Telecommun Syst (2013) 52:2387–2395
DOI 10.1007/s11235-011-9551-z

Medium reservation based sensor MAC protocol for low latency and high energy efficiency

Pranesh Sthapit · Jae-Young Pyun

Published online: 3 August 2011

Abstract In this paper, we present a new MAC protocol satisfying with both high energy efficiency and low transmission latency at the same time over wireless sensor network, named as medium reservation preamble based MAC (MRPM). Unlike other synchronized duty cycle MACs, MRPM does not have separate time frames for SYNC and data traffics. Both traffics are integrated in a short listen period. Also, the channel contention is excluded from listen period and transferred to new period called contention period. The contention period precedes the listen period, and only transmitters wake up in this contention period and contend for medium reservation, whereas non-transmitters bypass it. These approaches enable MRPM to achieve adaptive duty cycle and quite short listen period. Moreover, MRPM uses carrier sensing information for advanced adaptive listening which makes packets to travel multiple hops away in a single sleep/listen cycle. The simulation results verify that MRPM has features of high energy efficiency and low latency.

Keywords Energy efficiency · Low latency · Contention window · MAC · WSN

1 Introduction

A wireless sensor network (WSN) is a network of self-organizing low-powered devices having sensing and communication capabilities. These small and inexpensive devices are battery powered, thus sensor nodes must be energy efficient [1–4]. Especially, minimizing power consumption is a primary goal in sensor MAC protocol design.

The radio of a typical sensor device has four modes of operation: transmitting, receiving, listening, and sleeping. Ideally, nodes should be sleeping always to minimize energy consumption but this would mean no communication. Thus, to conserve power and prolong their lifetime, the MAC protocols of WSN save energy by introducing the concept of duty cycling in which node periodically alternates between listen and sleep. Introduction of duty cycling motivates the use of synchronization between neighboring nodes. MAC protocols using synchronization approaches are S-MAC, T-MAC, DSMAC, and TEEM [1–4]. These protocols locally manage synchronization by broadcasting periodic SYNC packet to their neighbors. These protocols have separate time frames for synchronization and data on their listen period.

S-MAC is a contention-based random access protocol with a fixed sleep/listen cycle [1]. It uses a coordinated sleeping mechanism, similar to the power saving mechanism of IEEE 802.11. A time frame in S-MAC is divided into two parts: a listen period and a sleep period. The listen period is further divided into SYNC, RTS, and CTS periods as shown in Fig. 1. Since nodes can only communicate in listen period, neighboring nodes must be synchronized together. Therefore, each S-MAC node periodically exchanges its schedule by broadcasting a SYNC packet to its neighbors at SYNC period. In S-MAC, RTS and CTS control packets are used for data communication similar to IEEE 802.11. RTS and CTS packets are transmitted at their respective periods in the listen period. The successful exchange of RTS/CTS packets between two nodes implies that they should stay awake in the whole sleep period for the completion of their data com-

P. Sthapit · J.-Y. Pyun (✉)
Department of Information and Communication Engineering, Chosun University, Gwangju, Republic of Korea
e-mail: jypyun@chosun.ac.kr

P. Sthapit
e-mail: pranesh@chosun.kr

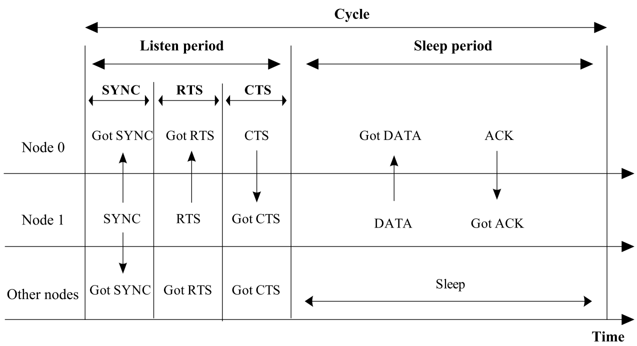


Fig. 1 Sleep/listen cycle of S-MAC

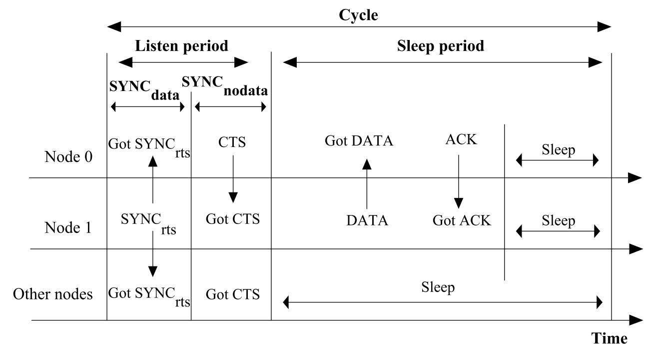


Fig. 2 Sleep/listen cycle of TEEM

munication. Again, all other nodes that are not involved in data communication can enter a sleep mode. Figure 1 shows the data communication between node 0 and node 1 in S-MAC. The nodes overhearing RTS or CTS wake up for a short time when the ongoing communication finishes. This adaptive listening can forward the data to 2 hops in one sleep/listen cycle, reducing the latency to some extent. S-MAC has a fixed long listen period. The problem is that, even when nodes have no data or SYNC packet to send, the nodes still have to be awake in listen time, draining their energies.

Unlike S-MAC, in TEEM [4], the listen period consists of only two parts, SYNC_{data} and SYNC_{nodata}, and the time interval of the listen period is also shorter compared to S-MAC as shown in Fig. 2. The SYNC_{data} contains data packets, whereas the SYNC_{nodata} contains SYNC packets. Both packets are used for synchronization. Each node will listen in SYNC_{data}, whether a node has data to transfer or not. Nodes having data will contend for medium in this period. If there is no communication in this period, then nodes having SYNC packet contend for medium in the SYNC_{nodata} period and the winner sends the SYNC packet. Instead of using RTS and SYNC separately, TEEM combines the RTS packet with a SYNC packet and sends it in SYNC_{data} period. This combination is called SYNC_{rts}. Since the data traffic is transferred in the very first period of listen time, nodes that are not involved in current communication can go to sleep immediately. Furthermore, nodes that are involved in communication can go to sleep as soon as communication between them is finished as depicted in Fig. 2. These procedures make TEEM's listen period adaptive and much more energy efficient than S-MAC.

The basic working principle of LE-MAC [5] is same as that of S-MAC. The difference is the approach of using carrier sensing signals for reducing sleep delay in multi-hop transmission. The nodes that detect the transmission but unable to interpret, wake up at appropriate time in their sleep period such that they are likely be the next candidate nodes in the current multi-hop data transmission. This technique allows nodes to forward data to few more hops away than SMAC at the cost of some energy required for adaptive listening [1, 5].

Usually, synchronized duty cycle MAC protocols have separate time frames for synchronization and data on their listen period, which contribute to the long listen period. The shorter the listen period is, the longer the network life is. Furthermore, these protocols use CSMA/CA based random access method for channel access. Therefore, the backoff duration (contention period) is also included in the listen period (in both SYNC and DATA periods). Due to the backoff duration of listen period, large energy consumption can be inevitable. A number of previous approaches have been proposed to reduce energy consumption as well as latency in synchronized duty cycle MAC protocols for WSN, but most of them are promising to be effective in one performance metric. In this paper, we present a new MAC protocol, called MRPM [6, 7] having both high energy efficiency and low latency. In order to achieve these properties, MRPM excludes the contention from the listen period and transfers to a new period called contention period. Exclusion of contention from listen period makes listen period very short. Moreover, the listen period is further shorten by integrating SYNC and data traffics into a single short time frame during the listen period. These techniques made MRPM possible to achieve listen period of very short length. Furthermore, MRPM achieves low latency by continuously transmitting data multiple hops away in one listen/sleep cycle by using its advanced adaptive listening (AAL). In this paper, we present refined MRPM using AAL with more simulation results and more detail explanations.

The rest of this paper is organized as follows. Section 2 presents the motivation of our work. Section 3 shows the main features and algorithms for the proposed MRPM protocol. Section 4 demonstrates the energy efficiency and low data delivery latency achieved by MRPM through simulation results. Finally, we conclude the paper in Sect. 5.

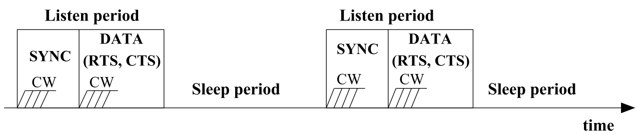


Fig. 3 Inclusion of contention windows in SYNC and DATA periods of typical synchronized duty cycle MAC

2 Motivation of our work

The S-MAC, TEEM, and other synchronized duty cycle MAC protocols periodically send SYNC packets for synchronization of listen period among the neighboring nodes. Thus, to deal with SYNC and data traffics, these protocols have separate time frames on their listen period, which make the listen period quite long. The shorter the listen period is, the longer the network life is. Furthermore, these protocols use CSMA/CA based random access method for channel access. Therefore, the backoff duration (contention duration) is also included in the listen period (in both SYNC and DATA periods) as shown in Fig. 3, which further lengthens the listen period. Due to the backoff duration of listen period, large energy consumption can be inevitable. In a typical synchronized duty cycle MAC such as in S-MAC, the length of the listen period is given by

$$L_p = t_{sync} + t_{data} \quad (1)$$

where t_{sync} and t_{data} represent the time duration of SYNC and data period respectively.

$$t_{sync} = \sigma \times CW_{sync}^{max} + Tx_{sync} \quad (2)$$

$$t_{data} = \sigma \times CW_{data}^{max} + Tx_{data} \quad (3)$$

Thus, the duration of SYNC and data periods are given by (2) and (3), where σ represents slot time, CW_{sync}^{max} and CW_{data}^{max} represent the maximum contention windows (CWs) for SYNC and data. Tx_{sync} represents time required to transmit SYNC packet and Tx_{data} represents time required to transmit RTS and CTS packets. Since the lengths of SYNC, RTS, and CTS are of only some bytes, the most dominant parameter that occupies the most of the time in the listen period is contention window. Listen period can be made much shorter if contention windows are excluded from it. Therefore, the exclusion of contention time from the listen period makes nodes listen the channel during much shorter time in every cycle regardless of traffic. Since contention is done only by the transmitters, non-transmitters can go to sleep during contending periods. Therefore, introduction of contention period also makes duty cycle adaptive because contention period is only used by nodes when they have data and in other time they are bypassed. The above-mentioned findings motivated to have a separate period (contention period) for transmitters contending for medium reservation. Taking above points into consideration, we proposed MRPM, which is expected to be efficient than these conventional synchronized duty cycle MAC protocols. With the energy efficiency feature, MRPM also has low data delivery latency by adopting the physical carrier sensing and adjusting nodes duty cycle dynamically [5]. The proposed MRPM is suitable candidate MAC for delay sensitive WSN applications.

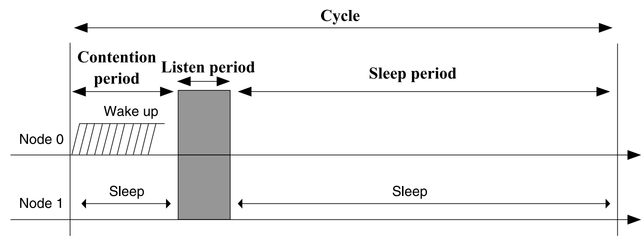


Fig. 4 Sleep/listen cycle in MRPM

3 Proposed MRPM design

MRPM is a synchronized duty cycle MAC protocol, where each cycle is divided into three periods, i.e., contention, listen, and sleep as shown in Fig. 4. In the contention period, nodes contend for the medium. Only the transmitters wake up in contention period, whereas all neighbors wake up at the listen period. Nodes with SYNC and data traffics compete for channel during the contention period, and the winner gets the chance to use the listen period. The basic concept of MRPM is to make nodes listen for very short time. If the node hears transmission within this listen period, it remains awake, otherwise goes to sleep.

3.1 Contention period

MRPM excludes the contention from listen period and moves it to a new period called contention period as shown in Fig. 4. The length of contention period, C_p is given by

$$C_p = t_{difs} + \sigma \times CW_{sync}^{max} + \sigma \times CW_{data}^{max} + Tx_{mrp} + t_{guard} \quad (4)$$

where t_{difs} is the distributed inter-frame space and t_{guard} is the guard time for preventing small synchronization errors. Tx_{mrp} represents the time required to transmit medium reservation preamble (MRP) packet. MRP is used to notify a certain node is occupying the channel and will be explained in detail in next section. MRPM takes the contention windows of SYNC and data periods and moves it to contention period. This transfer of contention to new period drastically reduces the duration of listen period.

In Fig. 4, node 0 has data to transfer, whereas node 1 does not have data. As shown in the figure, only the nodes

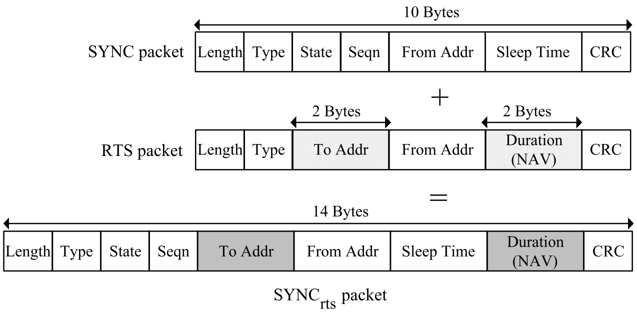


Fig. 5 Packet structure of SYNC_{rts}

that have packets to transfer wake up at contention period. Nodes that have nothing to transmit are still sleeping at this period. This nature of MRPM makes its duty cycle adaptive, and makes it highly energy efficient.

As mentioned earlier, MRPM integrates SYNC and data traffics into a single time frame in a short listen period. Therefore, during the contention period, nodes with SYNC or data traffics contend for channel access using CSMA/CA protocol as in IEEE 802.11, and the winner uses the listen period. To give priority to the data traffic, the contention windows for data and SYNC traffics are respectively assigned as shown in (5) and (6). Here, CW_{sync} and CW_{data} represent the contention windows for SYNC and data traffics respectively. Similarly, random [0–CW] generates random number between 0 and CW, and $SYCRO_{period}$ is the synchronization period which is used for sending SYNC packet. As we can see from the equations, if both traffics compete, data packets are mostly the winner. To prevent the nodes with SYNC packet from starvation, if the nodes with SYNC packets are unable to get medium even after trying for more than two synchronization period, the right hand side of (5) is used to assign the CW_{sync}. This will eventually give node a chance to transmit its SYNC packet.

$$CW_{data} = random[0 - CW_{data}^{max}] \tag{5}$$

$$CW_{sync} = \begin{cases} random[0 - CW_{sync}^{max}] + CW_{data}^{max} \\ \quad \text{if, } N_{Fail}^{access} < 2 \times SYCRO_{period} \\ random[0 - CW_{data}^{max}], \quad \text{else} \end{cases} \tag{6}$$

To reduce the number of SYNC traffic, SYNC information is also transferred during data traffic. In MRPM, SYNC and RTS packets are combined and newly generated packet, called SYNC_{rts} is used in place of RTS as in TEEM [4]. Figure 5 shows the packet structure of SYNC_{rts}. With this new packet, nodes do not need to send SYNC packets when they also have data packets. This single packet can be used for synchronization as well as RTS packet. This method obviously reduces the amount of SYNC traffic, which in turn reduces the channel contention, and also saves energy.

3.2 Medium reservation preamble

In MRPM, nodes have two wake up points: transmitters wake up early at contention period, whereas other nodes wake up later at listen period as shown in Fig. 4. During the contention period, nodes with SYNC and data traffics contend for channel access by using CSMA/CA protocol with their respective CW values (CW_{data} or CW_{sync}). Whichever node backoff first, sends a short packet called MRP. MRP is a regular bit pattern of some bytes and does not contain any useful information. Its sole purpose is to make other nodes realize that a certain node has gained the channel. Nodes do not need to decode MRP, thus just realizing transmission in contention period is enough for nodes to give up contention. Since carrier sensing can be done from multiple-hops away, hidden node problem can be avoided here.

Since the sender is already decided at contention period, transmitting and receiving nodes can immediately communicate data each other at listen period. But, there may be chances that two nodes employ same CWs during the contention period leading to collision of MRP packets. Since data are comparatively larger than control packets, collision of MRP may leads to waste of energy as well as time. Thus, our protocol employs RTS/CTS mechanism as in S-MAC to resolve the collision. If there is collision in MRP, there will be also collision in RTS. But there will be no CTS because of collision in RTS. This will make nodes to backoff for sending new RTS. Eventually, there will be one transmitter. Furthermore, the employment of RTS/CTS enables MRPM for adaptive listening.

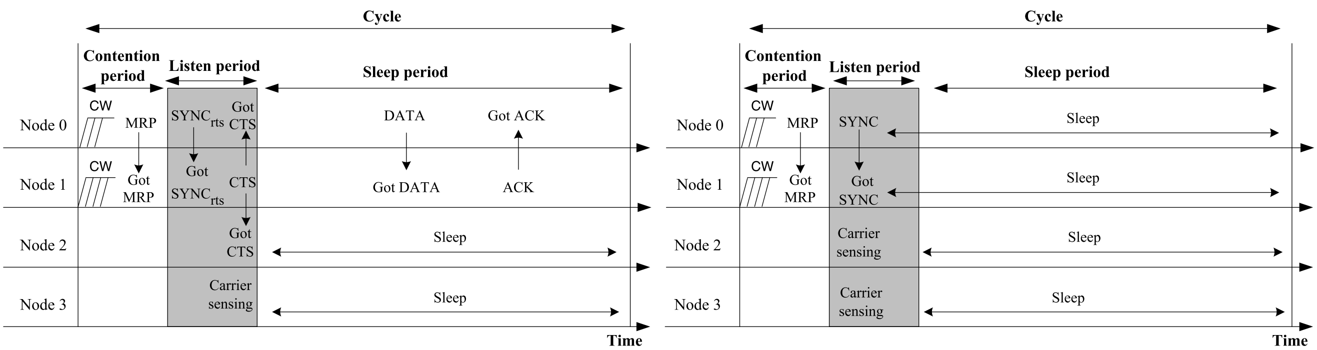
3.3 Short listen period

MRPM is unique in the way that the sender is resolved beforehand the listen period. Furthermore, it does not have separate time frame for SYNC and data traffics. Thus, nodes wake up for short duration during listen period and both SYNC and data traffics are handled in this short listen period. The listen period is shown in Fig. 4, which is represented by the shaded region. The length of listen period (L_p) should be at least the duration taken to exchange SYNC_{rts} and CTS packets completely and is given by

$$L_p = Tx_{rts} + t_{sifs} + Tx_{cts} + t_{guard} \tag{7}$$

where t_{sifs} is short inter-frame space, Tx_{rts} and Tx_{cts} are the transmission time for SYNC_{rts} and CTS packets respectively. The listen period duration is required to make sure that the nodes, that are located within the carrier sensing range of the node originating CTS packet, do not miss the carrier sensing by early sleeping.

Nodes only remain awake if they hear transmission within this listen period, otherwise they go to sleep after the end of listen period. In most of the WSN applications,



(a) When nodes have data traffic (b) When nodes have no data traffic

Fig. 6 Basic mechanism of the proposed MRPM

nodes in a neighborhood do not have packets to transmit. Thus, most of the time, the nodes wake up only in listen period for a short time and go to sleep immediately. This adaptive nature of MRPM makes much more energy efficient than conventional MAC protocols are.

If there is no collision during contention period, there is always one transmitter ready to transmit in the listen period. MRPM uses RTS-CTS-DATA-ACK frame sequence for transferring data as in IEEE 802.11. In the listen period, the sender transmits after waiting for small guard time to prevent from synchronization error. The overall protocol can be seen with an example. In Fig. 6(a), node 0 and 1 want to transmit data. Thus, they wake up early in the contention period and contend for the medium. Here, node 0 finishes backoff first and transmits the MRP. Node 1 hears MRP and gives up contention. Then, the nodes enter into listen period. At this time, all neighbors wake up. Node 0 transmits SYNC$_{rts}$ packet. Upon receiving SYNC$_{rts}$, node 1 acknowledges with CTS packet. The successful exchange of SYNC$_{rts}$/CTS between two nodes implies that these two nodes should stay awake until the completion of data communication. All other nodes that are not involved in data communication can go to sleep. After receiving the data from node 0, node 1 acknowledges with ACK as shown in the figure. In this way data are transferred between nodes. As similar to adaptive listening in S-MAC [1], nodes in MRPM overhearing SYNC$_{rts}$/CTS perform adaptive listening. Node 3 can't decode CTS but can sense it because it is within the carrier sensing range of node 1. Here, node 3 performs advanced adaptive listening. Node 3 goes to sleep with completion of its listen period.

For synchronization, nodes periodically send their SYNC packet to their neighbors. As mentioned earlier, synchronization is also done with SYNC$_{rts}$ packet. Figure 6(b) shows the exchanging of SYNC packets, which is exactly same as the case of transmitting SYNC$_{rts}$ as explained above. Since SYNC packet is normally received when there is no queued data packet in the neighborhood, nodes immediately go to sleep after exchanging SYNC packet as shown in Fig. 6(b).

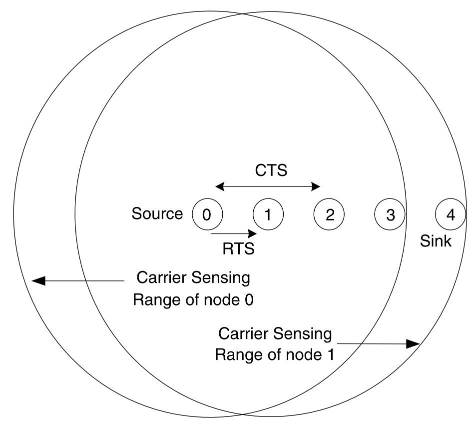


Fig. 7 Carrier sensing range of node 0 and node 1

3.4 Advanced adaptive listening

For the adaptive listening [1], the nodes that overhear SYNC$_{rts}$/CTS packets schedule themselves to wake up in their sleep period after completion of current transmission, such that the data packets can be received in the same cycle. With this technique, packets are transmitted to 2 hops away in single sleep/listen cycle. On the other hand, the adaptive listening in MRPM is not limited to these 2 hops. The carrier sensing ability of nodes is also taken into account for advanced adaptive listening, which increases the range of adaptive listening [5]. To differentiate with adaptive listening, we call this new adaptive listening as advanced adaptive listening (AAL). The nodes that are unable to decode the SYNC$_{rts}$/CTS, are assumed to be at least two hops away from the sender or receiver [1]. Let us show this whole process with an example. In Fig. 7, source node 0 wants to transfer data to sink node 4 via the intermediate nodes 1, 2, and 3. The transmission range is of a single hop. The two circles here represent the carrier sensing range of node 0

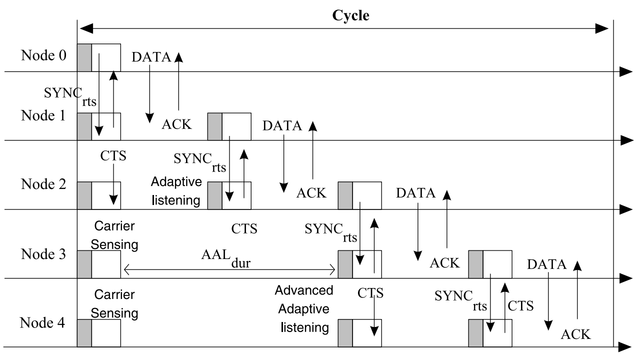


Fig. 8 Advanced adaptive listening in MRPM

and node 1 respectively. The working process of adaptive listening is shown in Fig. 8. The gray rectangular box in the figure represents contention period, whereas white rectangle represents the listen period. Initially, node 0 transfers the SYNC$_{rts}$ packet and node 1 reply with CTS packet. This CTS packet is overheard by the node 2 which schedules itself for adaptive listening. Since node 3 and 4 are within the carrier sensing range of node 1, they both can sense the CTS. Since a packet is transmitted up to 2 hops by adaptive listening mechanism, nodes 3 and 4 schedule themselves for AAL after AAL$_{dur}$ duration. AAL$_{dur}$ is a duration required by a fixed length packet to reach 2 hops away, and is given by

$$AAL_{dur} = 2(Tx_{ack} + Tx_{data}) + C_p + L_p \tag{8}$$

where Tx_{ack} and Tx_{data} represent the durations taken for the transmission of ACK and data packets, and C_p and L_p represent the contention period and listen period respectively. The SIFS and guard time are assumed to be embedded in the packets where necessary in above representation. Data packets are of fixed length. The derivation of equation (8) can be easily explained through Fig. 8. To reach the data from node 0 to node 2 (2 hops), it requires 2 data packets, 2 ACK packets, 1 contention period, and 1 listen period as shown in the figure. Note that listen period is equal to duration required to exchange SYNC$_{rts}$ and CTS. As seen in the Fig. 8, node 3 and 4 perform the AAL after AAL$_{dur}$. Thus, they wake up at same time. Node 4 can overhear the CTS transferred from node 3 to node 2. From CTS, node 4 acknowledges when it should perform adaptive listening. In this way, multiple hops can be achieved in a single sleep/listen cycle with this new approach. With this new enhanced adaptive listening, data can travel to multiple hops away till where the carrier sensing of SYNC$_{rts}$ or CTS can be realized.

The AAL may lead to inefficiency in the situations when there are only SYNC packets and no data packets. All the nodes sensing SYNC packets perform adaptive listening unnecessarily wasting energy. Generally, there are less data transmissions in WSN applications. Thus, in order to prevent this inefficiency of adaptive listening in low traffic load situation, nodes logically divide their listen period into SYNC$_{rts}$ period and CTS period. Nodes do not perform adaptive listening if they sense transmission in SYNC$_{rts}$ period. They only perform adaptive listening, if they sense transmission in CTS period. That means nodes do not perform adaptive listening if they sense either SYNC packet or SYNC$_{rts}$. We believe that nodes in routing path definitely hear or sense the CTS packet if the node would be the next hop after the completion of current transmission. This management considerably removes negative effect on the new advanced adaptive listening. Doubtlessly, there may be nodes sensing CTS, draining some network energy proportional to node density. But, since our proposed MRPM has quite short listen period, the amount drained by adaptive listening will not account much.

Table 1 Parameters for NS-2 simulation

Channel bandwidth	20 kbps
Reception power	14 mW
Transmission power	36 mW
Idle power	14 mW
Sleep power	15 µW
Transition power	28 mW
Transition time	2 ms
Slot time	1 ms

4 Performance evaluation

We implemented MRPM on the ns-2 network simulator [8]. For the performance evaluation, we compared MRPM with S-MAC and TEEM protocols. In our simulation model, the transmission and the carrier sensing (CS) ranges were of 250 m and 550 m respectively. For all the protocols, the simulated nodes were configured by using the parameters listed in Table 1. The duty cycles of S-MAC and TEEM protocols were set to 10%. The cycle period of MRPM was set same as that of S-MAC. The duty cycle of MRPM was just 1.93% for the same cycle period of S-MAC. The size of MRP used in the simulations is of 10 bytes. Various sets of simulations were performed to test the energy efficiency and end-to-end latency of MRPM. In all the simulations, nodes used NOAH static ad-hoc routing protocol [9]. Each sensor node in the experimental network was assumed to have an initial energy level of 10 Joules. For the traffic model, an UDP/CBR traffic model was used. The source node generates total of 50 messages of 50 bytes. Each message was transferred to sink and simulation ended with the transfer of the last packet. The intermediate nodes generated no data packets and only forwarded the data packets to next hop.

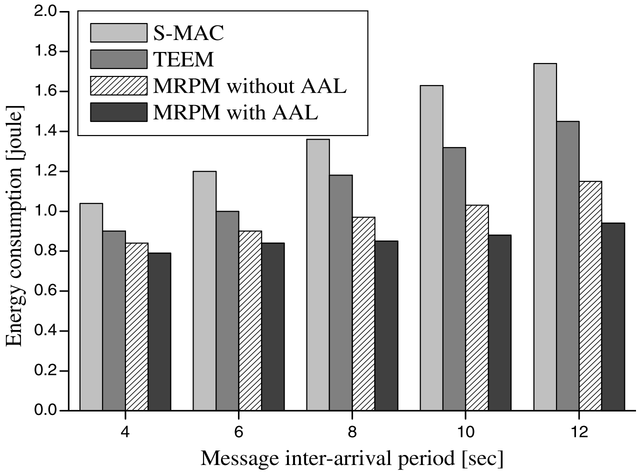


Fig. 9 Average energy consumption in linear topology

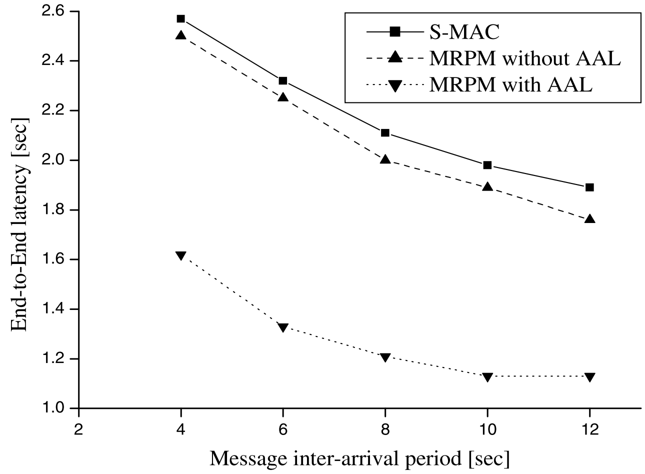


Fig. 10 Latency experienced in linear topology

In our first set of experiment, we took a linear topology of 4 nodes (3 hops) with the first node as source and the last node acting as sink. The nodes were at the distance of 250 m from each other. This set of experiment analyzed the performance of MRPM for the nodes involved in routing under varying traffic load. Here, the message inter-arrival period was varied from 4 to 12 secs. The average energy consumed by nodes involved in routing for all three protocols are shown in Fig. 9. We compared the energy efficiency of MRPM with and without AAL against S-MAC and TEEM. The results show that MRPM has highest energy efficiency in either way. We found that inclusion of AAL in MRPM makes it more energy efficient because in MRPM with AAL, packets travels more hops in a single cycle largely reducing the overall time needed to pass the fixed amount of data through the network. The experimental results show that MRPM with AAL achieved energy efficiency 45% and 35% higher than S-MAC and TEEM respectively at message inter-arrival period of 12 secs. The average latency recorded during the simulation for each message inter-arrival period is shown in Fig. 10. Since the TEEM does not have adaptive listening, its latency is poor as compared to others, thus is not shown in the figure. As we have expected, the latency of MRPM without AAL is less than that of S-MAC. This is because, firstly, in MRPM, nodes do not have to waste time in SYNC period. Secondly, nodes can transmit immediately as soon as they are in listen period. Finally, in MRPM, $SYNC_{rts}$ packet is used which greatly reduces the network congestion and latency due to the collision of SYNC packets. MRPM with AAL has least latency since MRPM with AAL can transfer packets more hops in single cycle. For the current configuration of simulation, it took single cycle to reach data from source to sink for MRPM, whereas S-MAC took at least 2 cycles. Moreover, at the message inter-arrival period of 4 secs, the latency of MRPM with AAL is 1.6 times better than that of S-MAC.

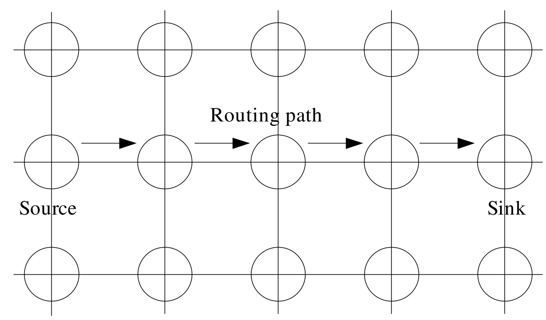


Fig. 11 Grid topology

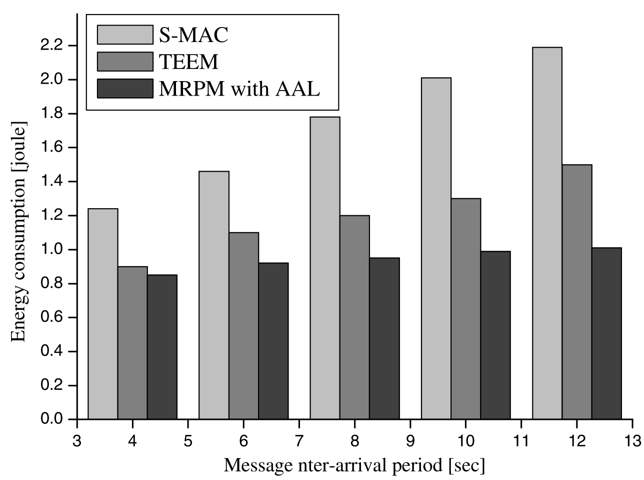


Fig. 12 Average energy consumption in grid topology

In our second set of experiment, we took more realistic grid topology of 15 nodes arranged in 3 rows with 5 nodes in each row as shown in Fig. 11. The nodes were at the distance of 250 m from each other. The first and the last nodes of the second row were source and sink. The other nodes, between the source and the sink nodes in the second row,

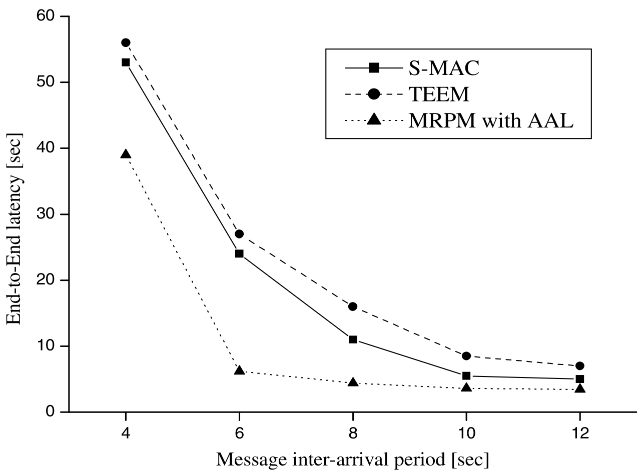


Fig. 13 Latency experienced in grid topology

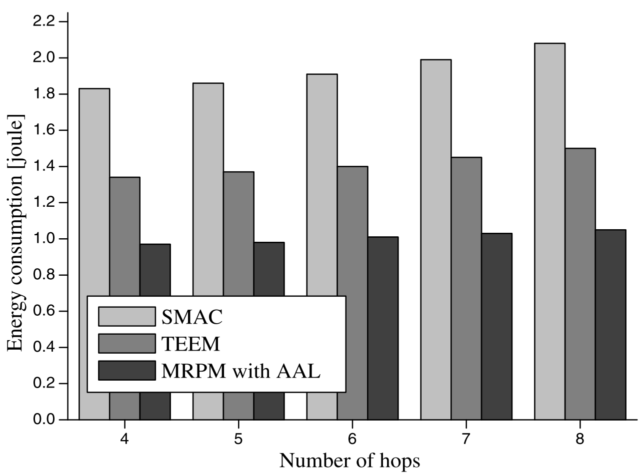


Fig. 14 Average energy consumption under variable hops

acted as intermediate relay nodes and forwarded data to the sink. Here also, the message inter-arrival period was varied from 4 to 12 secs. As mentioned earlier, MRPM has shorter listen period compared to other two protocols. Also, only the nodes with packets wake up in contention period making the duty cycle adaptive. Thus, the average energy consumption of MRPM is significantly less as compared to S-MAC and TEEM as shown in Fig. 12. Further, the energy consume is much less as the message inter-arrival time is getting larger in MRPM. The experimental results showed that MRPM achieved energy efficiency of 2.16 times and 1.5 times higher than S-MAC and TEEM respectively at message inter-arrival period of 12 secs. The average latency experienced by all the protocols is shown in Fig. 13. The graphs verify that MRPM has lower latency. Also, we observed the probability of network collision on all the protocols. Since, all the protocols are based on CSMA/CA, the probability of network collision was similar in all protocols.

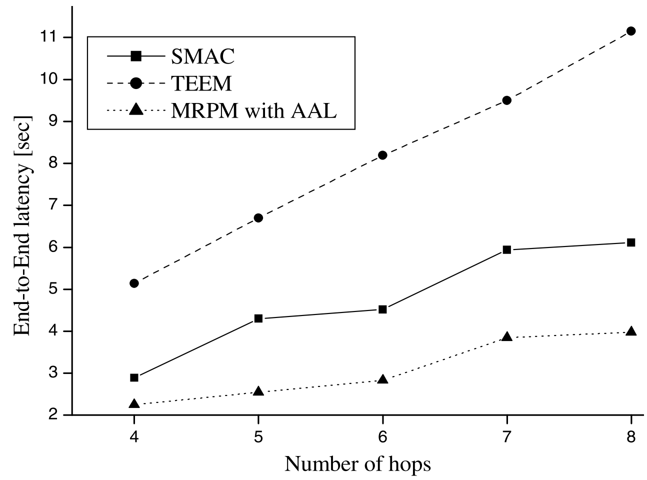


Fig. 15 Latency experienced under variable hops

For the final set of experiment, we took a linear topology where first node was source and last node was sink. Here, we fixed the message inter-arrival period to 10 secs, but we varied the numbers of intermediate nodes between the source and the sink. The distances between any two nodes were set to 250 m. For all the protocols, the cycle times were fixed at 1403 ms. This set of experiment focused on analyzing the latency under variable hops between the source and the sink. Figure 14 shows the average energies consumed by MRPM, TEEM, and S-MAC protocols. There is a noticeable change in energy consumption under varying number of hops for S-MAC and TEEM. Whereas, in case of MRPM, the energy consumption for transferring 50 packets on varying number of hops between source and sink is almost same. This is because, firstly the MRPM has short listen period. And secondly, in MRPM, packets move more hops in one cycle than in SMAC, which contribute to less energy consumption. The minimum latency recorded during the simulation for all the protocols in this set of experiment is shown in Fig. 15. We can see from the Fig. 15 that TEEM protocol has a linear nature of graph. This is because data in TEEM can travel only a single hop in a cycle. Whereas, because of the presence of adaptive listening in S-MAC, packets could travel 2 hops in a single cycle. The nature of the graphs in the figure also reveals the traveling of packets to 2 hops in a single cycle. However, in the case of MRPM, its latency is much less than that of S-MAC. Since MRPM also uses physical carrier sensing along with virtual carrier sensing, it achieves the delivery of packets into more hops than in S-MAC till where the transmission can be sensed. In the current set of experiment, data traveled one more hop than that of S-MAC in a single cycle. MRPM achieved the latency performance of 1.53 times higher than that of S-MAC when the source and sink was 8 hops away.

5 Conclusion

In this paper, we proposed a new energy efficient MAC protocol called MRPM, which is highly energy efficient, and also have low latency. In order to achieve these properties, MRPM excludes the contention from the listen period and moves to a new period called contention period. Exclusion of contention from listen period makes listen period quite shorter and also makes duty cycle adaptive. Moreover, the listen period is further shortened by integrating SYNC and data traffics into a single short time frame during the listen period. These techniques made MRPM possible to achieve listen period of quite short time with adaptive duty cycle. Furthermore, MRPM achieves low latency by continuously transmitting data multiple hops away in one listen/sleep cycle by using its advanced adaptive listening. Our simulation results demonstrated that our protocol is energy efficient and also has low latency that can be adapted for delay sensitive WSN applications.

References

1. Ye, W., Heidemann, J., & Estrin, D. (2004). Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Transactions on Networking* (pp. 493-506).
2. Dam, T. V., & Langendoen, K. (2003). An adaptive energy efficient MAC protocol for wireless sensor networks. In *Proc. of ACM SynSys '03* (pp. 171–180).
3. Lin, P., Qiao, C., & Wang, X. (2004). Medium access control with a dynamic duty cycle for sensor networks. *IEEE Wireless Communications and Networking Conference*, *3*, 1534–1539.
4. Suh, C., & Ko, Y. B. (2005). A traffic aware, energy efficient MAC protocol for wireless sensor networks. In *Proc. of the IEEE international symposium on circuits and systems (ISCAS'05)* (Vol. 3, pp. 2975–2978).
5. Suh, C., Shrestha, D. M., & Ko, Y. B. (2006). An energy-efficient MAC protocol for delay-sensitive wireless sensor networks. In *Lecture notes in computer science* (Vol. 4097, pp. 445–454).
6. Sthapit, P., Park, Y. T., & Pyun, J. Y. (2009). Medium reservation preamble based medium access control for wireless sensor network. In *Proc. of IEEE vehicular technology conference (VTC), fall* (Vol. 1, pp. 1–6).
7. Sthapit, P., Park, Y. T., & Pyun, J. Y. (2009). Medium reservation based MAC for delay-sensitive wireless sensor network. In *Proc. of IEEE international conference on computer and information technology (CIT)* (Vol. 2, pp. 122–127).
8. NS-2 website, http://www.isi.edu/nsnam/ns/.
9. The NO Ad-Hoc Routing Agent (NOAH) website, http://icapeople.epfl.ch/widmer/uwb/ns-2/noah/.

Pranesh Sthapit was born in Kathmandu, Nepal, on January 15, 1982. He received a B.I.T. degree from Purbanchal University, Nepal on 2005 and M.S. degree from Chosun University, Korea, in Information and Communication Engineering, in 2010. He is currently research assistant and Ph.D. student at Chosun University, Korea. He is currently a member of "Wireless and Mobile Communication System Lab". His current research focus is on the energy-efficient MAC protocols in wireless sensor networks.

Jae-Young Pyun (M'03) received a B.S. degree from Chosun University, an M.S. degree from Chonnam University, and a Ph.D. degree from Korea University, Korea, in Electronics Engineering, in 1997, 1999, and 2003, respectively. From 2003 to 2004, he was with Samsung Electronics, where he was involved in research and development of mobile phone communication systems. In 2004, he joined the Department of Information & Communication Engineering at Chosun University, where he is currently an associate professor. His research interests include Mobile QoS, IP QoS, Network protocols over WSN, and video communication.