Akan and I. F. Akyildiz, "ESRT: Event-to-sink reliable transport for wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 13, no. 5, pp. 1003-1016, 2005, also in *Proc. of ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc)*, pp. 177-188, 2003. Article (CrossRef Link).

[11] A. Woo and D. E. Culler, "A transmission control scheme for media access in sensor networks," in *Proc. of ACM Annual International Conference on Mobile Computing and Networking (MOBICOM)*, pp. 221-235, 2001. Article (CrossRef Link).

[12] A. Sfairopoulou M. Oliver C. Cano, B. Bellalta, "Low energy operation in WSNs: A survey of preamble sampling MAC protocols," *Elsevier Computer Networks*, vol. 55, no. 15, pp. 3351-3363, 2011. Article (CrossRef Link).

[13] S. Soltani M. W. Mutka N. Xi P. Huang, L. Xiao, "The evolution of MAC protocols in wireless sensor networks: A survey," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pp. 101-120, 2013. Article (CrossRef Link).

[14] B. Jang and M. L. Sichitiu, “IEEE 802.11 saturation throughput analysis in the presence of hidden terminals,” *IEEE/ACM Transactions on Networking*, vol. 20, no. 2, pp. 557-570, 2012. Article (CrossRef Link).

[15] M. M. Monowar, M. M. Alam, Md. O. Rahman, C. Hong, and S. Lee, “A load-aware energy-efficient and throughput-maximized asynchronous duty cycle MAC for wireless sensor networks,” *Annals of Telecommunications*, vol. 65, no. 11, pp. 777-794, 2010. Article (CrossRef Link).

[16] Md. A. Hamid, M. M. Alam, Md. S. Islam, C. Hong, S. Lee, “Fair data collection in wireless sensor networks: analysis and protocol,” *Annals of Telecommunications*, vol.65, no. 7, pp. 433-446, 2010. Article (CrossRef Link).

[17] M. H. Yaghmaee and D. A. Adjeroh, “Priority-based rate control for service differentiation and congestion control in wireless multimedia sensor networks,” *Elsevier Computer Networks*, 2009. Article (CrossRef Link).

[18] J. Lim, B. Jang, and M. L. Sichitiu, “A multichannel asynchronous scheduled MAC protocol for wireless sensor networks,” *Elsevier Computer Communications*, vol. 56, no. 1, pp. 98-107, 2015. Article (CrossRef Link).

[19] O. Mohammad, V. Ayatollahitafti, and M. Farahmand, "A congestion-aware routing algorithms based on traffic priority in wireless sensor networks,” *Journal of Soft Computing and Decision Support Systems*, 2017. Article (CrossRef Link).

**Beakcheol Jang** received the B.S. degree from Yonsei University in 2001, the M.S. degree from Korea Advanced Institute of Science and Technology in 2002, and the Ph.D. degree from North Carolina State University in 2009, all in Computer Science. He is currently employed as an assistant professor in Department of Computer Science, Sangmyung University, Seoul, South Korea. His primary research interest is in wireless networking with an emphasis on ad hoc networking, wireless local area networks, and mobile network technologies.

**Wonyong Yoon** received his B.S. and M.S. degrees in computer engineering from Seoul National University, South Korea, in 1996 and 1998, respectively, and his Ph.D. degree in computer engineering from Information and Communications University, South Korea, in 2002. From 2002 to 2004, he was a senior researcher at Samsung Electronics, Advanced Institute of Technology, South Korea. From 2004 to 2007, he was a visiting postdoctoral researcher at the University of Illinois at Urbana-Champaign, USA. From 2008 to 2011, he was a chief research engineer and part leader at LG Electronics, Mobile Communications Research, South Korea. Since 2011, he has been an associate professor at the Department of Electronic Engineering, Dong-A University, South Korea. He was a voting member of IEEE 802.15 WG in 2003-2004, a voting member of MBOA alliance in 2004, a member of WiMAX Forum in 2008, and a founding member of VoLGA Forum in 2009.