# AN ADAPTIVE ENERGY EFFICIENT MAC PROTOCOL FOR WIRELESS SENSOR NETWORKS

Lakshmanan M.<sup>1</sup>, Noor Mohammed V.<sup>2</sup> E-mail : <sup>1</sup>tmlakshmanan@gmail.com

### Abstract

Sensor networks are deployed in remote locations with limited processor capabilities, memory capacities, and battery supplies. Wireless Sensor Networks (WSN) detects environmental information with sensors in remote settings. One problem facing WSNs is the inability to resupply power to these energy-constrained devices due to their remoteness. Therefore to extend a WSN's effectiveness, the lifetime of the network must be increased by making them as energy efficient as possible. An energy-efficient medium access control (MAC) can boost a WSN's lifetime. This paper focuses on a protocol stack solution that deals with MAC layer, that minimizes the energy consumption and delay required to transmit packets across the network. It is based on Sensor Medium Access Control (S-MAC) called Adaptive SMAC protocol designed for sensor networks. It enables low duty cycle operation in a multi-hop network and common sleep schedules to reduce control overhead and enable traffic adaptive wakeup. To reduce control overhead and latency, introduces coordinated sleeping among neighboring nodes. It is a contention based protocol based on CSMA/CA mechanism. This protocol is simulated in NS-2 and performance evaluated using various topologies under various traffic conditions. In addition with this we tried to improve the energy efficiency of Adaptive SMAC with the help of a new design called Adaptive Cross MAC protocol

**Keywords:** Wireless Sensor Networks, Medium Access Control, Sensor MAC, Timeout MAC, Distributed Coordination Function, Carrier Sense Multiple Access /Collision Avoidance, Carrier Sense Multiple Access /Collision Detection

# I. INTRODUCTION

Wireless Sensor Network WSN) is a distributed sensing network comprised of thousands, or even tens of thousands small devices that sense, collect and disseminate information about the environment, and communicate that information back to some location for processing. With each node equipped with radios of wireless communication capability and sensors that can sense certain physical phenomena such as acoustics, light, temperature, humidity and vibrations. Lakshmanan. M is with the School of Electrical Sciences VIT University, Vellore, Tamilnadu 632014 India. (Phone: 91-97892-31031; E-mail: tmlakshmanan@gmail.com).

Noor Mohammed. V is with the School of Electrical Sciences VIT University, Vellore, Tamilnadu 632014 India. (Phone: 91-98429-95074; E-mail: noorb4u@gmail.com). Wireless sensor networks enable a wide range of applications, such as target tracking, habitat sensing and fire detection. The capabilities of sensor nodes are very different from traditional nodes in computer networks. These devices have very limited energy, processing power, storage, and communication range and data rate. Traditional wireless networks do not manage energy efficiently. WSN collects information until it runs out of power, which is currently a critical issue. Due to the inability to resupply the WSN with power after deployment, the lifetime of the network must be extended as much as possible to increase its effectiveness in information collection. In addition, many techniques used in wireless networks do not scale well and therefore would not be appropriate for a WSN which may need hundreds or even

thousands of nodes to be useful. Limiting the number of nodes per unit area limits the resolution of any data the sensor nodes are designed to capture. A small number of nodes in a large area do not have the same resolution as a large number of nodes in the same area. Furthermore, networks that do not scale well tend to consume more energy due to their inefficient use of resources. Such networks will consist of large numbers of distributed nodes that organize themselves into a multi-hop wireless network. Each node has one or more sensors, embedded processors, and low-power radios, and is normally battery operated. Typically, these nodes coordinate to perform a common task. The primary goal of this paper is to modify en existing MAC protocol to make it more energy aware MAC protocol [3] for Wireless Sensor Networks, which maximizing the network life time and reduce the latency under various traffic situations. Next is analyzing the performance based on energy and latency. And the purpose of this paper is to evaluate the importance of MAC layer and investigates medium access control protocol which improves on existing sensor S-MAC protocols [3] by not only creating additional opportunities to place the sensor platform into lower power saving modes,[9] but also by establishing a traffic rhythm which extends the sleep duration to minimize power mode transition.

### **II. DESIGN PRINCIPLES**

WSNs are a logical extension of wireless networks, though with different priorities on performance such as throughput, latency, bandwidth, and energy consumption. One of the key differences between regular wireless networks and WSNs are their limited lifetimes. Normally, nodes in a WSN are powered by batteries and deployed to remote locations where it is not possible to change the battery. Such networks are deployed ad hoc with a limited range of

communication implying multi-hop routing is required transfer data across the network. Since the energy supply is limited, energy consumption is one of the primary metrics of interest when designing a WSN. Many WSNs use an ad hoc configuration. In an infrastructure type of architecture, all traffic flows through a set of access points. The capability to re-supply the access points with power would not exist and since the access points would see the most traffic, they would be the first nodes to run out of energy. New nodes could be elected as access points, but this uneven distribution of energy consumption would likely partition the network and limit its functionality. An ad hoc approach more evenly distributes traffic load over all nodes. Therefore, the energy consumption is more uniform which would prevent network partitions. 2.1 Analytical WSN MAC Protocol Models

A) Latency analysis We analyze the latency of different MAC protocols in the simple case that the traffic load is very light, e.g., only one packet is moving through the network, so that there is no queueing delay and backoff delay. We further assume that the propagation delay and the processing delay can be ignored. In this case, only carrier sense delay, transmission delay and sleep delay are taken into account.

B) MAC protocol without sleeping

N -- Number of hops

tcs,n -- Carrier sense delay at hop 'n'.

(Its mean value is t<sub>cs</sub>)

 $t_{k}$  -- Transmission delay(Transmission delay is fixed if the packet length is fixed) The entire latency over N hops is,

$$D(N) = \sum_{n=1}^{N} (t_{cs}, +t_{tx})$$
(1)

The average latency over hops in the MAC is,

$$E[D(N)] = N(t_{cs,n} + t_{ts})$$
 (2)

It shows that multi hop latency linearly increases with number of hop

# C) S-MAC protocol with periodic sleeping

To achieve energy conservation, sensing nodes employing the MAC sleeps periodically to reduce duty cycle and minimize idle listening.

The delay at hop, 
$$n' = Dn = t_{s,n} + t_{cs,n} + t_{tx}$$
 (3)

Frame length = 
$$Tf = t_{s,n} + t_{cs,n-1} + t_{tx}$$
 (4)

Sleep delay at hop n =  $t_{s,n} = T_f - (t_{cs,n-1} + t_{tx})$  (5)

Substitute equation (5) in (3)

$$Dn = T_{f} + t_{cs,n} - t_{cs,n-1}$$
(6)

Overall delay of a packet over N hops

$$= D(N) = D_1 + \sum_{n=2}^{N} D_n$$
 (7)

Substitute equation. (6) in (7)

$$= t_{s,l} + t_{cs,N} + t_{tx} + \sum_{n=2}^{N} (T_f + t_{cs,n} - t_{cs,n-1})$$
(8)

Latency over N hops is,

$$D(N) = t_{s,1} + (N_{-1})T_{f} + t_{cs,N} + t_{tx}$$
(9)

The average latency over hops in the MAC,

$$E[D(N)] = NT_{f} - T_{f} / 2 + t_{cs} + t_{tx}$$
(10)

It shows that multi hop latency also linearly increases with the numer of hops in S-MAC when each node strictly follows its sleep schedule.

B) Energy Latency Tradeoffs in Medium Access Control Previous works have identified idle listening as a major source of energy wastage. To design an energy efficient MAC, [3] it is essential to turn the radio off when a node does not participate in any data delivery. However, a node that is sleeping is no longer part of the network, and thus cannot help to deliver the sensor data from its neighbors to its destination. When a node has a packet for its neighbor who is in asleep, it has to wait until its neighbor is active. This creates a fundamental trade-off between energy and latency.

### **III. DESIGN METHODOLOGIES**

A. Adaptive – Sensor Medium Access Control [A-SMAC] [1]

#### i) Periodic Listen and Sleep

In many sensor network applications, nodes are idle for long time if no sensing event happens. Given the fact that the data rate is very low during this period, it is not necessary to keep nodes listening all the time. Adaptive S-MAC reduces the listen time by putting nodes into periodic sleep state. Each node sleeps for some time, and then wakes up and listens to see if any other node wants to talk to it. During sleeping, the node turns off its radio, and sets a timer to awake it later. All nodes are free to choose their own listen/sleep schedules. However, to reduce control overhead, we prefer neighboring nodes to synchronize together. That is, they listen at the same time and go to sleep at the same time. It should be noticed that not all neighboring nodes can synchronize together in a multihop network.

### ii) Collision Avoidance

If multiple neighbors want to talk to a node at the same time, they will try to send when the node starts listening. In this case, they need to contend for the medium. Among contention protocols, the 802.11 does a very good job on collision avoidance. Adaptive S-MAC follows similar procedures (CSMA/CA), [5] including virtual and physical carrier sense, and the RTS/CTS exchange for the hidden terminal problem. All senders perform carrier sense before initiating a transmission. If a node fails to get the medium, it goes to sleep and wakes up when the receiver is free and listening again. Broadcast packets are sent without using RTS/CTS. Unicast packets follow the sequence of RTS/CTS/DATA/ACK between the sender and the receiver. After the successful exchange of RTS and CTS, the two nodes will use their normal sleep time for data packet transmission. They do not follow their sleep schedules until they finish the transmission. With the lowduty-cycle operation and the contention mechanism during each listen interval, S-MAC effectively addresses the energy waste due to idle listening and collisions [7].

# iii) Coordinated adaptive sleeping

Periodic sleeping effectively reduces energy waste on idle listening. In Adaptive S-MAC,[1] nodes coordinate their sleep schedules rather than randomly sleep on their own. This section details the procedures that all nodes follow to set up and maintain their schedules. It also presents a technique to reduce latency due to the periodic sleep on each node.

# iv) Choosing and Maintaining Schedules

Before each node starts its periodic listen and sleep, it needs to choose a schedule and exchange it with its neighbors. Each node maintains a schedule table that stores the schedules of all its known neighbors. It follows the algorithm to choose its schedule and establish its schedule table. Algorithm:

 A node first listens for a fixed amount of time, which is at least the synchronization period. If it does not hear a schedule from another node, it immediately chooses its own schedule and starts to follow it. Meanwhile, the node tries to announce the schedule by broadcasting a SYNC packet. Broadcasting a SYNC packet follows the normal contention procedure. The randomized carrier sense time reduces the chance of collisions on SYNC packets.

- If the node receives a schedule from a neighbor before choosing or announcing its own schedule, it follows that schedule by setting its schedule to be the same. Then the node will try to announce its schedule at its next scheduled listen time.
- 3) There are two cases if a node receives a different schedule after it chooses and announces its own schedule. If the node has no other neighbors, it will discard its current schedule and follow the new one. If the node already follows a schedule with one or more neighbors, it adopts both schedules by waking up at the listen intervals of the two schedules.

### V. Maintaining Synchronization

As mentioned earlier, schedule updating is accomplished by sending a SYNC packet. The SYNC packet is very short, and includes the address of the sender and the time of its next sleep. The next sleep time is relative to the moment that the sender starts transmitting the SYNC packet. When a receiver gets the time from the SYNC packet it subtracts the packet transmission time and use the new value to adjust its timer.



Fig. 1. The timing relationship of three possible situations that a sender transmits to a receiver. Sender 1 only sends a SYNC packet. Sender 2 only sends a unicast data packet. Sender 3 sends both SYNC and a data packet.

In order for a node to receive both SYNC packets and data packets, divide its listen interval into two parts. The first one is for SYNC packets, and the second one is for data packets, as shown in fig.1 Each part has a contention window with many time slots for senders to perform carrier sense. For example, if a sender wants to send a SYNC packet, it starts carrier sense when the receiver begins listening. It randomly selects a time slot to finish its carrier sense. If it has not detected any transmission by the end of that time slot, it wins the contention and starts sending its SYNC packet. The same procedure is followed when sending data packets.

### vi) Adaptive Listening

S-MAC proposes an important technique, called adaptive listen, to improve the latency caused by the periodic sleep of each node in a multihop network. The basic idea is to let the node who overhears its neighbor's transmissions (ideally only RTS or CTS) wake up for a short period of time at the end of the transmission. In this way, if the node is the next-hop node, its neighbor is able to immediately pass the data to it instead of waiting for its scheduled listen time. If the node does not receive anything during the adaptive listening, it will go back to sleep until its next scheduled listen time.

## vii) Overhearing avoidance

Collision avoidance is a basic task of MAC protocols. Adaptive S-MAC[1] adopts a contention-based scheme. It is common that any packet transmitted by a node is received by all its neighbors even though only one of them is the intended receiver. Overhearing makes contentionbased protocols less efficient in energy than TDMA protocols. In 802.11 each node keeps listening to all transmissions from its neighbors in order to perform effective virtual carrier sense. As a result, each node overhears many packets that are not directed to itself. It is a significant waste of energy, especially when node density is high and traffic load is heavy. Inspired by PAMAS [6], Adaptive S-MAC tries to avoid overhearing by letting interfering nodes go to sleep after they hear an RTS or CTS packet. Since DATA packets are normally much longer than control packets, the approach prevents neighboring nodes from overhearing long DATA packets and following ACKs.

### viii) Latency analysis

Fig.2 shows a part of a multi-hop network, the three hops are denoted as "n" to "(n+2)". Assume: all nodes follow the same sleep schedule.



Fig. 2. multi-hop latency analysis (Adaptive listen can reduce sleep latency by at least half)

The delay at hop n =  $D_n = t_{s,n} + t_{cs,n} + t_{tx}$  (11)

The delay at hop 
$$(n+1) = D_{n+1} = t_{cs,N+1} + t_{tx}$$
 (12)

The delay at hop  $(n+2) = D_{n+2} = t_{s,n+2} + t_{cs,n+2} + t_{tx}$  (13)

The entire latency over N hops is,

$$D(N) = t_{s,1} + (N/2 - 1)T_f + t_{cs,N-1} + t_{cs,N} + 2t_t x$$
(14)

The average latency over N hops,

$$E[D(N)] = NT_{f}/2 + 2t_{cs} + 2t_{tx} - Tf/2$$
(15)

We can see that the average latency in S-MAC [3] with adaptive listen still linearly increases with the number of hops. Now the slope of the line is  $T_r/2$  compared with that of no adaptive listen equation (10), it is reduced by half.

# B. Adaptive Cross Layer – SMAC [2]

It is based on a layered design approach, which means that previous design are focused on designing optimal strategies for "single" layer only. In this, we take an alternative approach i.e., a cross-layer design.. In this approach, the interactions between MAC and Routing layers are fully exploited to achieve energy efficiency for wireless sensor networks. More precisely, in the proposed routing information at the network layer is utilized for the MAC layer such that it can maximize sleep duration of each node.

A design goal of MAC-CROSS is to minimize energy consumption by continuously turning off the radio interface of unnecessary nodes that are not included in the routing path. In this method, categorize nodes into three types depending upon the state defined by data transmission: Communicating Parties, Upcoming communicating Parties and Third Parties. A state may dynamically change whenever data traffic is transmitted.

- Communicating Parties (CP): Any node currently participating in the actual data transmission. (Like nodes A and B in fig. 3).
- Upcoming communicating Parties (UP): Any node to be involved in the actual data transmission. (Like node C in fig. 3).
- Third Parties (TP): Any nodes that are not included on a routing path and hence not involved in the actual data transmission at all. (like nodes D-K in fig. 3)

Now, we explain the proposed MAC-CROSS scheme with the help of the following example. Remind that, in fig. 3 illustrating the main drawback of the adaptive S-MAC, all nodes are being awake when their NAV timer expire and consume unnecessary energy. The proposed MAC-CROSS can overcome this problem



Fig. 3. Adaptive SMAC operations

Fig. 4 shows the same scenario as fig. 3. Thus, with the MAC-CROSS, only a few nodes concerned of the actual data transmission (i.e., the necessary UP nodes like nodes B and C in fig. 3) are asked to wake up, while other TP nodes can continuously remain in their sleep modes



Fig. 4. MAC-CROSS operation

### **IV. IMPLEMENTATION**

### A. Adaptive S-MAC

The algorithm has many parameters and features. In this section, we will try to study the performance of the algorithm with respect to network dynamics (such as Energy changes in the nodes, latency). The simulation results show the potential of the algorithm in achieving fair energy distribution across the network nodes. We conducted separate simulations for measurement of energy consumption of the nodes and measurement of end to end delay. In addition, we evaluated the performance of the protocol using different topologies under different traffic situations. The results show the ability of the algorithm.

#### I) Measurement of Energy Consumption

To measure the energy consumption on the radio, we measured the amount energy consumption of each node through the trace graph results after running the simulation as per the scenario. With the help of trace files we will get all information about energy consumption of each node for each purpose on a specific time. Result as spent in different modes: sleep, idle, receiving or transmitting. The energy consumption in each mode is then calculated by multiplying the time with the required power to operate the radio in that mode. We measure energy indirectly (trace file) because of the difficulty in directly observing current draw on physically small, low power nodes. We conducted test on two hop network, multi-hop linear networks under different traffic loads.

### ii) Tests on a Two-Hop Network

These tests are based on [1]. The topology is a twohop network with two sources and two sinks, as shown in Fig. 5. Packets from source A flow through node C and end at sink D, while those from B also pass through C but end at E. We change the traffic load by varying the inter-arrival period of messages. If the message inter-arrival period is 5(s), a message is generated every 5(s) by each source node. In this experiment, the message inter-arrival period varies from 1 to 10(s). For the highest rate with a 1(s) interarrival time, the wireless channel is nearly fully utilized due to its low bandwidth. For each traffic pattern, we have done ten independent tests when using different MAC protocols.





In each test, each source periodically generates ten messages,. Thus, in each experiment, there are 200 cbr data packets (each 40 bytes) to be passed from their sources to their sinks. We measured the energy consumption of the radio on each node to pass the fixed amount of data. In the Adaptive S-MAC module with periodic sleep, each node is configured to operate in 50% duty cycle.

### iii) Tests on a linear Multi-hop Network

These multi-hop experiments are based on [1]. The topology is a linear network with 8 nodes, and 7 hops between source to destination. as shown in fig. 6. The first node is the source, and the last node is the sink.



# Fig. 6. Topology 2: seven -hop linear network with one source and one sink.

As before, we vary the traffic load by changing the packet inter-arrival time on the source node. This time the packet inter-arrival time changes from 0 to 10(s), where 0(s) means all the packets are generated and queued at the same time on the source node. Under each traffic condition, the test is independently carried out 11 times. In each test, the source node sends 20 messages that are each 100 bytes long. In the Adaptive S-MAC module with periodic sleep, each node is configured to operate in 10% duty cycle. We tested the performance of the protocol under ten hop networks, and simulation procedures are similar as seven hop linear multi-hop network.

### iv) Tests on a Multi-hop Network (25 nodes)

The topology is a multi-hop network with 25 nodes and six hops between source to destination. As shown in fig. 7, the first node is the source, and the last node is the sink. As before, we vary the traffic load by changing the packet inter-arrival time on the source node. This time the packet inter-arrival time changes from 0 to 10(s), where 0(s) means all the packets are generated and queued at the same time on the source node. Under each traffic condition, the test is independently carried out 11 times. In each test, the source node sends 20 messages that are each 100 bytes long. In the Adaptive S-MAC module with periodic sleep, each node is configured to operate in 10% duty cycle.



# Fig. 7. Topology 3: 25 node network with one source and one sink.

### v) Measurement of End-to-End Latency

Since S-MAC makes the tradeoff of latency for energy savings, we expect that it can have longer latency in a multi-hop network due to the periodic sleep on each node. Adaptive listen is designed to minimize such additional latency. We use seven-hop network topology in fig. 6 to measure the end-to-end latency of Adaptive S-MAC. We consider two extreme traffic conditions,

# ✤ lowest traffic load

highest traffic load

Under the lowest traffic load, the second message is generated on the source node after the first one is received by the sink. To do this, a coordinating node is placed near the sink. When it hears that the sink receives the message, it signals the source directly by sending at the highest power. In this traffic load, there is no queuing delay on each node. Compared with the MAC without sleep, the extra delay is only caused by the periodic sleep on each node.

Under the highest traffic load, all messages are generated and queued on the source node at the same time. So there is a maximum queuing delay on each node including the source node. In both cases, we begin measuring the latency of each message from the time it is generated on the source node.

In each test, the source node generates a cbr message, size 100 bytes. For the lowest traffic load, the packet generation time is uniformly distributed within one frame. The test is repeated ten times under both the lowest and the highest traffic load. The measurement is on the same. Adaptive S-MAC modes as we used in measuring the energy consumption in the same seven-hop network. In addition we compared the Adaptive SMAC with basic IEEE802.11 MAC protocol [9] using the same network and traffic conditions.

# 13

# V. RESULTS AND DISCUSSION

### A. Measurement of Energy Consumption

### i) Tests on a Two-Hop Network

ii) A-SMAC with periodic sleep

In each test, each source periodically generates 100 small data packets (40 bytes each). Thus, in each experiment, there are 200 data packets to be passed from their sources to their sinks. We measured the energy consumption of the radio on each node to pass the fixed amount of data. The actual time to finish the transmission is different for each MAC module. In the Adaptive SMAC module with periodic sleep, each node is configured to operate in 50% duty cycle. Message inter arrival period is varying from 1(s) to 9(s), and total ten different tests conducted for this evaluation. Simulation results are given below in the table I.

# Table 1. Energy values of each source node- Adaptive SMAC with periodic sleep

Message inter arrival period (s)	Aggregate energy(J)
1	0.491
2	0.462
3	0.471
4	0.480
5	0.490
6	0.503
7	0.509
8	0.519
9	0.5275

ii) A-SMAC without periodic sleep

Each test conducted similar as the above experiments. Each source periodically generates 100 small data packets (40 bytes each). There are 200 data packets to be passed from their sources to their sinks In the Adaptive SMAC module without periodic sleep, each node is working with out any sleep cycle. Table II shows the simulation results.

Message inter arrival period (s)	Aggregate energy(J)
1	0.435
2	0.4375
3	0.461
4	0.476
5	0.4965
б	0.518
7	0.536
8	0.560
9	0.5755



# Fig. 8. Mean energy consumption of source nodes (two hop network – X graph plot)



(two hop network - MATLAB plot)

Fig. 8 and fig. 9 shows the measured average energy consumption on the source nodes A and B. The traffic is heavy when the message inter-arrival time is less than

 Table 2. Energy values of each source node 

 Adaptive SMAC with out periodic sleep

4(s). In this case, ASMAC with periodic sleep uses more energy than with out sleep. Since idle listening rarely happens, energy savings from periodic sleeping is very limited. ASMAC achieves good energy savings in low traffic situations. i.e., inter-arrival period is more than 5(s).

- ii) Tests on a linear Multi-hop Network
- ii-a) Seven hop linear network

As before, we vary the traffic load by changing the packet inter-arrival time on the source node. This time the packet inter-arrival time changes from 0 to 10(s), where 0(s) means all the packets are generated and queued at the same time on the source node. Under each traffic condition, the test is independently carried out five times. In each test, the source node sends 20 messages that are each 100 bytes long. Table III shows the simulation results.

# Table 3. Aggregate energy consumption values of node under varying inter arrival period for seven hop linear network

Message inter arrival period (s)	Aggregate energy(J)
0	1.811
1	1.564
2	1.86
3	2.1247
4	2.486
5	2.7891
б	3.0845
7	3.3971
8	3.706
9	4.0055
10	4.126

Simulation Screen shots



Fig. 10. Nam out (seven hop linear multi-hop network)





Fig. 11 and shows the measured energy consumption on radios in the entire network to pass the fixed amount of data from the source to the sink. The result conforms to that we have obtained on the two-hop network. S-MAC with periodic sleep achieves substantial energy savings over the MAC without periodic sleep in the multi hop network, especially when traffic load is light.



Fig. 12. Aggregate energy consumption (MATLAB plot)

B. Measurement of end to end delay

I) Lowest traffic - Tests on a linear Multi-hop Network (7hops)

We consider two extreme traffic conditions, the lowest traffic load and highest traffic load. Compared with 802.11 MAC without sleep, the extra delay is only caused by the periodic sleep on each node.



Fig. 13. Message latency under lower traffic load (MATLAB plot)



Fig. 14. Message latency under lower traffic load (X graph plot)

Fig. 13 shows the measured mean message latency on each hop in the lowest traffic load. In the S-MAC mode, the latency increases linearly with the number of hops. However, ASMAC has much higher latency than the 802.11. The reason is that each message has to wait for one sleep cycle on each hop.

# ii) High traffic load

The latency of S-MAC with adaptive listen, by comparison, differ is small to that of the MAC without any periodic sleep, because adaptive listening often allows S-MAC to immediately send a message to the next hop. However, it does not reach the shortest latency in the MAC of fully active mode. In this case interarrival period is 45 (ms), and initial CBR packets are dropped. Delay of the packet in each hop is taken from the trace values of the ns-2 simulation.



Fig. 15. Message latency under higher traffic load (MATLAB plot)



Fig. 16. Message latency under higher traffic load (X graph plot)

Fig. 15 shows the mean message latency on each hop in the highest traffic load. Again, the low-duty-cycle mode without adaptive listen has the highest latency. With adaptive listen, the latency is high comparatively with 802.11 MAC

# C. Tests on multi-hop network (25 nodes)

We evaluated the performance of the protocol in case of twenty five node dense network. Area of simulation is 120(sqm), and tested the performance matrices as per the seven hop network. Communication range of each node is fixed in 40(m).

# I) Measurement of energy consumption

Each test conducted similar as the above experiments. Source periodically generates 20 small data packets (100 bytes each). In the Adaptive SMAC module without periodic sleep, each node is working with out any sleep cycle. Result shows the protocol is consuming more energy in the case of dense network.





 ii) Measurement of end to end delay - Low traffic vs High traffic

We evaluated the performance of ASMAC in this dense network, and compared the result of latency under low traffic and high traffic situations. We fixed inter arrival period 100(ms) for low traffic and 6000(ms) for high traffic.



Fig. 18. Message latency (low traffic Vs high traffic) of 25 node network.

# **VI. CONCLUSION**

In this paper we implemented a MAC protocol for wireless sensor network namely, Adaptive Sensor MAC [A-SMAC]. Simulation have been conducted in different network topologies like two hop network, linear multi-hop network (seven hop and ten hop) and dense multi-hop network containing 25 nodes using ns-2 simulator, and evaluated the performance of the protocol under low and higher traffic conditions. We compared the performance of the protocol. Performance evaluation matrices are energy and delay. The simulation result shows that the performance of Adaptive SMAC is satisfactory under both low and high traffic conditions. But we identified the problems of A-SMAC in terms of energy efficiency. And we tried for the new cross layer architecture design to improve energy efficiency of A-SMAC.

# REFERENCES

 W. Ye, J, Heidemann, and D. Estrin, 2004, "Medium Access Control With Coordinated Adaptive Sleeping for Wireless Sensor Networks", IEEE/ACM Transactions on Networking, vol. 12, no. 3.

- [2] Changsu Suh, Young Bae Ko, and Dong Min Son, 2006, " An Energy Efficient Cross - Layer MAC Protocol for Wireless Sensor Networks", APWeb, Springer - Verlag Berlin Heidelberg, LNCS 3842, pp. 410–419
- [3] W. Ye, J. Heidemann, and D. Estrin, 2002, "An energy-efficient mac protocol for wireless sensor networks", in Proc. IEEE INFOCOM, New York, NY, , pp. 1567–1576.
- [4] Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification, 1999 edition IEEE Std. 802.11.
- [5] A.Woo and D. Culler, 2001, "A transmission control scheme for media access in sensor networks," in Proc. ACM/IEEE Int. Conf. Mobile Computing and Networking, Rome, Italy, pp. 221–235
- [6] S. Singh and C. S. Raghavendra, 1988, "PAMAS: Power aware multi-access protocol with signalling for ad hoc networks", ACM Comput. Commun. Rev., vol. 28, no. 3, pp. 5–26.
- [7] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang,1994. "MACAW: A media access protocol for wireless lans", in Proc. ACM SIGCOMM, London, U.K., pp.212–225.
- [8] T. S. Rappaport, Wireless Communications, Principles and Practice, Englewood Cliffs, NJ: Prentice-Hall, 1996
- [9] Y.-C. Tseng, C.-S. Hsu, and T.-Y. Hsieh, 2002, "Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks", in Proc. IEEE INFOCOM, New York, NY, pp. 200–209.
- [10] NS website: http://www.isi.edu/nsnam/ns/