

# ENERGY EFFICIENT SCHEDULING ALGORITHM FOR S-MAC PROTOCOL IN WIRELESS SENSOR NETWORK

D Saha, M R Yousuf, and M A Matin

Department of Electrical Engineering and Computer Science, North South University,  
Dhaka, Bangladesh

m.a.matin@ieee.org

## **ABSTRACT**

*In Sensor-MAC (S-MAC) protocol, a node located between two or more virtual clusters is called border node that adopts different listen and sleep schedules. These border nodes consume a large amount of energy as they switch to the listening mode often due to diversified scheduling which in turns decreases the lifetime of the wireless sensor network. This paper proposed a new unified scheduling method to solve the diversified scheduling problem of border nodes in S-MAC and evaluated the performance through simulation. It has been observed from the simulated results that the border nodes have consumed less power in case of large network as well as small networks.*

## **Keywords**

*S-MAC protocol, Wireless Sensor Network (WSN), Uni-Scheduling packets*

## **1. INTRODUCTION**

Wireless sensor network (WSN) has become known as one of the potential applications of the emerging technology. It has been used for a wide range of purposes such as robotic exploration, environment monitoring, medical systems, etc. It has become a hot research issue which is regarded as one of the ten influencing technologies in the 21<sup>st</sup> century [1]. The WSN is made up of micro devices called Sensor Nodes (SN) which build the backbone of such networks and operate upon a predetermined set of instructions. As these sensor nodes are normally powered by batteries, they can only provide small and limited processing capabilities. The problem become very serious when these batteries are non-rechargeable in practical environment and a significant amount of power is spent for processing of required information. Thus a variant rate of depleted energy of the nodes can seriously hamper the network's efficiency and therefore its lifetime [2]. The major source of energy loss in WSN is *collision* or *corruption* during broadcasting of packets. The second major cause of energy loss is *idle listening*, that is, listening to an idle channel in order to receive any possible traffic. The next source of energy waste is *overhearing*, of a node to receive some packets that are destined for other nodes. The use of *control packet overhead* to setup data transmission is another source of energy loss. *Over emitting* is another source for energy waste which is caused by transmission of a message when the destination node is not ready.

Several type of protocols like TDMA and MAC [3] protocols are proposed to prolong the lifetime of WSN. In our paper, S-MAC (Sensor-MAC) protocol [4] is considered as it has shown better performance in minimizing power consumption. Yan-Xiao Li et al. [5] evaluated the performance of S-MAC protocol by improving latency and jitter. They have achieved their desired result by varying the duty cycle of the S-MAC. Though S-MAC protocol has shown better performance, it has some problems in initializing the listen and sleep schedules. To overcome this problem W. Lee et al. [6] designed an algorithm to minimize the energy consumption of border nodes by using unified schedule and modifying the synchronization packet. However, the nodes would incur delay and go through idle listening time if collision happened due to large packet size. E. M. Shakshaki et al. [7] presented a unified scheduling approach where the network got synchronized under a single schedule. Though the longevity of the WSN life has increased to some extent, the border nodes die out quickly as they stay in the listening state for longer period. T. S. Lee et al. [8] had shown the longevity of border nodes; but it is not suitable for large network size. Y. Yang et al. [9] performed scheduling from the sink node and therefore, in large networks the synchronization time will be longer and energy is wasted due to idle listening.

In this paper, an algorithm has been proposed to overcome the above limitations by using synchronization of sensor nodes under a unified schedule. This schedule will be broadcasted periodically from the border nodes. This means it broadcast constant synchronization packets to the whole sensor network, so that all the nodes can communicate with each other under a single schedule. In this way, the life time of all the nodes in the network will increase.

## 2. S-mac Protocol

S-MAC protocol is a medium access control protocol which is widely used in WSNs for energy conservation. It retains the flexibility of contention-based protocols similar to IEEE 802.11. There are three major energy consumptions in S-MAC that are identified. These are: a) *Collision* which results in energy waste due to retransmission of collided packets. b) *Overhearing* that occurs when a node listens to transmissions which are not intended for it. c) *Idle listening* which occurs when a node listens to receive any possible data that is not sent.

The communication between nodes takes place when S-MAC protocol exchange packets that starts with Carrier Sense (CS) to avoid collision. Then followed by Read to Send and Clear to Send (RTS/CTS) packets exchanged for unicast type data packets shown in the Fig-1. Upon successful transmission of these packets data communication takes place.

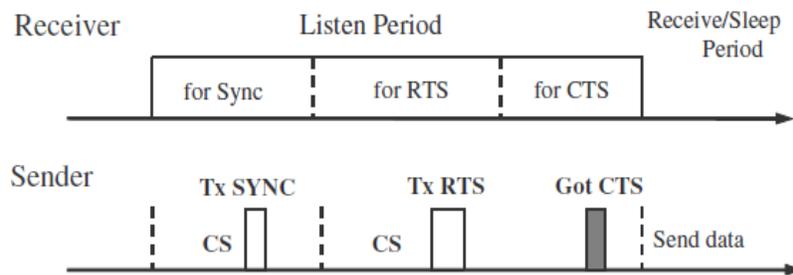


Fig.-1: S-MAC Messaging Mechanism

In [10], the nodes in WSN with S -MAC protocol keep on the listen state for 10 seconds in every 2 minutes. The node is not engaged in transmission or reception if S-MAC is in the sleep

state or if its neighbors are involved in communication. The sleep state exits in S-MAC protocol to reduce collision and overhearing. The node wakes up at the end of its neighbor's transmission to relay the packets. This task is performed by overhearing neighbor's RTS and CTS exchanges and then the node goes to sleep and serve the purpose of reducing latency. This behavior of S-MAC protocol is called *adaptive listening* [11, 12] and the technique for optimizing power consumption is *message passing*.

Since all the immediate nodes have their own sleep schedules, periodic sleep may results in high latency. The latency caused by periodic sleep is called *sleep delay*. Schedules are periodically exchanged by broadcasting SYNC packets among neighboring nodes. The SYNC packet is very small and includes the address of the sender and the time of its next sleep. As soon as, a receiver gets the time from the SYNC packet it subtracts the packet transmission time and use the new value to adjust its timer. Due to the inconsistent of time cycle, different virtual clusters are formed. The communication between these virtual clusters takes place when a common node between them adopts both the schedules. In this way, the border nodes are listening for a longer period of time and die out quickly.

There are two main reasons of multiple scheduling in a single network. Firstly, when nodes establish their own schedules, some nodes are situated far away and cannot hear each other's schedules. Secondly, if two nodes broadcast their schedules at the same time, collisions may take place. In both the situations, the nodes must now choose their own schedules.

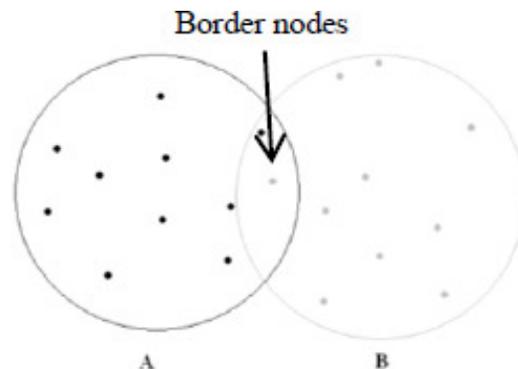


Fig.-2: Border node and Virtual Cluster

### 3. PROBLEMS WITH DIVERSIFIED SCHEDULING IN S-MAC PROTOCOL

#### 3.1. DELAY TIME IN SYNCHRONIZATION OF SCHEDULE

S-MAC scheduling mechanism works when self-configuration is in set mode. In the listen period, a node senses its neighbor nodes and transmits SYNC packets that contain randomly generated schedule. Thus a long time is taken by each node to get synchronized. For instance, if 10 nodes are implemented in the network, they have to wait 100 seconds to setup the schedule and for 15 nodes the time rises to 150 seconds. Thus a longer time for stabilization takes place in proportion to the number of nodes in a network.

#### 3.2. ISOLATED SCHEDULE CLUSTER

The virtual clusters are formed when some nodes have a common schedule. In case of individual cluster having one neighbor node or more is overlapped with another cluster of different period get to be synchronized, by receiving a SYNC packet. However in some cases the clusters can fail to acknowledge each other and no communication will take place between

them, even though all the nodes are active. The S-MAC's solution to this problem is that all nodes work in the listen state for a given time.

In the actual implementation, all nodes sense SYNC packets and seek neighbor nodes while they are in the listen state for 10 seconds per 2 minutes. This phenomenon goes against the basic purpose of S-MAC, which states reduction of energy in the idle listen state. Thus a lot of energy is consumed while locating a hidden schedule cluster node.

### 3.3. DATA TRANSMISSION AFTER SYNCHRONIZATION OF BORDER NODE

The scheduling mechanism of S-MAC illustrates that each node has its own schedule generated randomly. It will then wait for a given time and if it fails to receive a SYNC packet, it will set its own schedule and broadcast its SYNC packet to the neighbor nodes. The neighbor nodes will receive the SYNC packet and use that schedule to get synchronized, but the whole network is not unified under same schedule. Therefore an independent schedule cluster having an independent schedule gets to be made and a node between heterogeneous schedules gets to receive SYNC packets which are different from one another and work as a border node [10].

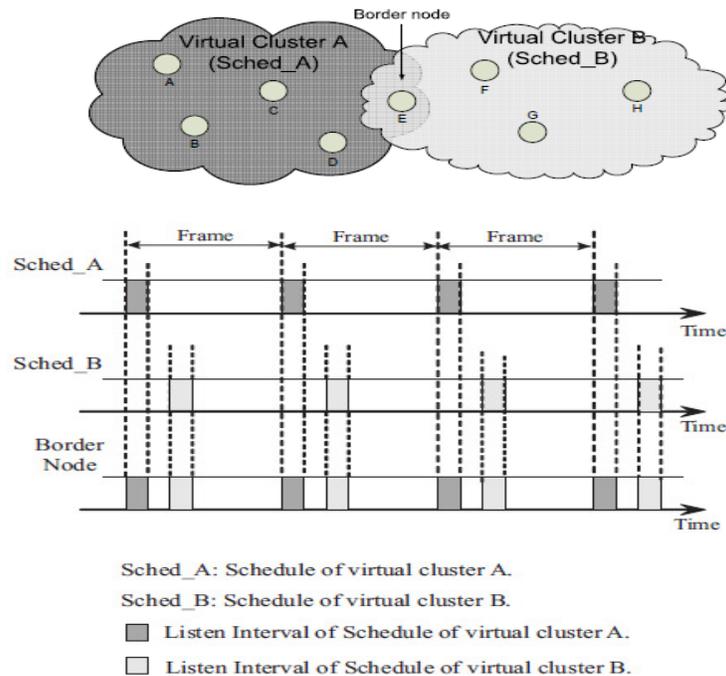


Fig.-3: Border Node adopting and handling both the schedules

As shown in Fig. 3, the border node adopts and handles both the schedules and creates a link between the virtual clusters. Therefore, it has to be in the listen state twice and the power consumption will be twice of a general node. If the border node dies out quickly, the clusters will not be able to interact with each other and no data transmission will take place between them. Thus the power consumption of the border node will increase in proportion to the number of different schedules adopted by it.

In the existing S-MAC code, this problem is somewhat minimized as the border node will only adopt one schedule depending upon the SYNC packet received first. However, this does not change the longevity of the border node as it stays in the listen state for a longer time.

### 3.4. ERROR DEVELOP DURING SYNCHRONIZATION

When the neighbor nodes synchronize their sleep schedule, the clock drift between each node can cause synchronization errors. This error takes place when SYNC packets are broadcasted periodically. At present, two techniques are in the S-MAC protocol code to minimize this error. First, all exchanged timestamps are relative rather than absolute. Second, the listen period is significantly longer than clock drift rates. The clock drift between two nodes does not exceed 0.2ms per second [10]. If we consider the listen time to be 0.5 second, as about 2500 second pass for a clock drift to take place. The listen time will increase the clock drift and consequently, increase the synchronization error. Thus, a time error resulting from a long clock drift is a critical factor in the actual implementation.

### 4. ALGORITHM DEVELOPMENT

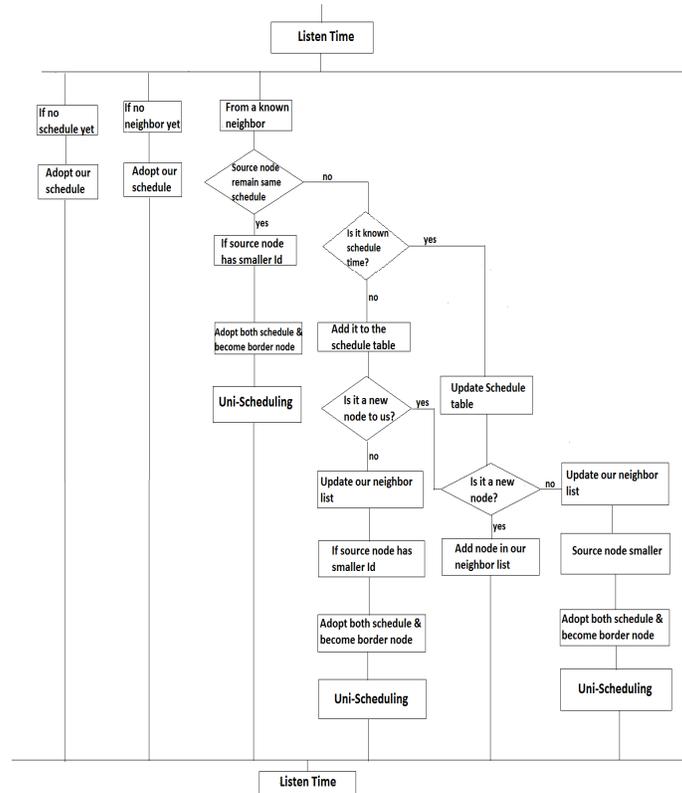


Fig.-4: Implementation of border node in the proposed algorithm

Our main focus will be on implementing a scheduling scheme so that all the nodes can communicate with each other at the same time. In order to synchronize the whole WSN quickly, our proposed algorithm is performed from the border nodes and more data packets can be transferred between the different virtual clusters

Firstly, if a node has its own schedule and receives a SYNC packet from a node with larger ID, it will not adopt the latter schedule. This is because, nodes with larger ID are newly joined in the network and will stay in the network for a shorter time. Secondly, the Neighbor list Table is updated after updating the Schedule Table in order to know the entire active neighbor surrounding a node. Thirdly, the broadcasting of *Uni-Scheduling packets* will be performed

after the construction of border nodes. The construction of a border node will take place during the listening period and will perform the following tasks as shown in Fig.4. Fourthly, the whole synchronization takes place during the listening period.

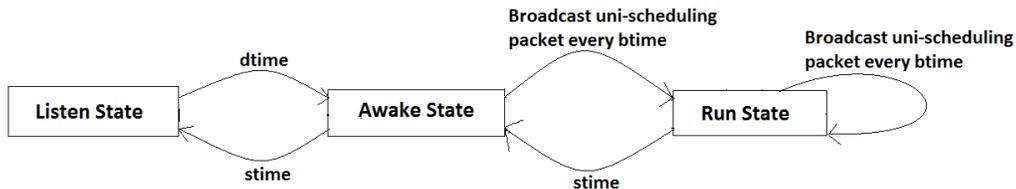


Fig.-5: State transition for Proposed Uni-Scheduling

After the border nodes are constructed, the next task is to set up a common schedule across the network so that all the nodes can be unified under a single schedule instead of heterogeneous schedules. The proposed unified scheduling algorithm is shown in Fig.5. Before going to analysis the proposed algorithm, several assumptions are taken which are as follows:

- i.) All the nodes should be fixed without any mobility.
- ii.) All the nodes have the same listen and sleep period throughout the simulation.
- iii.) The adaptive listening is not applied for this simulation.
- iv.) All the nodes are synchronized according to the schedule in the Uni-Scheduling packets.
- v.) The *Uni-Scheduling packets* are broadcasted periodically from the border nodes.
- vi.) If a node is running under uni-schedule and if it still receives any more *Uni-Scheduling packets*, it will discard all the latter packets.

The sensor nodes deal with *Uni-Scheduling packets* in the proposed algorithm and the transmission of these packets to all the nodes are done by using three types of timers. All the nodes go through three transition state before they are synchronized. The detail explanation of the different transition state and timers are given below

- The *LISTEN STATE* is the initial node state in which the node works in the listen state. The state is periodically changed to *AWAKE STATE* according to the *D-time*.
- In the *AWAKE STATE*, the border nodes adopt the uni-schedule as their primary schedule in the Schedule Table. Then *Uni-Scheduling packets* are broadcasted periodically to all the nodes in the network after every *B-time*. No data packets transmission is performed until the nodes receive *Uni-Scheduling packets*.
- After receiving *Uni-Scheduling packets* the nodes state change to the *RUN STATE*. In the *RUN STATE* the nodes will sense if their neighbors are synchronized under uni-schedule. After that data transmission will take place.
- The *D-timer* is used to change the transition state of the node periodically from the *LISTEN STATE* to the *AWAKE STATE*.
- The *B-timer* is used so that *Uni-Scheduling packets* are broadcasted periodically from the border nodes to all the other nodes in different virtual clusters.
- The *P-timer* is used to set a time of which the nodes waits to receive the subsequent *Uni-Scheduling packets* and if a node failed to receive any *Uni-Scheduling packets*, the node will move back to the previous state after the timer expire.

The *Uni-scheduling packet* as shown in Fig.6 contains a scheduling time and this packet is broadcasted from the border nodes to the whole sensor network. According to this signal, the nodes reset their timers and set the uni-schedule as their primary schedule in their Schedule Table. After that the nodes will be synchronized and data transmission begins.

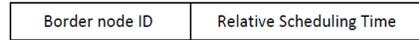


Fig.-6: The structure of Uni-Scheduling Packet

When all the nodes follow this proposed uni-scheduling scheme, the whole network gets to be synchronized under a single schedule.

## 5. ADVANTAGE OF THE PROPOSED ALGORITHM

Using this proposed Uni-Scheduling algorithm, the nodes do not have to take longer period of time for synchronization. This is because all the nodes will run under a single schedule and will transmit data at the same time. Thus the delay time eliminated as stabilization of schedule has been solved through the synchronization of the network. There is no generation of isolated schedule cluster due to Uni-Scheduling process. Even if, a new node wants to add to the network, it can receive the *Uni-Scheduling packets* in the *AWAKE STATE* and quickly change to the *RUN STATE* and the network can be expanded easily without any problem. Moreover, all the nodes in the network are unified under the same schedule and the energy consumption in the idle state is reduced.

As soon as, the nodes get synchronized after receiving the *Uni-Scheduling packets*, the data can be transmitted and received continuously. Moreover, the conventional border node concept does not exist no more and works similar to other nodes. In this way the energy consumption of the sensor nodes are almost similar.

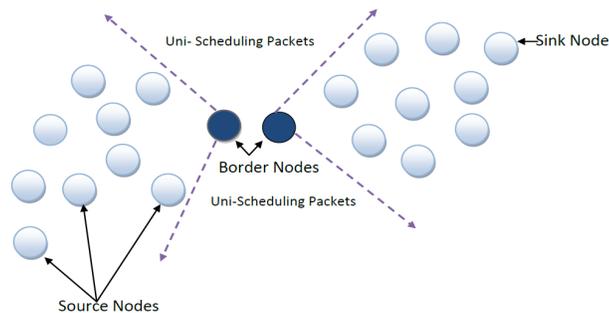


Fig.-7: Broadcasting *Uni-Scheduling Packets* to the network

When all the nodes get synchronized under a single schedule after receiving the *Uni-Scheduling packets*, the clock drift between two nodes are minimized. This clock drift error is reduced because all the nodes will perform their transmission at the same time. The synchronization error in the WSN is also reduced and this effect is clearly visible when the network size increases.

## 6. NETWORK SETUP

**Table-1: PARAMETERS FOR SIMULATION**

Parameter	Values
RTS/CTS/ACK Size	10 Bytes
Sync Packet Size	9 Bytes
Uni-Scheduling Packet Size	9 Bytes
Simulation area	1000m x 1000m
Routing Protocol	DSR
Listen Time	0.5 sec
D-Time, B-Time & P-Time	5 sec, 1sec & 200 sec
Simulation Time	1000 sec
Duty cycle	10%
Initial Energy	100 J
Tx and Rx Power	10 and 20

The proposed scheme is simulated in the Network Simulator-2 (ns-2) [13] under Linux environment. The platform for this simulation has provided us the perfect way to visualize the sensor nodes. All the necessary parameters for the wireless network and the energy model are included in the Tcl script. The TCP sink Agent is attached to the biggest node and TCP Agent is attached to all the remaining nodes in the network. The traffic application for this simulation is CBR traffic. Four kinds of topologies are considered for testing purpose and these are shown in Fig.8.

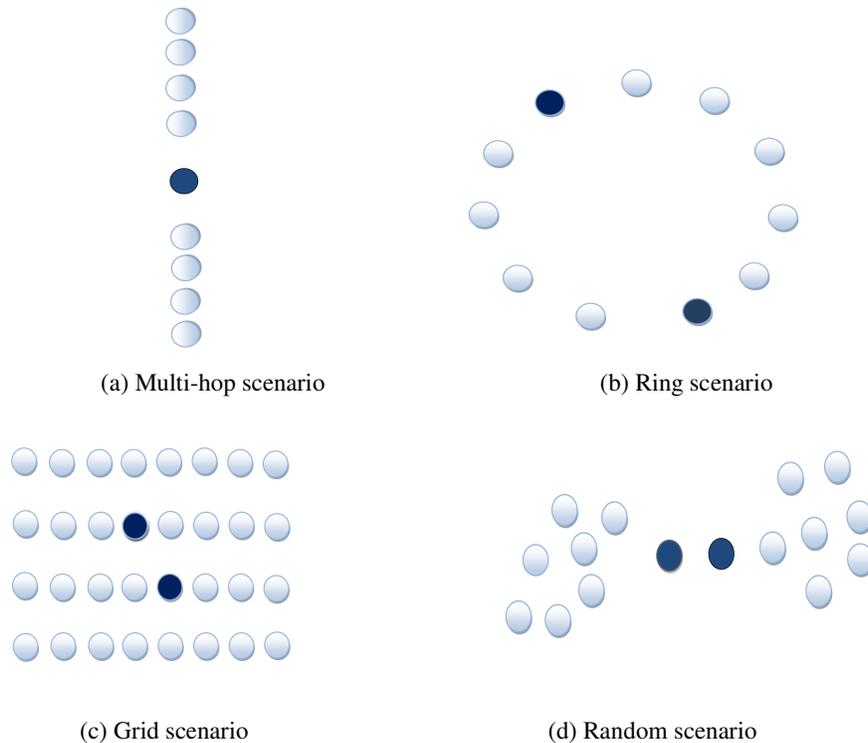
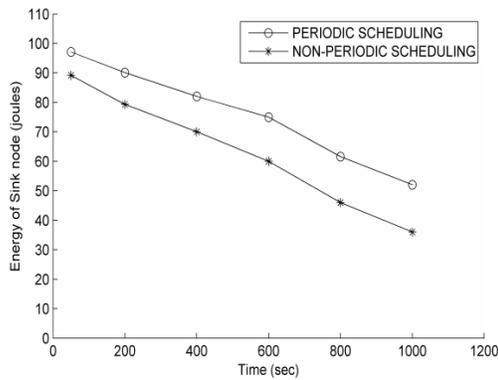


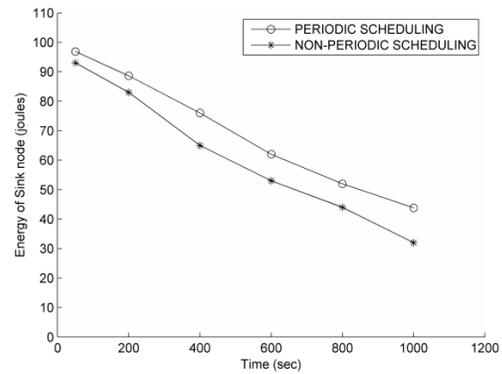
Fig.-8: Network Topologies

## 7. SIMULATED RESULTS AND ANALYSIS

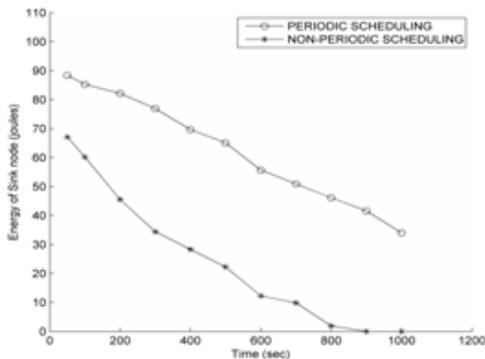
In this section, the simulation results for different topologies are presented. The topologies that are used consist of multi-hop network scenario, ring network scenario, grid network scenario and random network scenario. The number of nodes used for first two network scenarios are 10 nodes and the last two scenarios consist of 40 nodes.



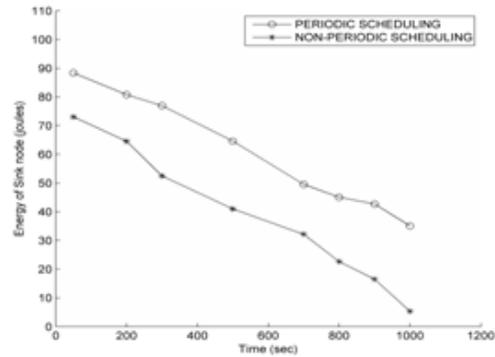
(a) Multi-hop scenario



(b) Ring scenario



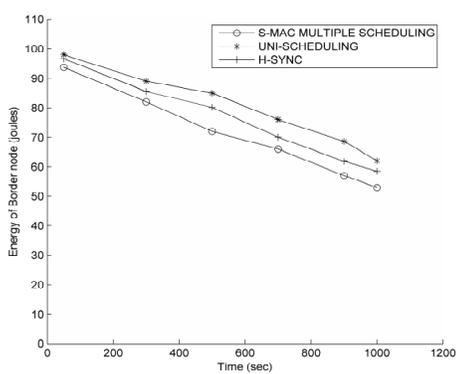
(c) Grid scenario



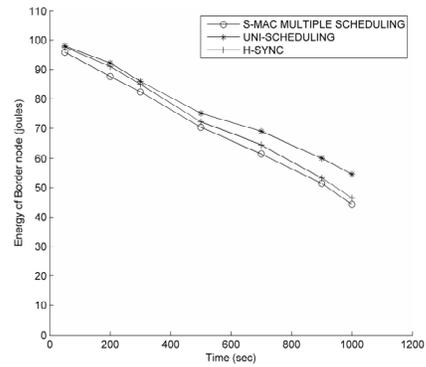
(d) Random scenario

Fig.-9: Energy consumption of the sink node

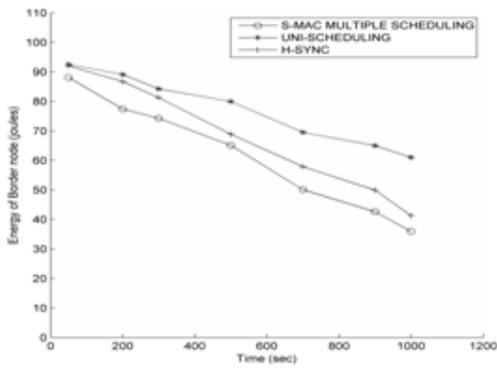
The above figures 9 (a), 9 (b), 9(c) and 9(d) show the result of energy consumption of the networks when the nodes adopt sleep/listen schedule and the other algorithms do not adopt sleep/listen schedule, while all other factors of the network remain unchanged. It is observed that if the network topologies are less complicated, energy consumption is less due to less overhearing and collisions. The simulated results prove that periodic scheduling saves large amount of energy in more complex networks.



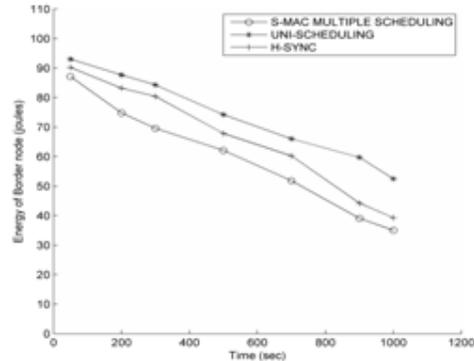
(a) Multi-hop scenario



(b) Ring scenario



(c) Grid scenario



(d) Random scenario

Fig.-10: Energy consumption of the border nodes

Fig.10 shows the energy consumption of the border nodes by applying different scheduling methods. In this experiment we have verified that our proposed Uni-Scheduling scheme performs more efficiently when compared with one of the unified scheduling methods called H-SYNC mentioned in paper [8] and S-MAC multiple scheduling. In smaller network topologies like the ring network, the energy consumption of all the scheduling methods is almost similar; no major changes can be distinguished from smaller network topologies. A significant result can be obtained from complicated networks, where the border nodes tend to lose more energy when they adopt H-SYNC and S-MAC scheduling. In the case of random scenario the difference in energy consumption between H-SYNC and S-MAC is 14.82% and between Uni-Scheduling and S-MAC is 69.67%. Thus we see that at the end of simulation Uni-scheduling saves 4.70 times more energy when compared with other scheduling methods. Therefore the life span of the border nodes increases and more data can be transferred between different virtual clusters.

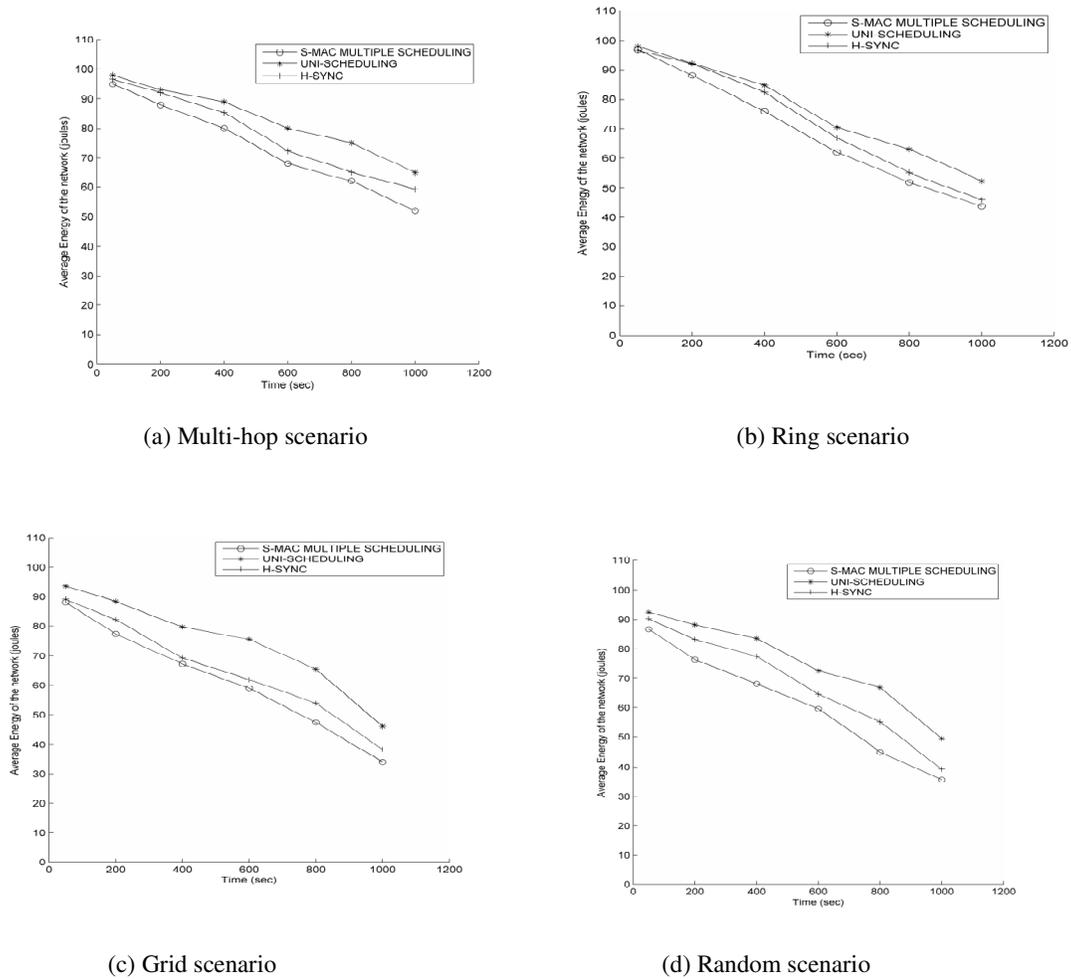


Fig.-11: Energy consumption of the network

From the previous simulated results shown in Fig.10, it has been noticed that the longevity of the border nodes for Uni-Scheduling are much longer than other scheduling methods. The graphs clearly show that as the network scenario changes, the energy consumption of the nodes increase significantly. In the case of grid and random scenarios, the energy consumption between Uni-Scheduling and H-SYNC is 11.91% and 18.73% respectively. Thus the Uni-Scheduling method is more appropriate than any other unified scheduling method. This also means that Uni-Scheduling is more appropriate for a complex WSNs.

## 8. CONCLUSION

The proposed scheme has unified the scheduling of S-MAC protocol and overcame the problems of diversified scheduling in wireless sensor networks. The border nodes in between virtual clusters broadcast Uni-Scheduling packets which synchronize the network under a single schedule. The simulated results showed that our proposed unified scheduling scheme performed much better than other scheduling methods. Thus, the lifetime of the network increases in compare to the existing S-MAC protocol.

## REFERENCES

1. S. Wen-Miao, L. Yan-ming and Z. Shu-e, "Research on SMAC protocol for WSN" in Proc. IEEE 4<sup>th</sup> International conference on Wireless Communication, Networking and Mobile Computing WiCOM '08, Oct 2008, pp.1-4.
2. B Paul and M. A. Matin, "Optimal geometrical sink location estimation for two-tiered wireless sensor networks" IET Wirel. Sens. Syst., 2011, vol.1, no. 2, pp.74-84
3. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, IEEE Std 802.11-1999 edition.
4. Wei Ye, John Heidemann and Deborah Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks", in Proc. IEEE INFOCOM, New York, pp. 1567-1576, June 2002.
5. Yan-xiao Li, Hao-Shan Shi, and Shui-Ping Zhang, "An Energy-Efficient MAC protocol for Wireless Sensor Network", 3rd International Conference on Advanced Computer Theory and Engineering(ICACTE), Chengdu, Vol. 6, No. 4, pp. 619-623 Aug. 2010.
6. Woonsik Lee, Minh Viet Nguyen, Arabinda Verma, Student Member, IEEE, and Hwang Soo Lee, Member, IEEE, "Schedule Unifying Algorithm Extending Network Lifetime in S-MAC Based Wireless Sensor Networks", IEEE Transactions on Wireless Communication, Vol. 8, No. 9, pp. 4375-4379, Sept. 2009.
7. Elhadi M. Shakshuki, Tarek R. Sheltami, Haroon Malik, Chuoxian Yang, "Investigation and Implementation of Border Nodes in S-MAC", International Conference on Network-Based Information Systems, Indianapolis, Indiana, pp 350-357, Aug. 2009.
8. Tae-Seok Lee, Yuan Yang, Ki-Jeong Shin and Myong-Soon Park, "An Energy-Efficient Uni-Scheduling based on S-MAC in Wireless Sensor Network", pp 293-304, 2005.
9. Yuan Yang, Fu Zhen, Teo-Seok Lee and Myong-Soon Park, "TASL: A Traffic-Adapted Sleep/Listening MAC Protocol for Wireless Sensor Network", International Journal of Information Processing Systems, Vol.2, No.1, pp. 39-43, March 2006.
10. Wei Ye, John Heidemann and Deborah Estrin, "Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks", IEEE/ACM Transactions on Networking, VOL. 12, No. 3, pp. 493-506, June 2004.
11. Baoshan Zhang, Xiangdong Wang, Shujiang Li, Leishu Dong, "An Adaptive energy-efficient Medium Access Control Protocol For Wireless Sensor Networks", IEEE 2009 Fifth International Conference on Mobile Ad-hoc and Sensor Networks, Fujian, Wu Yi Mountain, China, Vol. 6, No. 9, pp. 350-357, Dec. 2009.
12. Behnam Dezfouli, Marjan Radi and Shukor Abd Razak, "A Cross-Layer Approach For Minimizing Interference And Latency Of Medium Access In Wireless Sensor Networks", International Journal of Computer Networks & Communication, Vol.2, No.4, pp. 126-142, July 2010
13. <http://www.isi.edu/nsnam/ns/>