Asym-MAC: A MAC Protocol for Low-Power Duty-Cycled Wireless Sensor Networks with Asymmetric Links

Myounggyu Won, *Member, IEEE,* Taejoon Park, *Member, IEEE,* and Sang H. Son, *Fellow, IEEE*

***Abstract*—Duty-cycling is a primary technique significantly improving the energy efficiency of Wireless Sensor Networks (WSNs). Thus, a large number of MAC protocols have been developed for duty-cycled WSNs. Especially, recently proposed receiver-initiated MAC protocols are well suited for extremely low duty-cycled WSNs. However, as it is shown in this paper, they perform poorly when highly asymmetric links are present. In this paper, a hybrid MAC protocol called Asym-MAC is proposed which takes advantage of both receiver-initiated and sender-initiated MAC protocols to combat performance degradation due to asymmetric links. Experimental results demonstrate that Asym-MAC has up to 2.8X higher packet reception ratio and up to 66.7% smaller packet transmission delay compared with A-MAC, the state-of-the-art receiver-initiated MAC protocol.**

***Index Terms*—Wireless sensor networks, MAC protocols.**

I. Introduction

WIRELESS Sensor Networks (WSNs) comprise of a multitude of nodes with wireless communication, computation, and sensing capabilities. WSNs have been deployed for diverse applications in medical, military, and manufacturing fields revolutionizing the way humans perceive physical environments. In recent years, WSNs have become a key component for Cyber-Physical Systems (CPS) [1].

Energy-efficiency is a fundamental issue in WSNs because nodes are usually battery-powered due to the ubiquitous characteristics of WSNs. One major breakthrough for promoting energy-efficiency is duty-cycling that allows nodes to periodically cycle between a sleep state and an awake state. In the sleep state, nodes completely turn off their radio and processing hardware modules, thereby significantly saving energy–the current consumption of a typical Mica2/TelosB node operated by two AA batteries is 216μA/21μA, 3.2mA/2.4mA, 7.0mA/21mA, and 8.5mA/23mA for sleep, idle listening, transmission (TX at 0dBm), and reception (RX) modes, respectively [2][3]. There are mainly two types of duty-cycling mechanisms: synchronous and asynchronous duty-cycling. In synchronous duty-cycling, all nodes have synchronized duty cycles, whereas, in asynchronous duty-cycling, nodes have their own duty cycles. Between the two, asynchronous duty cycling has been more popular because synchronous duty-cycling requires time synchronization. Various MAC protocols have been proposed for asynchronously duty-cycled WSNs. For example, LB-MAC [4] is designed to prolong network lifetime; DutyCon [5] provides end-to-end delay guarantee.

Manuscript received December 3, 2013. The associate editor coordinating the review of this letter and approving it for publication was S. Vural.

The authors are with the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, South Korea (e-mail: {mgwon, tjpark, son}@dgist.ac.kr).

T. Park and S. H. Son equally contributed to this work.

Digital Object Identifier 10.1109/LCOMM.2014.032014.132679

TABLE I
Percentage of Asymmetric Links [10]

Tx Power	Symmetric	Asymmetric	Unidirectional
-19dBm	50%	43%	7%
-14dBm	65%	22%	13%
-5dBm	88%	6%	6%

Recent studies are largely categorized into transmitter-initiated and receiver-initiated mechanisms. Transmitter-initiated MAC protocols like X-MAC [6] and BoX-MACs [7] rely on transmission of *preamble packets*, thus suffering from wasted energy for sending the preamble packets. On the other hand, in receiver-initiated MAC protocols like Ri-MAC [8] and A-MAC [9], a receiver broadcasts a probing packet whenever it wakes up from the sleeping state, while a sender with data to transmit waits in the listening state until a probing packet from the indented receiver is received. Once the probing packet is received, the sender begins its data transmission. This way receiver-initiated MAC protocols avoid the overhead of sending preamble packets.

However, as it is shown later, receiver-initiated MAC protocols perform poorly in networks with asymmetric links. A problem arises when a sender fails to receive a probing packet from the intended receiver. Especially, if a probing packet is missed multiple times due to the asymmetric link, the sender must stay in the listening state until the probing packet arrives, wasting energy and increasing delay. In the worst case of a unidirectional link (or a very highly asymmetric link), data packets are dropped, degrading PRR.

In this paper, Asym-MAC, a MAC protocol designed for low-power duty-cycled WSNs with asymmetric links is proposed. A key idea is to dynamically switch between two modes of operation: transmitter-initiated and receiver-initiated modes depending on the asymmetry of a link which is measured as a PRR difference between the two directions (up/down links) – a larger PRR difference means higher asymmetry [10]. More specifically, the receiver-initiated mode is used as the default MAC protocol; when the link is determined to be asymmetric, Asym-MAC switches its mode to the transmitter-initiated protocol. The hybrid method is shown to achieve up to 66.7% smaller delay and 2.8X higher packet reception ratio compared with the state-of-the-art receiver-initiated MAC protocol A-MAC, in a given experimental setup.

II. Motivation

Recent studies have shown that a significant portion of radio links are asymmetric in a typical WSN deployment [10]. Table I shows the percentage of asymmetric links measured in Kansei testbed [11]. With smaller TX power, the testbed

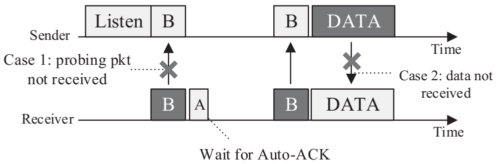

Fig. 1. Operation of a receiver-initiated MAC protocol.

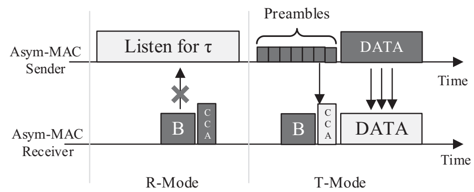

Fig. 2. Operation of the Asym-MAC protocol.

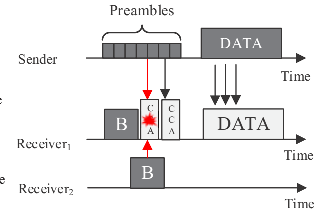

Fig. 3. A collision scenario.

had more asymmetric links, and a non-negligible portion of the links were unidirectional.

Figure 1 illustrates how asymmetric links may degrade the performance of receiver-initiated MAC protocols. The receiver periodically broadcasts a probing packet denoted by B. If B is received by the sender, the sender starts its data transmission denoted by $DATA$. However, due to the asymmetry of the link, two potential failure scenarios may arise. The first case (Case 1) is when B is failed to be delivered to the sender. In this case, the sender has to wait for the next probing packet in the listening state, consuming time and energy. The second case (Case 2) arises when the sender fails to transmit data due to the low-quality link from the sender to the receiver (i.e., the down-link). The focus of this paper is on addressing the issue of the asymmetry of the link from the receiver to sender (i.e., the up-link). However, it is worth to note that a simple yet effective strategy, e.g., increasing transmit power or increasing the number of retransmissions, can be adopted to address Case 2, thereby attempting to address both cases.

III. Protocol Design

The objective of this work is to design a MAC protocol that works well under the circumstances of highly asymmetric links. To achieve this goal, a hybrid approach that leverages the advantages of both transmitter-initiated and receiver-initiated MAC protocols is employed. Specifically, Asym-MAC reduces the overhead of preamble packets by relying on the design of a receiver-initiated MAC protocol. At the same time, it addresses the issue of asymmetric links by appropriately integrating the design principles of a transmitter-initiated MAC protocol into the receiver-initiated MAC protocol.

Figure 2 illustrates how Asym-MAC works. Asym-MAC has two modes of operation: *R-mode* and *T-mode*. *R-mode* is the default mode in which a receiver-initiated MAC is used. Asym-MAC switches its mode to *T-mode* when it fails to receive a probing packet from the intended receiver – more specifically, when a timeout (waiting time exceeding a probing period) occurs for receiving the probing packet τ times, where τ is a system parameter. In *T-mode*, the transmitter sends a series of small preamble packets to notify the receiver of the pending data transmission. Once the receiver captures the signal of on-going transmission of preamble packets, it receives data from the sender.

A challenge is that, while the sender knows when to change its mode to *T-mode* (i.e., when a timeout for receiving a probing packet from the intended receiver occurs more than τ times), the receiver has no idea about when to change the mode. A key idea to address this challenge is basically to add a short clear channel assessment (CCA) period at the end of each probing packet transmission. Adding the CCA period does not degrade the energy efficiency much given the facts that: first, the CCA check lasts for only a very short period of time, and second, the CCA check consumes an order of magnitude smaller energy than the TX and RX modes [9]. In each CCA period, the receiver checks for potential preamble packet transmission from the transmitter who is in *T-mode*, so that the receiver receives data regardless of the quality of the up-link. The pseudo-code of Asym-MAC presented in Algorithm 1 summarizes the operation of Asym-MAC.

Algorithm 1 Asym-MAC

/* Sender s sending data to Receiver r */
if A probing packet received from r **then**
    Send data to r.
else if Number of timeout $> \tau$ **then**
    Switch to *T-mode* and Send data to r.
end if
Switch back to *R-mode*.
/* r receiving data from s */
if Preamble received from s **then**
    Wait for data from s after the preamble.
end if
Go to sleep mode.

Another challenge is a potential collision between a preamble and a probing packet. Consider a scenario consisting of three nodes denoted by Sender, Receiver_1 and Receiver_2 (See Figure 3). If Receiver_2's probing packet is sent out right after the transmission of Sender's preamble, Receiver_1 will not be able to detect the preamble during the short CCA period due to the collision of the preamble packet from Sender and probing packet from Receiver_2. To address this issue, the *CCA-Extension* is employed: when a collision occurs, a receiver performs another CCA check, increasing the chance of receiving the next preamble.

IV. Performance Evaluation

The PRR and packet transmission delay of Asym-MAC are measured and compared against A-MAC [9]. For completeness, performance measurements for a transmitter-initiated MAC protocol, i.e., BoX-MAC-1 [7], are also provided, which can be considered as an "upper bound performance" as a transmitter-initiated MAC protocol is not affected by the asymmetry of the link (i.e., from receiver to sender) because it does not rely on a probing packet. However, it should be noted that a sender-initiated MAC protocol suffers from low energy efficiency and causes the hidden terminal problem as

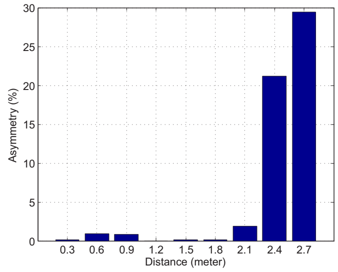

Fig. 4. Asymmetry for different distances.

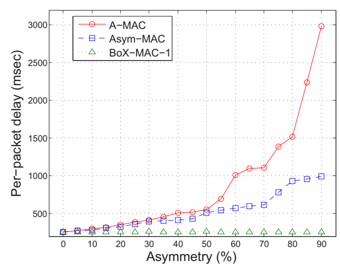

Fig. 5. Per-packet delay per asymmetry.

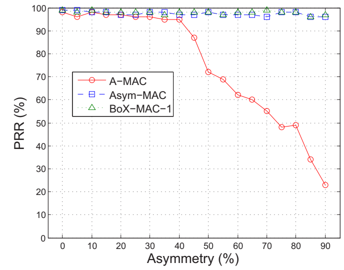

Fig. 6. PRR per asymmetry (500msec).

well as low channel utilization [8]. For this experiment, a pair of TelosB nodes running TinyOS 2.1.0 is deployed in an office area. One node transmits a packet of 15 bytes every second – a total of 1,000 packets – to the other node. To investigate the effect of collisions, the number of receivers within one-hop of the sender is increased up to 10 nodes.

Previous work [12] notes that there are various factors causing asymmetric links like hardware, transmit power, interference, and distance between nodes. Among the factors, Sang *et al.* [10] considered the transmit power and distance as major factors. Thus, to characterize the asymmetry of the link between the two nodes, the PRR difference between the up and down links (i.e., the asymmetry) is measured by varying the distance between the two nodes with TX power of -25dBm. Figure 4 illustrates that the PRR difference is small when the two nodes are close enough to each other, while the PRR difference significantly increases when the distance becomes greater than 2.1 meters, showing the asymmetry of the link.

A challenge is that the asymmetry of the link changes over time due to the changes in the interference level around the experimental environment (e.g., WiFi signals in the building); thus, it is difficult to reproduce the asymmetry for the same distance for different runs of experiments unless a complete control over the WiFi signals in the building is given. As a result, the degree of asymmetry is controlled by artificially dropping packets at the MAC layer while using the maximum transmit power (0dBm) with the minimum distance (less than 5cm) between the two nodes – it is experimentally confirmed that with this optimal setting, nodes constantly maintained PRR greater than 97% in both directions of the link. Specifically, the packet drop ratio of the up-link is changed from 10% to 100%, creating controllable asymmetry from 0% to 90%. Now given the control over the asymmetry of the link, the measurements of PRR and delay of Asym-MAC, A-MAC and BoX-MAC-1 by varying the asymmetry can be performed.

A. Per-packet Delay

Per-packet transmission delay is defined as a period of time between loading a data packet in a transmit buffer and successfully sending it to the intended receiver. For this experiment, an average of per-packet delay for 1,000 packet transmissions is measured; the probing interval is set to 500msec (i.e., the receiver sends a probing packet every 500msec); and τ is set to 1. Figure 5 depicts per-packet delay as a function of the asymmetry of the link. It is observed that per-packet delay increases as the asymmetry increases. This is mainly because the sender misses more probing packets when the asymmetry is high; when the transmitter misses a probing packet, it needs to wait for the next probing packet, thereby increasing per-packet delay. Asym-MAC has delay no longer than 1sec because it changes its mode to *T-Mode* when it consecutively fails to receive a probing packet twice (i.e., $\tau = 1$). In contrast, A-MAC has particularly high per-packet delay when the asymmetry is very high (i.e., over 50%), because the higher the asymmetry, more likely to have consecutive failures of receiving a probing packet. Overall, in the given experimental setup, Asym-MAC successfully reduces the per-packet delay of A-MAC by up to 66.7%.

B. Packet Reception Ratio (PRR)

The PRR of Asym-MAC and A-MAC is measured. Figure 6 shows the results (probing interval of 500msec). As shown, PRR decreases for both Asym-MAC and A-MAC as the asymmetry increases. Since a data packet is sent per second, missing a probing packet at least two consecutive times leads to a packet drop, thereby degrading PRR, assuming no buffering scheme is used. Note that for small asymmetry, both A-MAC and Asym-MAC perform relatively well; however, PRR of A-MAC starts to decrease significantly at high asymmetry. The reason is that it is more likely for the transmitter to miss a probing packet two consecutive times when the asymmetry is very high. In contrast, since Asym-MAC changes its mode to *T-mode* and sends the packet directly through the down-link when it detects a missing probing packet more than τ times, Asym-MAC has higher PRR than A-MAC. The same experiments with the probing interval of 1sec are performed, i.e., even a single miss of a probing packet potentially leads to decreased PRR. Figure 7 proves this by showing that PRR for both Asym-MAC and A-MAC degrades even for very small asymmetry. However, it is worth to note that PRR of Asym-MAC is still higher than A-MAC, because although Asym-MAC drops a packet for the first miss of a probing packet, it starts to combat the loss starting from the second loss of the probing packet because $\tau = 1$.

C. Energy Consumption

The average current drawn by both receiver and transmitter is measured for the three protocols. For this experiment, the probing interval is set to 500msec, asymmetry to 50%, and tx power to 0dBm. Figure 8 shows that compared with A-MAC, the receiver of Asym-MAC consumes more energy,

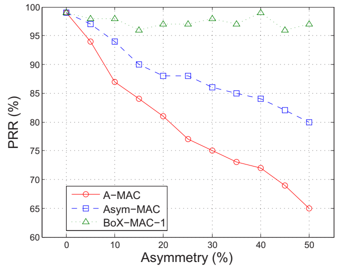

Fig. 7. PRR per asymmetry (1sec).

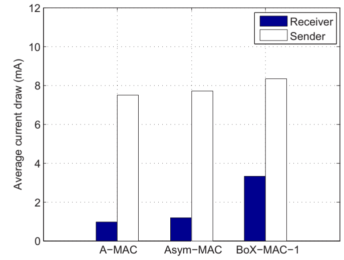

Fig. 8. Average current consumption.

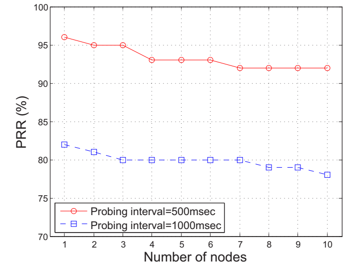

Fig. 9. PRR per number of nodes.

which is the cost for Asym-MAC's CCA checks. Similarly, the transmitter of Asym-MAC consumes more energy than A-MAC; however, interestingly, the difference is not large with an increase of only 2.67%, implying that the transmitter of A-MAC consumes energy unnecessarily waiting for the probing packet especially when the link is asymmetric, although Asym-MAC consumes energy by changing the mode and transmitting the preambles. Compared with the receiver-initiated protocols, the graph shows that BoX-MAC-1 has higher energy consumption, which coincides with the reported results [8].

D. Collisions

The different numbers ($1 \sim 10$) of receivers are deployed within the communication range of the sender to see the impact of collisions. All nodes periodically broadcast a probing packet, potentially causing a collision at the sender. The asymmetry of the link is set to 50%, and PRR is measured by varying the number of receivers. As Figure 9 shows, PRR decreases when the number of nodes is increased. The results indicate that although collisions between a probing packet and a preamble are avoided at each CCA check using the CCA-Extension scheme, a data packet may still collide with a probing packet, which is a known drawback of receiver-initiated MAC protocols [9]. However, it is worth to note that PRR may improve if the probe times are carefully designed using a protocol like [13].

E. Storage Overhead

One concern for Asym-MAC is the storage overhead compared with A-MAC, because Asym-MAC has more functionalities. Thus, RAM and ROM sizes required for the implementation of Asym-MAC and A-MAC on a TelosB mote are measured. A simple RadioCountToLeds application [14] is implemented on top of the two MAC protocols running on TinyOS 2.1.0. It is found that Asym-MAC requires 16.3% and 32.9% more space for ROM and RAM respectively. However, considering the small RAM and ROM sizes of A-MAC, we believe that the required space for Asym-MAC is acceptable.

V. CONCLUSION

Asymmetric links frequently observed in real-world deployment of WSNs degrade the performance of existing receiver-initiated MAC protocols. In this paper, a hybrid MAC protocol (Asym-MAC) designed for WSNs with highly asymmetric links is proposed. Future work includes the investigation of optimal τ for various parameters like probing interval.

REFERENCES

[1] R. R. Rajkumar, I. Lee, L. Sha, and J. Stankovic, "Cyber-physical systems: the next computing revolution," in *Proc. 2010 DAC.*

[2] V. Shnayder, M. Hempstead, B.-R. Chen, G. W. Allen, and M. Welsh, "Simulating the power consumption of large-scale sensor network applications," in *Proc. 2004 Sensys.*

[3] Crossbow Inc, "Telosb datasheet." Available: http://www.willow.co.uk/TelosB_Datasheet.pdf

[4] Y. Peng, Z. Li, W. Zhang, and D. Qiao, "LB-MAC: a lifetime-balanced MAC protocol for sensor networks," in *Proc. 2012 WASA.*

[5] X. Wang, X. Wang, G. Xing, and Y. Yao, "Dynamic duty cycle control for end-to-end delay guarantees in wireless sensor networks," in *Proc. 2010 IWQoS.*

[6] 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. 2006 Sensys.*

[7] D. Moss and P. Levis, "BoX-MACs: exploiting physical and link layer boundaries in low-power networking," in Technical Report SING-08-00, Stanford University, 2010.

[8] Y. Sun, O. Gurewitz, and D. B. Johnson, "RI-MAC: a receiver-initiated asynchronous duty cycle MAC protocol for dynamic traffic loads in wireless sensor networks," in *Proc. 2008 Sensys.*

[9] P. Dutta, S. Dawson-Haggerty, Y. Chen, C.-J. M. Liang, and A. Terzis, "Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless," in *Proc. 2010 Sensys.*

[10] L. Sang, A. Arora, and H. Zhang, "On link asymmetry and one-way estimation in wireless sensor networks," *ACM Trans. Sensor Networks*, vol. 6, no. 2, pp. 1–25, 2010.

[11] E. Ertin, A. Arora, R. Ramnath, M. Nesterenko, V. Naik, S. Bapat, V. Kulathumani, M. Sridharan, H. Zhang, and H. Cao, "Kansei: a testbed for sensing at scale," in *Proc. 2006 IPSN.*

[12] G. Wang, D. Turgut, L. Blni, Y. Ji, and D. C. Marinescu, "A MAC layer protocol for wireless networks with asymmetric links," *Ad Hoc Networks*, vol. 6, no. 3, pp. 424–440, 2008.

[13] J. Degesys, I. Rose, A. Patel, and R. Nagpal, "Desync: self-organizing desynchronization and TDMA on wireless sensor networks," in *Proc. 2007 IPSN.*

[14] RadioCountToLeds-TinyOS. Available: http://www.tinyos.net/tinyos-2.x/apps/RadioCountToLeds/