Research Article

A Dynamic Traffic-Aware Duty Cycle Adjustment MAC Protocol for Energy Conserving in Wireless Sensor Networks

Tz-Heng Hsu,[1] Tai-Hoon Kim,[2] Chao-Chun Chen,[1, 3] and Jyun-Sian Wu[1]

[1] *Department of Computer Science and Information Engineering, Southern Taiwan University, Tainan, Taiwan*
[2] *GVSA and University of Tasmania, Tasmania, Australia*
[3] *Institute of Manufacturing Information and Systems, National Cheng Kung University, Tainan, Taiwan*

Correspondence should be addressed to Tai-Hoon Kim, taihoonn@paran.com

Wireless sensors are battery-limited sensing and computing devices. How to prolong the lifetime of wireless sensors becomes an important issue. In order to reduce the energy consumptions when nodes are in idle listening, duty-cycle-based MAC protocols are introduced to let node go into sleep mode periodically or aperiodically. The long duty cycle makes sensors increase the transmission throughput but consumes more energy. The short duty cycle makes sensors have low energy consumption rate but increases the transmission delay. In this paper, a dynamic traffic-aware MAC protocol for energy conserving in wireless sensor networks is proposed. The proposed MAC protocol can provide better data transmission rate when sensors are with high traffic loading. On the other hand, the proposed MAC protocol can save energy when sensors are with low traffic loading. Simulation results show that the proposed protocol has better data throughput than other duty-cycle-based MAC protocols, for example, S-MAC and U-MAC. We also developed a set of comprehensive experiments based on the well-known OMNET++ simulator and revealed that our proposed TA-MAC performs significantly outstanding than related schemes under various situations.

1. Introduction

A wireless sensor network (WSN) is composed of hundreds or thousands of autonomous and compact sensing devices, gathering and delivering information about events of interest to observers. In a wireless sensor network, the energy efficiency of communications among sensors is a major issue. Various energy conservation schemes for sensor networks have been proposed [1–5]. TDMA-based protocols, for example, LEACH [1] and BMA [2], provide contention-free mechanisms for sensors to avoid the competition of the radio channels. The major advantage of TDMA-based protocols is energy efficiency due to no energy wasted on collisions caused by channel contention [6–8]. However, classic TDMA protocols are hard against the unpredictable variations of sensor networks.

In traditional wireless sensor networks, sensors need to be in listening for sensing data transmission. If no sensing event happens, nodes are in idle for a long time. Idle listening wastes energy when a node is active, even there is no meaningful task on the radio channel. Traditional MAC protocols, such as 802.11, are unsuitable for sensor networks' data delivery. Idle listening in 802.11 consumes as much energy as it does when receiving data. In order to reduce the energy consumption when nodes are in idle listening, duty-cycle-based MAC protocols are introduced to let nodes go into sleep mode periodically or aperiodically [9–13].

S-MAC is a MAC protocol that can specify when nodes are awake and asleep within a frame [9]. In S-MAC, the period of a regular cycle consists of a listen and a sleep state. The duty cycle is defined as the ratio of the node in listen state compared with the period of a regular cycle. Within a virtual cluster, a node exchanges synchronization and schedule information with its neighbors to ensure that the node and its neighbors wake up at the same time. In S-MAC, a uniform duty cycle is assigned across the whole network. The uniform duty cycle assignment is not suitable for all sensors because not all sensors have the same duty or workloads. If the uniform duty cycle is set too long, it may cause energy wastage on nodes with low data traffic because the nodes are in idle listening. If the uniform duty cycle is set too short, it may increase transmission latency on nodes

with heavy data traffic because they do not have enough time to transmit all collected data [9]. T-MAC protocol enhances the design of S-MAC by listening to the radio channel for only a short time, and then nodes enter into sleep mode if no data is received during that time [10]. After synchronization, T-MAC shortens the awake period of a node if the radio channel is idle for a short time. If data is received, the node remains awake until the transmission is finished or the awake period ends. By using adaptive duty cycling, T-MAC can reduce energy usage for data transmission. However, in return for gaining energy, the costs are the low data throughput and the long transmission latency.

U-MAC protocol tunes duty cycle according to the traffic loads of sensor nodes. In wireless sensor networks, making nodes asleep for a long time to lower the energy consumption may result in higher transmission latency [11]. U-MAC tries to solve the problem by tuning duty cycle according to sensors' traffic loads. To reflect the traffic loads, U-MAC adopts a utilization function to calculate the load of each node. Upon synchronization, a node calculates its traffic utilization since the last synchronization time. The node adjusts its duty cycle according to the calculated utilization and then broadcasts the new schedule to its neighbors. The performance evaluation shows that the U-MAC can have good throughput when traffic loading is high. Meanwhile, U-MAC can save more energy than S-MAC when traffic loading is low. However, the problems of duty cycle synchronization between sender and receiver are not well solved in U-MAC. For a forwarding sensor, the duty cycle will be increased for getting more time to transfer data. For a receiving sensor with low traffic utilization, the duty cycle will be decreased. After a long time, the forwarding sensor is working in the high duty cycle and the receiving sensor may work in the low duty cycle. When the forwarding sensor sends a packet to a receiving sensor, the packet may be lost due to the receiving sensor enters into its sleep period early. As a result, channel utilization and transmission delay are varying in U-MAC.

In this paper, we propose a dynamic traffic-aware duty cycle adjustment MAC protocol named "TA-MAC" which can adjust sensor's duty cycle adaptively according to status of sending/receiving buffer, traffic loading, and battery life. After the information of a sensor is obtained, the sensor adaptively adjusts the duty cycle for sending and receiving packets. The goals of the proposed MAC protocol are twofold: (i) to conserve energy on sensors with low traffic loading and (ii) to decrease transmission latency and increase data throughput on sensors with heavy traffic loading. Each sensor can have different duty cycle for data transmission, long duty cycle is allocated for sensors that have more data need to be sent in sending queue. Each sensor extends its duty cycle if the sensor has data need to transmit. For sensors that failed to send data due to radio channel competition, these sensors will go to sleep state for energy conserving and wait for next wake-up time to compete the radio channel again for transmitting data.

The rest of the paper is organized as follows. Section 2 describes the related works. Section 3 presents the proposed dynamic duty cycle adjustment MAC protocol. Section 4 shows the experimental results. Section 5 concludes the proposed MAC scheme.

2. Related Works

This section provides a brief overview of existing energy conserving MAC protocols in asynchronous and synchronized approaches.

For asynchronous protocols, for example, B-MAC [12], WiseMAC [13], and X-MAC [14], they are utilizing adaptive preamble sampling schemes to reduce duty cycle and minimize idle listening. In B-MAC, nodes are awake and asleep asynchronously. When a sender has data, the sender sends a packet with a preamble longer than the sleep period of the receiver. When the receiver wakes up and detects the preamble, it stays awake to receive the data. B-MAC equips periodic channel sampling, called low-power listening (LPL), to enable low-power communication, without the need of explicit synchronization among the nodes. The receiver only wakes for a short time to sample the medium, which reduces the energy consumptions. In [13], El-Hoiydi and Decotignie proposed the WiseMAC protocol for the downlink channel of infrastructure wireless sensor networks. WiseMAC is based on the preamble sampling technique [15]. By sampling the medium, WiseMAC listens to the radio channel for a short duration, for example, the duration of a modulation symbol, to check for network activities. Nodes sample the medium with the same constant period. In WiseMAC, the access point learns the sampling schedule of all sensor nodes. Having the sampling schedule of the destination, the access point starts the transmission with a wake-up preamble. To reduce the length of the extended preamble, the receiver sends the time of its next awake period in the data acknowledgment frame. By using the preamble sampling technique, WiseMAC can reduce energy usage for transmitting data. However, WiseMAC is not suitable for long distance multihop data transmission. In [14], Buettner et al. proposed the X-MAC protocol that uses a shortened preamble approach for retaining the benefits of low-power listening. In wireless sensor networks, long preamble introduces excess latency at each hop and suffers from excess energy consumption at nontarget receivers. Instead of sending a long preamble stream, X-MAC inserts small pauses in the preamble stream, which forms a series of short preamble streams. These gaps enable the receiver to reply an early acknowledgement packet back to the sender during the short pause between preamble streams. When a sender receives an acknowledgement packet from the receiver, it stops sending preamble and then starts the transmission of data packet. Nontarget receivers who overhear the strobed preamble can go back to sleep immediately. X-MAC's shortened preamble approach can reduce energy usage at both the transmitter and receiver side. X-MAC also proposed an algorithm for adapting the duty cycle of the receiver to adapt to different traffic loads. However, the effect of the proposed adaptive algorithm is not clear in the performance evaluation.

Synchronized protocols, such as S-MAC [9], T-MAC [10], and U-MAC [11], can specify when nodes are awake

and asleep within a frame. In S-MAC, neighboring nodes form virtual clusters to synchronize the working schedules [7]. Nodes periodically wake up, receive and transmit data, and then return to sleep to reduce energy consumptions. Within a virtual cluster, a node exchanges synchronization and schedule information with its neighbors to ensure that the node and its neighbors wake up at the same time. After the schedule is synchronized, nodes with data can transmit packets using RTS (Request To Send) and CTS (Clear To Send) packets when the nodes awake. Nodes without data will be in idle state for waiting connection until the end of the awake period and the nodes then enter sleep mode. S-MAC uses a SYNC packet to accomplish the synchronization of working schedule. When a node receives the SYNC packet, the node will adjust its timer immediately. In S-MAC, the listen interval is divided into two parts: (i) interval for receiving SYNC packets and (ii) interval for receiving RTS/CTS packets.

In [10], Van Dam and Langendoen proposed the T-MAC protocol that enhances the design of S-MAC by listening to the channel for only a short time, and if no data is received during this window, the node enters sleep mode. After the synchronization, T-MAC shortens the awake period of a node if the channel is idle for a short time. Once a data is received, the node remains awake until the transmission is finished or the awake period ends. By using adaptive duty cycling, T-MAC can reduce energy usage for data transmission. As a tradeoff, the costs are the reduced data throughput and the increased transmission latency. In [11], Yang et al. proposed the U-MAC protocol that tunes duty cycle according to the traffic loads of sensors. In a wireless sensor network, making nodes asleep for a long time to lower the energy consumption may result in higher latency. U-MAC tries to solve the problem by tuning sensor's duty cycle according to the traffic loads and selective sleeping after transmission. To reflect the traffic loads, U-MAC adopts a utilization function to calculate the loads of each sensor. Upon synchronization, the sensor calculates its traffic utilization since the last synchronization time. The sensor adjusts its duty cycle according to the calculated utilization and then broadcasts the new schedule to its neighbors.

Most of the recent works focus on how to prolong a sensor's or network's lifetime. However, these works do not consider the performance of data transmission when sensors have different workload. Our work differs from all of the above works by striking the balance between the battery life and transmission latency according to the traffic situations. The main difference is that the proposed MAC protocol is based on the demand of data transmission of sensors. The use of traffic-aware protocol can conserve sensors' battery life and reduce the transmission latency caused by sleep.

3. The Dynamic Traffic-Aware Duty Cycle Adjustment MAC Protocol

Most energy conserving MAC protocols adjust the sensor's duty cycle according to some factors like transmission latency and traffic load. For example, DS-MAC [16] adjusts the
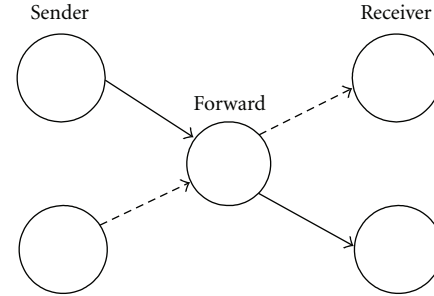

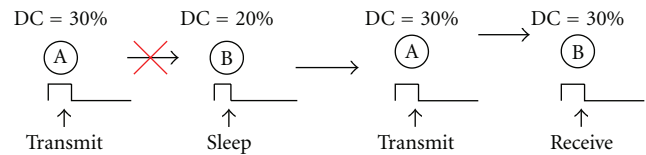FIGURE 1: Sensors in U-MAC with cross topology.

FIGURE 2: The increasing of sensors' duty cycles raises transmission efficiency.

sensor's duty cycle according to the data transmission delay of sensors. U-MAC [5] determines the sensor's duty cycle according to sensor's traffic utilization. Some duty cycle adjustment algorithms change the sensor's listening time, others adjust the sensor's frame time [16, 17]. Existing adaptive duty cycle algorithms consider only limited factors to adjust the sensor's duty cycle. The lack of complete information of the network and sensor's sending/receiving status leads to additional energy consumption. Figure 1 shows an example of data transmission in U-MAC with cross topology. In Figure 1, the forwarding node needs to receive data from senders and then transmit the data to receivers. Since U-MAC adjusts the sensor's duty cycle according to the calculations of sensor's traffic utilization, the forwarding node has higher duty cycle than the other nodes. On the other hand, the sending nodes and receiving nodes will have lower duty cycle than the forwarding node. If a receiving node's duty cycle is the same as the forwarding node, increasing duty cycle can indeed improve the data transmission rate. Figure 2 shows that the duty cycle adjustment can enhance the transmission efficiency if the participant sensors increase their duty cycles.

In U-MAC, senders and receivers try to reduce the duty cycle to save energy consumption if they are in low traffic utilizations. Figure 3(a) shows that all sensor nodes have the initial duty cycle of 20%. Over a long time, Figure 3(b) shows the senders A, B, C and receivers E, F, G reduce the duty cycle because of the low traffic utilizations. However, the forwarding node D increases its duty cycle in high usage status because of node D is communicating with multiple senders and receivers. When the senders and the receivers go to sleep, the forwarding node keeps on trying to transfer data packets from senders to receivers because of the forwarding node is in high duty cycle. In case of senders and receivers in low duty cycle, the forwarding node cannot receive packets from senders because of senders are in sleeping. Meanwhile,
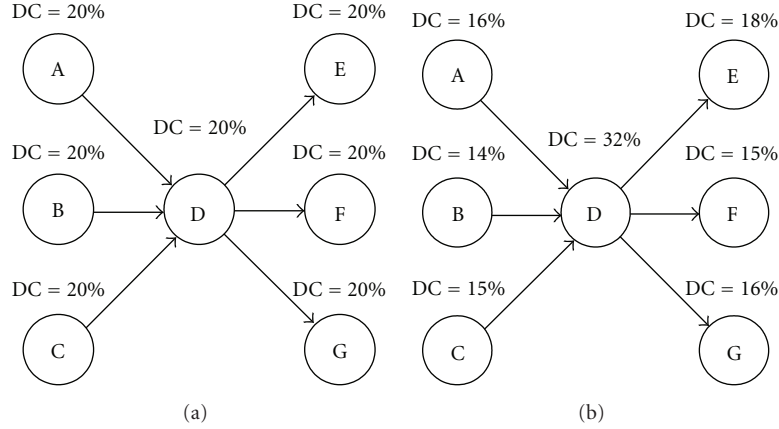
Figure 3: The result of duty cycle adjustment of sensors in U-MAC over a long time.

the forwarding node cannot transfer data packets to receivers because of receivers are in sleeping too. As a result, the forwarding node is wasting energy as it is in idle state.

Reducing the duty cycle of sensors can save energy to prolong sensors' life when sensors are in low traffic loads. Extending the duty cycle of sensors can improve data efficiency and reduce transmission delay when sensors are in high traffic loads. However, the duty cycle synchronization problems are not well solved in U-MAC. As a result, channel utilization and transmission delay are varying case by case in U-MAC.

To overcome the aforementioned issues, we propose a dynamic traffic-aware duty cycle adjustment MAC protocol named "TA-MAC" that can adjust duty cycle adaptively according to status of sensor's sending/receiving buffer, traffic loading, and battery life. The proposed traffic-aware duty cycle adjustment MAC protocol is based on the existing U-MAC protocol, and more functions are designed by us to enhance it. In U-MAC, the forwarding node has higher duty cycle due to its high traffic load. When senders continue to send data packets to receivers, this makes the forwarding node continues to increase the duty cycle to adapt to the current load. However, the senders and receivers do not have such a high traffic load. Therefore, the senders and receivers will lower their duty cycle. To solve such issue, the proposed MAC protocol utilizes an additional control message to achieve duty cycle synchronization between two sensor nodes. The receiving node can decide whether to keep the same duty cycle or not. If the forwarder and receiver have the same duty cycle, the sensors can have more time to transmit data. Based on such idea, the proposed MAC protocol performs outstanding than related schemes under various situations, and the performance comparisons are shown in next section. In the following, we present the technical details of the TA-MAC protocol.

3.1. TA-MAC Algorithm.
Algorithms 1 and 2 present the proposed TA-MAC algorithm. Table 1 depicts the symbols used in the proposed algorithm. The initial duty cycle of all nodes is set to the minimum duty cycle value DC_{min}.

Table 1: Symbols used in the TA-MAC algorithm.

Symbol	Meaning
$Queue_{high}$	Queue high
WR	Working rate
WR_{high}	Working rate high
WR_{low}	Working rate low
T_{tx}	Total transmission time
T_{rx}	Total receive time
Tidle	Total idle time
DC	Duty cycle
DC_{max}	The maximum of duty cycle
DC_{min}	The minimal of duty cycle
PL	Power level

The algorithm is divided into two parts: (i) sending duty cycle adjustment packet and (ii) processing the duty cycle adjustment packet.

3.1.1. SendDCAdjust: Sending Duty Cycle Adjustment Packet.
When the transfer of a synchronization packet is completed, the node enters into duty cycle adjustment mode, where the adjustments can be further divided into two conditions: (i) normal state and (ii) abnormal state.

The duty cycle adjustment in normal state is based on the sensor's working rate, duty cycle, and the remaining energy. Working rate WR denotes the proportion of working time of a wireless sensor in a period of duty cycle. The working rate WR can be calculated by the total working time divided by the total operating time (active time). If the current working rate WR is higher than the working rate threshold WR_{high}, the current duty cycle is lower than the maximum duty cycle DC_{max}, and the remaining energy is in normallevel, then the duty cycle will be increased by n%. If the current working rate WR is lower than the working rate threshold WR_{low} and the current duty cycle is higher than the minimum duty cycle DC_{min}, then the duty cycle will be decreased by n%. When the traffic load is high, the sender's duty cycle will continue to increase until it reaches the DC_{max} value. The sender will

---

Symbols definition:
Pkt: the packet that generated by sensor
end definition
1: Initial $DC : DC_{min}$
2: **while** a SYNC packet is sent **do**
3:       Sending_Queue = Sensor.GetSendingQueueLength()
4:     **if** Sending_Queue $\geq$ Queue$_{high}$ and Sensor.DC < DC$_{max}$ and Sensor.PL = N and Sensor.State $\neq$ Queue$_{high}$ **then**
5:         Sensor.DC = DC$_{max}$
6:         Pkt.Type = DC
7:         Pkt.Data = Sensor.DC
8:         Send(Pkt)
9:         Sensor. State = Queue$_{high}$
10:         break
11:     **end if**
12:     $WR = \dfrac{T_{tx} + T_{rx}}{T_{tx} + T_{rx} + T_{idle}} * 100\%$
13:     **if** Sensor.WR $\geq$ WR$_{high}$ and Sensor.DC < DC$_{max}$ and Sensor.PL = N **then**
14:         Sensor.DC = Sensor.DC $*(1 + n\%)$
15:     **else if** Sensor.WR $\leq$ WR$_{low}$ and Sensor.DC > DC$_{min}$ **then**
16:         Sensor.DC = Sensor.DC $*(1 - n\%)$
17:     **end if**
18:     Curren_DC = Sensor.DC
19:     **if** Currcnt_DC $\geq$ DC$_{max}$ and Sensor. State $\neq$ DC$_{max}$ **then**
20:         Pkt.Data = Sensor.DC
21:         Send(Pkt)
22:         Sensor.State = DC$_{max}$
23:         break
24:     **end if**
25:     //when the DC is less than DC$_{max}$, clean the request
26:     **if** Sensor.State = DC$_{max}$ and Sensor.DC < DC$_{max}$ **then**
27:         Sensor.State = Normal
28:     **end if**
29:     //when the queue length less than the queue threshold, clean the request
30:     **if** Sensor. State = Queue$_{high}$ and Sending_Queue < Queue$_{high}$**then**
31:         Sensor.State = Normal
32:     **end if**
33: **end while**

ALGORITHM 1: Send DCAdjust packet.

send a control packet to notify the receiver that the duty cycle is adjusted. The receiver will then adjust its duty cycle if the receiver has enough energy. If the receiver does not have enough energy, the receiver will not change its duty cycle.

The duty cycle adjustment in abnormal state is triggered when some situations occurred, for example, sending/receiving queue is full. Each node has a sending queue and a receiving queue for storing incoming/outgoing packets temporarily. When a packet is generated, the packet is immediately put into the sending queue and waiting for transmission. Since the queue length is limited, when the queue is full, later packets will be dropped due to out of memory, resulting in the reduction of transmission throughput. Figure 4 shows the illustration of packets that are dropped because the sending queue is full.

To overcome the above-mentioned issue, before the sending queue is going to full, the proposed MAC protocol will increase the sensor's duty cycle rapidly to the maximum duty cycle value DC$_{max}$ for prolonging the working time to transmit queued packets. However, the sensor's survival time will be greatly reduced if the sensor's duty cycle is further increasing when a sensor is in low-battery condition. Therefore, the current sensor's remaining energy is an important factor for the duty cycle adjustment. The states of remaining energy can be divided into (i) low-energy state and (2) normal-energy state. As shown in Table 2, L for low level, representatives the remaining energy is currently less than 40% of full battery life. N is normal, representatives the current residual energy is sufficient for the general use. In the case of the sensor is in the low-energy state, the duty cycle cannot be increased in order to prolong the sensor's life time. When the remaining energy is in the normal energy state, the proposed MAC protocol will increase the sensor's duty cycle for sending more queued packets.

Increasing the duty cycle can improve the efficiency of packet transmission; however, the receiver must be synchronized to follow the same schedule in order to receive the data sent from the sender. The duty cycle in the sender and receiver is independent. When the sender's duty cycle is higher than the receiver's duty cycle, it will cause the receiver

```
1: while receiving a packet do
2:         if Pkt.Type = DC then
3:                 if SensorDC < DC_max and Sensor.PL = N then
4:                         Sensor. DC = Pkt.Data
5:                 end if
6:         else
7:                 if Pkt.Type = SYNC then
8:                         Sync()
9:             end if
10:        else
11:                if Pkt.Type = RTS then
12:                        Rts()
13:                end if
14:        else
15:                if Pkt.Type = CTS then
16:                        Cts()
17:                end if
18:        else
19:                if Pkt.Type = DATA then
20:                        Data()
21:                end if
22:        end if
23: end while
```

ALGORITHM 2: Handle DCAdjust packet.
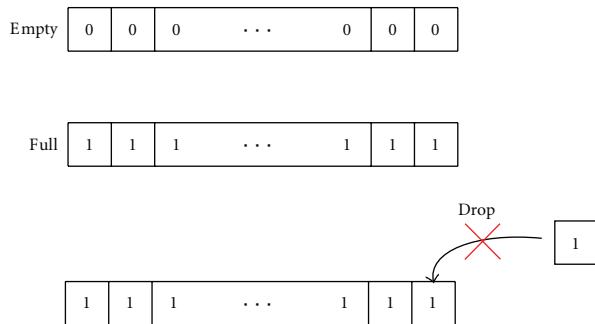
FIGURE 4: The illustration of packets that are dropped because the sending queue is full.

TABLE 2: Energy Level.

Name	Low level	High level
Remaining energy	0% ~ 39%	40% ~ 100%
PL	L (low)	N (normal)

to sleep earlier than the sender. As a result, the data cannot be sent to the receiver even the sender is still in working condition. In Figure 5, node A is the sender and node B is the receiver, the packet throughput cannot be increased even the node A increases its duty cycle, because of node B is in sleeping state. Therefore, node A wastes its precious battery power in idle waiting.

For the above problem, the proposed MAC protocol sends an additional control packet to let the sender and receiver have the same duty cycle. In the proposed MAC

protocol, a control packet named DCAdjust is defined. In Figure 6, node A and node B can have the same duty cycle after additional duty cycle synchronization. Finally, node A can have more time to transmit data to node B. As a result, data throughput is increased. Figure 7 illustrates the structure of the control packet DCAdjust. The packet is used to tell the receiver about the current state of the sender's duty cycle. The receiver will determine whether to synchronize the sender's duty cycle or not according to its remaining battery power. The structure of the control packet is similar to the IEEE 802.11 RTS (Request To Send) packet. The control packet has a 1-byte field named DC information, which is used to record the sender's duty cycle. Since the length of the control packet DCAdjust is small, the energy spent in sending and receiving the control packet DCAdjust is low.

3.1.2. Handle the DCAdjust Packet. When the receiver is receiving the control packet, DCAdjust, the receiver will adjust its duty cycle according to its remaining energy. If the receiver does not have enough energy, the receiver will not change its duty cycle.

4. Performance Evaluation

In order to evaluate the performance of the proposed control scheme, we use the OMNeT++ discrete event simulator with EYES WSN simulation framework to simulate the wireless sensor networking environment [12]. There are five nodes in a 10 by 10 grid in the simulated sensor model. Nodes exchange messages using the wireless communication. A message will be heard by all the neighbors situated within the transmission range. Three duty-cycle-based MAC protocols
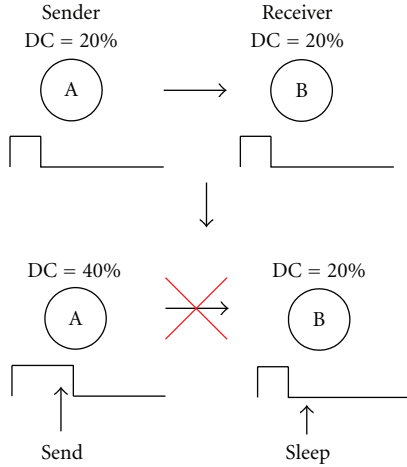
FIGURE 5: The packet throughput cannot be increased even if the node A increases its duty cycle because node B is in sleeping state.
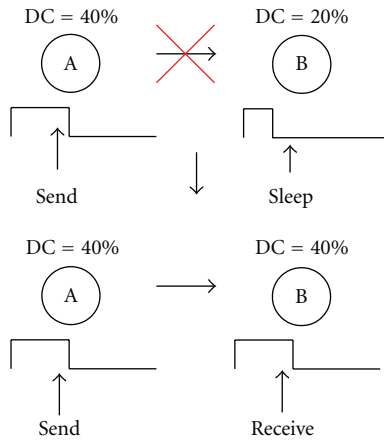

FIGURE 6: Node A and node B have the same duty cycle after duty cycle synchronization.
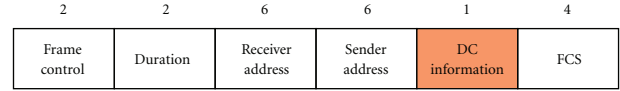

FIGURE 7: The data structure of the DCAdjust control packet.
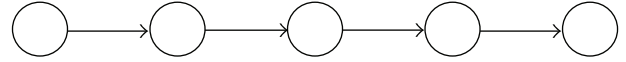

FIGURE 8: Linear topology.

TABLE 3: Common parameters of different MAC protocols.

Queue = 100
Sleep power = 0.02 mA
Transmission power = 10 mA
Receive power = 4 mA
Simulation time = 300 s
Bandwidth = 115.2 Kbps
Data packet size = 100 byte
SYNC period = 10 s
Frame time = 1 s
Initial duty cycle = 10%
DC_{max} = 40%
DC_{min} = 10%
Transmission range = 20 M
Node = 5∼20

TABLE 4: Unique parameter of U-MAC and TA-MAC.

U-MAC		TA-MAC	
U_{high}	0.08	WR_{high}	8%
U_{low}	0.04	WR_{low}	4%
D_{max}	1.8	PL	2

are evaluated in the simulation: S-MAC, U-MAC, and TA-MAC. In the S-MAC control scheme, a frame length is set to one second and with a length of the active time 100 ms. The topologies used to evaluate our proposed scheme are shown in Figures 8 and 9. Figure 8 shows five nodes that transmit data in a linear topology. Figure 9 shows five nodes that transmit data in a cross topology.

In the simulation, the power consumption and data communication parameters are based on the value of TR1001 hybrid radio transceiver [13]. In our experiments, we set the transmission range to 20 meters. The bandwidth is set to 115.2 kbps. Each data packet size is 100 bytes and 4 bytes for control packets including RTS, CTS, and SYNC/LSYNC. The simulated parameters can be divided into two categories: (i) common parameters and (ii) unique parameters. The common parameters are used in all simulated MAC protocols. The unique parameters are used in specific MAC protocols for specifying special parameters to specific MAC protocols. Table 3 depicts the common parameters, and Table 4 depicts the unique parameter of different MAC protocols in the simulating environment.

Figures 10, 11, and 12 show the simulation results of the different MAC protocols in linear topology. Figure 10 shows the data throughput of the different MAC protocols. In Figure 10, TA-MAC and U-MAC increase sensors' duty cycle with n% under high traffic load when the sending/receiving queue is not full. When the sending queue usage exceeds the queue threshold (Queue$_{high}$), TA-MAC rapidly adjusts its duty cycle to the maximum duty cycle (DC_{max}) and sends a DCAdjust packet to notify the receiver. In TA-MAC, the sender can have more time to transmit data. In U-MAC, the duty cycle is in shortage due to U-MAC increases its duty cycle slowly with n% in such a condition. S-MAC adopts fixed duty cycle design to transmit data packets whether the traffic load is high or not. As a result, TA-MAC has better data throughput than U-MAC and S-MAC.

Figure 11 shows the transmission delay of the different MAC protocols in linear topology. The lack of working time in S-MAC and U-MAC increases the transmission latency between senders and receivers. TA-MAC can increase duty cycle significantly when the traffic load is high. Meanwhile,
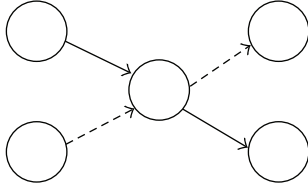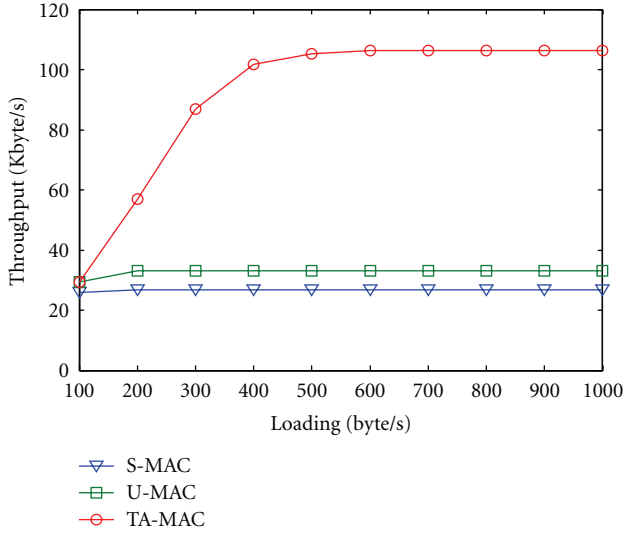
FIGURE 9: Cross topology.

FIGURE 10: Data throughput for nodes in linear topology.

FIGURE 11: Transmission latency for nodes in linear topology.

FIGURE 12: Energy consumption for nodes in linear topology.

the sender and receiver can achieve duty cycle synchronization. Therefore, TA-MAC can have lower transmission latency than that in U-MAC and S-MAC.

Figure 12 shows the total energy consumptions of the different MAC protocols in linear topology. S-MAC uses a fixed duty cycle to transmit data. Therefore, the energy consumption of S-MAC does not increase as the traffic load increases. U-MAC increases sensor's duty cycle according to sensor's traffic load; however, the increasing is slow. Therefore, the energy consumption of U-MAC is increasing slowly until it reaches the maintenance level. TA-MAC adjusts its duty cycle to the maximum duty cycle (DC_{max}) and sends a DCAdjust packet to notify the receiver for improving data throughput, which consumes more battery energies. Therefore, TA-MAC takes more energy consumption than that in S-MAC and U-MAC for increasing data throughput and reducing transmission latency. In linear topology, the proposed TA-MAC can have good performance than S-MAC and U-MAC because TA-MAC can adjust the sensor's duty cycle efficiently. Meanwhile, the DCAdjust packet can make the sender and receiver have the same duty cycle, which can reduce the energy wasted in idle listening.

In the cross topology, the forwarding node suffers from high traffic loads from senders. The packet queue becomes full immediately when senders send data. Figure 13 shows the data throughput of the different MAC protocols in cross topology. In Figure 13, TA-MAC adjusts the forwarding node's duty cycle with n% according to its traffic load and sends a DCAdjust packet to the receiver for synchronizing
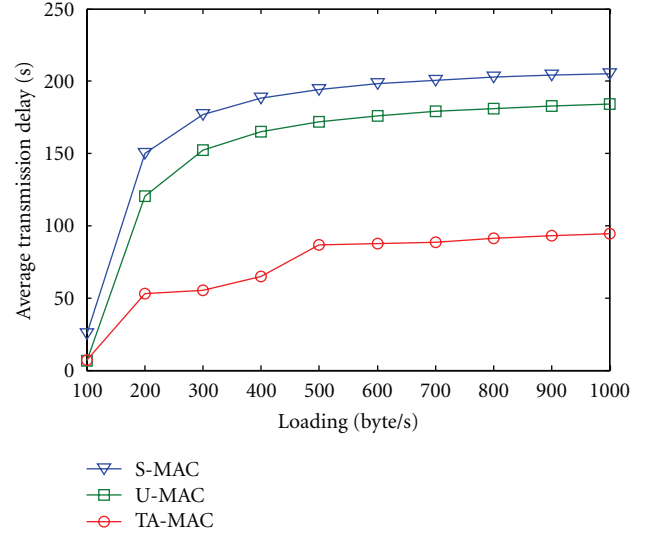
duty cycle. When the receiver receives the DCAdjust packet, the receiver adjusts its duty cycle to the same value of the sender. After the duty cycle is synchronized, the sender can transmit data to the receiver without scheduling miss and then the data throughput is increasing rapidly. As the traffic load increases, the duty cycle of the forwarding node reaches the maximum duty cycle (DC_{max}) value. Then, the data throughput will be decreased due to the forwarding node cannot get more time to transmit data. S-MAC uses fixed duty cycle design to transmit data packets whether the traffic load is high or not. In S-MAC, the data throughput remained in a stable state. In U-MAC, the forwarding node increases its duty cycle with n% in such a condition. However, the duty cycles in sender and receiver are not synchronized. Therefore, the forwarding node cannot transmit data to the receiver due to scheduling miss. As a result, TA-MAC has better data throughput than U-MAC and S-MAC.
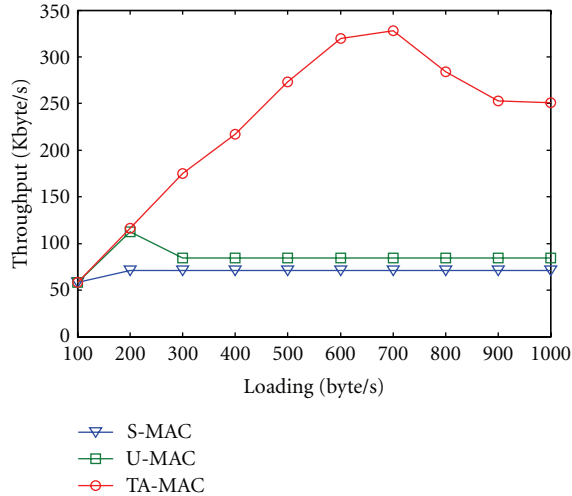
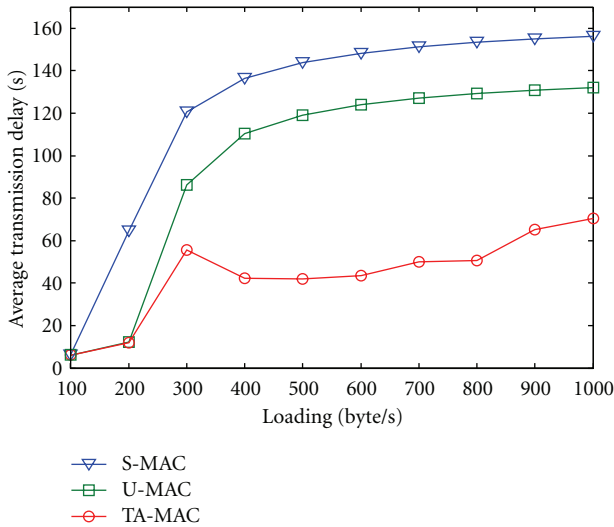FIGURE 13: Data throughput for nodes in cross topology.

FIGURE 14: Transmission latency for nodes in cross topology.

Figure 14 shows the transmission delay of the different MAC protocols in cross topology. U-MAC adjusts the sensor's duty cycle according to its traffic load. U-MAC has longer duty cycle than that in S-MAC. Therefore, U-MAC has shorter transmission delay, compared with S-MAC. TA-MAC can increase duty cycle significantly when the traffic load is high. Meanwhile, the sender and receiver can achieve additional duty cycle synchronization. As a result, TA-MAC can have lower transmission latency than that in U-MAC and S-MAC.

Figure 15 shows the total energy consumptions of the different MAC protocols in cross topology. The energy consumption of S-MAC does not increase as the traffic load increases due to its fixed duty cycle design. U-MAC increases a sensor's duty cycle according to the sensor's traffic load. The forwarding node consumes more energy than other nodes. U-MAC has lower energy consumption than TA-MAC because the receivers in U-MAC are sleeping more due to out of duty cycle synchronization. On the contrast, T-MAC
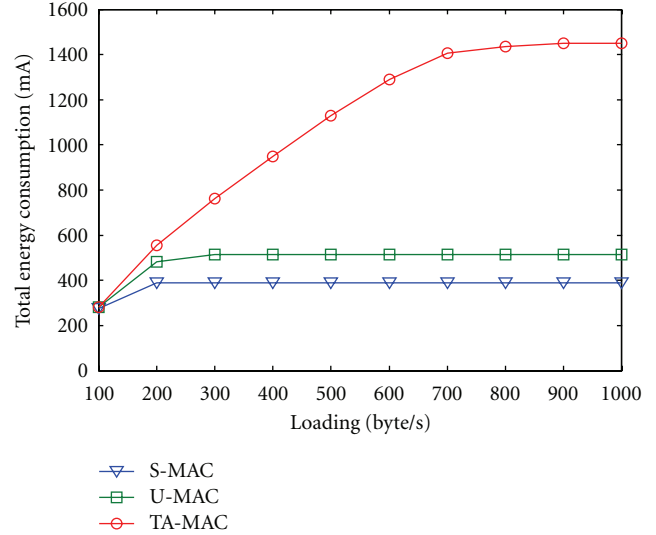

FIGURE 15: Energy consumption for nodes in cross topology.

has more time to transmit queued data. Therefore, the energy consumption in TA-MAC is higher than that in S-MAC and U-MAC.

5. Conclusions

In order to reduce the energy consumptions when nodes are in idle listening, duty-cycle-based MAC protocols are introduced to let node go into sleep mode periodically or aperiodically. When a sensor's listening time is extended, the sensor can have more time to receive data, which can enhance the transmission efficiency and reduce the latency when sensors have the heavy load. When a sensor's duty cycle is reduced, the sensor has long sleep time to reduce energy consumption in idle listening. In this paper, a dynamic traffic-aware MAC protocol for energy conserving in wireless sensor networks is proposed. The proposed MAC protocol can provide better data transmission rate when sensors are with high traffic loading. Meanwhile, the proposed MAC protocol can save energy when sensors are with low traffic loading. Simulation results show that the proposed protocol has better data throughput than other duty-cycle-based MAC protocols, for example, S-MAC and U-MAC.

Disclosure

C.-C. Chen was an associate professor in Southern Taiwan University while this work was initiated, and he is with Institute of Manufacturing Information and Systems, National Cheng Kung University since 2011.

References

[1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, "Energy-efficient communication protocol for wireless microsensor networks," in *Proceedings of the 33rd Annual Hawaii International Conference on System Siences (HICSS '00)*, pp. 3005–3014, January 2000.

[2] J. Li and G. Y. Lazarou, "A bit-map-assisted energy-efficient MAC scheme for wireless sensor networks," in *Proceedings of the 3rd International Symposium on Information Processing in Sensor Networks (IPSN '04)*, pp. 55–60, April 2004.

[3] Y. S. Chen and Y. W. Lin, "C-MAC: an energy-efficient MAC scheme using chinese-remainder-theorem for wireless sensor networks," *Journal of Information Science and Engineering*, vol. 23, no. 4, pp. 1057–1071, 2007.

[4] C. Shanti and A. Sahoo, "DGRAM: a delay guaranteed routing and MAC protocol for wireless sensor networks," in *Proceedings of the 9th IEEE International Symposium on Wireless, Mobile and Multimedia Networks (WoWMoM '08)*, June 2008.

[5] H. Gong, M. Liu, L. Yu, and X. Wang, "An event driven TDMA protocol for wireless sensor networks," in *Proceedings of the WRI International Conference on Communications and Mobile Computing (CMC '09)*, vol. 2, pp. 132–136, January 2009.

[6] L. Shi and A. O. Fapojuwo, "TDMA scheduling with optimized energy efficiency and minimum delay in clustered wireless sensor networks," *IEEE Transactions on Mobile Computing*, vol. 9, no. 7, Article ID 5432185, pp. 927–940, 2010.

[7] D. Wu, G. -Y. Wang, and X. -L. Li, "Distributed TDMA scheduling protocol based on conflict-free for wireless sensor networks," in *Proceedings of the International Conference on Intelligent Computing and Integrated Systems (ICISS '10)*, pp. 876–879, October 2010.

[8] W. Zhao and X. Tang, "Scheduling data collection with dynamic traffic patterns in wireless sensor networks," in *Proceedings of the International Conference on Computer Communications (IEEE INFOCOM '11)*, pp. 286–290, April 2011.

[9] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the IEEE International Conference on Computer Communications (IEEE INFOCOM '02*, vol. 3, pp. 1567–1576, June 2002.

[10] T. van Dam and K. Langendoen, "An adaptive energy-efficient MAC protocol for wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 171–180, November 2003.

[11] S. H. Yang, H. W. Tseng, E. H. K. Wu, and G. H. Chen, "Utilization based duty cycle tuning MAC protocol for wireless sensor networks," in *Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM '05)*, vol. 6, pp. 3258–3262, December 2005.

[12] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," in *Proceedings of the Second International Conference on Embedded Networked Sensor Systems (SenSys '04)*, pp. 95–107, November 2004.

[13] A. El-Hoiydi and J. D. Decotignie, "Low power downlink MAC protocols for infrastructure wireless sensor networks," *Mobile Networks and Applications*, vol. 10, no. 5, pp. 675–690, 2005.

[14] M. Buettner, G. Yee, E. Anderson, and R. Han, "X-MAC: a short preamble mac protocol for duty-cycled wireless networks," Tech. Rep. CU-CS-1008-06, University of Colorado at Boulder, 2006.

[15] A. El-Hoiydi, "Aloha with preamble sampling for sporadic traffic in Ad Hoc wireless sensor networks," in *Proceedings of the International Conference on Communications (ICC '02)*, pp. 3418–3423, April 2002.

[16] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '04)*, vol. 3, pp. 1534–1539, March 2004.

[17] G. Xing, C. Lu, Y. Zhang, Q. Huang, and R. Pless, "Minimum power configuration in wireless sensor networks," in *Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC '05)*, pp. 390–401, May 2005.