

Survey and Analysis of Medium Access Control Protocols for Wireless Sensor Network

Ms. Suchita S. Jogi¹, Prof. Amarsinh Vidhate²

^{1,2} Department of Computer Engineering, Ramarao Adik Institute of Technology, Nerul, Navi Mumbai, India

Abstract: *Wireless sensor networks are appealing to researchers due to their wide range of application in several fields such as military settings, critical infrastructure protection, target detection and tracking, environmental monitoring, industrial process monitoring etc. Communication among wireless sensor nodes is usually achieved by means of a unique channel. It is the characteristic of this channel that only a single node can transmit a message at any given time. Therefore, shared access of the channel requires the establishment of a MAC protocol among the sensor nodes. The objective of the MAC protocol is to regulate access to the shared wireless medium. For sensor nodes that are battery powered, it is sometimes difficult or impractical to charge or replace exhausted battery. So the medium access protocol (MAC) for wireless sensor network must be energy efficient. In this paper, we outline the sensor network properties that are crucial for the design of MAC layer protocols. Then, describe four MAC protocols proposed for sensor networks emphasizing their strengths and weaknesses and compared those protocols. Analysis of power consumption and latency performance for these four MAC protocol is also shown in this report.*

Keywords: *Latency, Medium access control, power consumption.*

I. Introduction

Wireless Sensor Networks (WSNs) are networks formed by small and low-capability devices that are able to sense environmental metrics and to communicate them wirelessly to a central unit, known as a sink. There is a wide range of potential applications in various areas, including industrial, military, environmental, health and home automation applications, which can profit from using WSNs to collect data [1]. However, the large number of constraints on the sensor nodes and the special characteristics and uses of WSNs impose several challenges on the design of a WSN. Some of these characteristics are summarized next.

To summarize, the constraints and characteristics of WSNs are the following:

- High application dependence
- Large, remote, dense and difficult-to-access deployments
- Low-cost and small devices
- Devices with limited memory, transmitting and computing resources
- Devices with limited energy resources
- High network dynamics
- Periodic, event-based and query-based communication patterns
- Data flowing from all or a group of sensors to the central unit and vice versa

Among the different constraints, limitations on energy resources are the most important because this limitation directly affects the network lifetime. This relationship is particularly important given that, in typical WSN deployments, it is too costly or even impossible to access the sensor nodes to replace the batteries. In such a situation, it is obvious that the WSN protocol stack must be designed to obtain the highest possible energy savings and thus extend the network lifetime, while maintaining the performance of the target application. Unlike communication over a guided medium in wired networks, however, communication in wireless networks is achieved in the form of electromagnetic signal transmission through the air. This common transmission medium must therefore be shared by all sensor network nodes in a fair manner. To achieve this goal, a medium access control protocol must be utilized. The choice of the medium access control protocol is the major determining factor in WSN performance. A number of access control protocols have been proposed for WSNs. The objective of this report is to discuss the fundamental design issues of medium access control for WSN methods and to provide an overview of these protocols.

This paper is organized as follows: Section II gives brief literature survey of MAC protocols. Section III represents detailed description of each protocol studied. Section IV gives analysis of each protocol with respect to power consumption and latency. Section V gives numerical analysis result. Finally, Section VI gives conclusion.

II. Trade-Off In MAC Design Of Wireless Sensor Network

Maximizing the network lifetime is a common objective of sensor network research, since sensor nodes are assumed to be disposed when they are out of battery. Under these circumstances, the proposed MAC protocol must be energy efficient by reducing the potential energy wastes presented in Section 2.1 [2] [3]. MAC Performance Matrices in 2.2 [2], Types of communication patterns that are observed in sensor network applications should be investigated since these patterns are used to extract the behaviour of the sensor network traffic that has to be handled by a given MAC protocol. Categorization of the possible communication patterns are outlined in Section 2.3.

2.1 Major Sources of Energy Wastes

Major sources of energy waste in wireless sensor network are basically of four types [2] .

- 2.1.1 Collision: The first one is the collision. When transmitted packet is corrupted due to interference, it has to be discarded and the follow on retransmissions increase energy consumption. Collision increases latency also.
- 2.1.2 Overhearing: The second is overhearing, meaning that a node picks up packets that are destined to other nodes.
- 2.1.3 Packet overhead: The third source is control packet overhead. Sending and receiving control packets consumes energy too and less useful data packets can be transmitted.
- 2.1.4 Idle listening: The last major source of inefficiency is idle listening i.e., listening to receive possible traffic that is not sent. This is especially true in many sensor network applications. If nothing is sensed, the sensor node will be in idle state for most of the time. The main goal of any MAC protocol for sensor network is to minimize the energy waste due to idle listening, overhearing and collision.

2.2 MAC Performance Matrices

In order to evaluate and compare the performance of energy conscious MAC protocols, the following matrices are being used by the research community.

- 2.2.1 Energy consumption per bit: The energy efficiency of the sensor nodes can be defined as the total energy consumed / total bits transmitted. The unit of energy efficiency is joules/bit. The lesser the number, the better is the efficiency of a protocol in transmitting the information in the network. This performance matrices gets affected by all the major sources of energy waste in wireless sensor network such as idle listening, collisions, control packet overhead and overhearing.
- 2.2.2 Average Delivery Ratio: The average packet delivery ratio is the number of packets received to the number of packets sent averaged over all the nodes.
- 2.2.3 Average Packet Latency: The average packet latency is the average time taken by the packets to reach to the sink node.
- 2.2.4 Network Throughput: The network throughput is defined as the total number of packets delivered at the sink node per time unit.

2.3 Communication Pattern

Kulkarni *et al.* defines three types of communication patterns in wireless sensor networks [9]: *broadcast*, *convergecast*, and *local gossip*. *Broadcast* type of communication pattern is generally used by a base station (sink) to transmit some information to all sensor nodes of the network. Broadcasted information may include queries of sensor query-processing architectures, program updates for sensor nodes, control packets for the whole system. The *broadcast* type communication pattern should not be confused with *broadcast* type packet. For the former, all nodes of the network are intended receivers whereas for the latter the intended receivers are the nodes within the communication range of the transmitting node. In some scenarios, the sensors that detect an intruder communicate with each other locally. This kind of communication pattern is called *local gossip*, where a sensor sends a message to its neighbouring nodes within a range. The sensors that detect the intruder, then, need to send what they perceive to the information center. That communication pattern is called *convergecast*, where a group of sensors communicate to a specific sensor. The destination node could be a cluster head, data fusion center, base station. In protocols that include clustering, cluster heads communicate with their members and thus the intended receivers may not be all neighbours of the cluster head, but just a subset of the neighbours. To serve for such scenarios, we define a fourth type of communication pattern, *multicast*, where a sensor sends a message to a specific subset of sensors.

III. Description of Mac Protocols

3.1 .T-MAC (Time-out MAC)[3]: The simplicity of S-MAC [2] using a fixed duty cycle has two drawbacks. First, an application developer is left with the burden of selecting the optimal duty cycle before deployment

commences. Second, traffic fluctuations can only be dealt with by over provisioning, that is, by setting the duty cycle to the (anticipated) maximum load at any moment, at any location in the network. In this regard convergecast and event-based reporting leave S-MAC wasting lots of energy. To address these issues, the Timeout MAC protocol by van Dam and Langendoen introduced an adaptive active period. By default nodes listen only for a short duration at the beginning of a slot (15ms for T-MAC vs. 300ms for S-MAC) and go back to sleep when no communication happens. If, on the other hand, a node engages or overhears a message transfer it will schedule another listen period after this transfer to determine if it can then go to sleep. The end result is that a node will stay active until no communication has been observed for the duration of the 15ms timeout period.



Fig. 1: Basic T-MAC Scheme

Simulations have shown that T-MAC [3] is capable of adapting to traffic fluctuations both in time (event-based reporting) and place (convergecast), and that it outperforms SMAC running at a fixed duty cycle by as much as a factor of 5 in energy consumption. In principle the timeout mechanism will automatically adapt the duty cycle to the actual traffic in a node's neighbourhood. However, T-MAC is a bit too aggressive in shutting down the radio, leaving messages queued for the next slot, which effectively increases latency and reduces throughput. In T-MAC early sleeping problem is observed where node goes to sleep when a neighbour still has messages for it. In the nodes-to-sink communication pattern, the early sleeping problem reduced the total possible throughput of T-MAC to less than half of the maximum throughput of traditional protocols or S-MAC. Early sleeping problem is simplified in Figure 2.

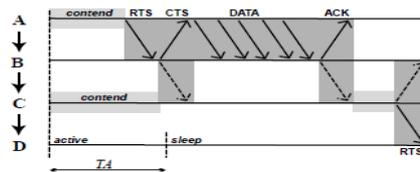


Fig. 2: Early sleeping problem

3.2 B-MAC (BERKELEY MAC)[4]:

B-MAC is a carrier sense media access (CSMA) protocol that utilizes low power listening and an extended preamble to achieve low power communication [4]. Furthermore, B-MAC is designed for duty cycled WSN, so nodes have an awake and a sleep period, and each node can have an independent schedule.

Periodic channel sampling or low-power listening (LPL) is the primary technique that B-MAC employs. LPL is carried out as follows. A node wakes up every check-interval; it turns on the radio and samples the channel. If activity (a preamble) is detected, the node remains awake for the time required to receive the incoming data packet. After reception, the node returns to sleep. However, if no packet is received, a timeout forces the node back to sleep.

If a node wishes to transmit, it precedes the data packet with a preamble that is slightly longer than the sleep period of the receiver. The preamble is predefined data automatically appended at the beginning of transmitted data. By using an extended preamble, that is at least as long as the sleep period, a sender is assured that at some point during the transmission of the preamble, the receiver will wake up and detect the preamble, and remain awake to receive the data packet.

A key challenge of B-MAC is implementing check intervals that are very short which then ensure a reasonable length for the preamble. Carrier sense duration also has to be very short so that receiver does not have to spend too much energy listening to the communication channel. A carrier sense must be accurate to reduce latency of transmission and energy consumption at sender.

B-MAC additionally utilizes software automatic gain control as a method of Clear Channel Assessment (CCA), which accurately determines if the channel is clear, thus effectively avoiding collisions. This is a necessity so that the node can determine what is a noise and what is a signal, due to the fact that ambient noise is prone to environmental changes. This is achieved by taking signal strength samples when the channel is assumed to be free, such as immediately after transmitting a packet. These samples are stored in a FIFO queue and the median of the queue is added to an exponentially weighted moving average with decay. This value gives a fairly accurate estimate of the noise floor of the channel. Effectively, a node, before transmission, takes a sample of the channel; if the noise is below the noise floor, the channel is clear and it can send immediately.

3.3 WiseMAC[7]:

Spatial TDMA and CSMA with Preamble Sampling protocol is proposed in [6] where all sensor nodes are defined to have two communication channels. Data channel is accessed with TDMA method, whereas the control channel is accessed with CSMA method. Enz et al. proposed WiseMAC [7] protocol which is similar to Hoiydi et al's work [6] but requires only a single-channel. WiseMAC protocol uses non-persistent CSMA (np-CSMA) with preamble sampling as in [7] to decrease idle listening. In the preamble sampling technique, a preamble precedes each data packet for alerting the receiving node. All nodes in a network sample the medium with a common period, but their relative schedule offsets are independent. If a node finds the medium busy after it wakes up and samples the medium, it continues to listen until it receives a data packet or the medium becomes idle again. The size of the preamble is initially set to be equal to the sampling period.

However, the receiver may not be ready at the end of the preamble, due to reasons like interference, which causes the possibility of overemitting type energy waste. Moreover, overemitting is increased with the length of the preamble and the data packet, since no handshake is done with the intended receiver. To reduce the power consumption incurred by the predetermined fixed-length preamble, WiseMAC offers a method to dynamically determine the length of the preamble. That method uses the knowledge of the sleep schedules of the transmitter node's direct neighbours. The nodes learn and refresh their neighbour's sleep schedule during every data exchange as part of the acknowledgement message. In that way, every node keeps a table of sleep schedules of its neighbours. Based on neighbour's sleep schedule table, Wise MAC schedules transmissions so that the destination node's sampling time corresponds to the middle of the sender's preamble. To decrease the possibility of collisions caused by that specific start time of wake-up preamble, a random wake-up preamble is advised. Another parameter affecting the choice of the wake-up preamble length is the potential clock drift between the source and the destination. A lower bound for the preamble length is calculated as the minimum of destination's sampling period, T_w , and the potential clock drift with the destination which is a multiple of the time since the last ACK packet arrival. Considering this lower bound, a preamble length, T_p , is chosen

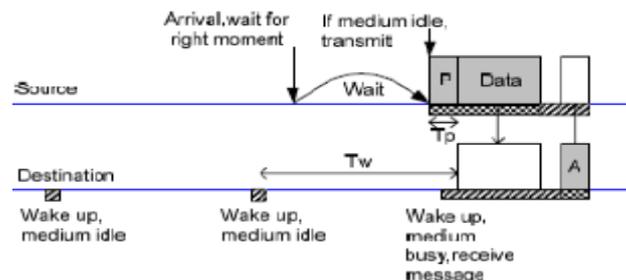


Fig.3: WiseMAC Concept

Advantages: The simulation results show that WiseMAC performs better than one of the S-MAC variants [3]. Besides, its dynamic preamble length adjustment results in better performance under variable traffic conditions. In addition, clock drifts are handled in the protocol definition which mitigates the external time synchronization requirement.

Disadvantages: Main drawback of WiseMAC is that decentralized sleep-listen scheduling results in different sleep and wake-up times for each neighbor of a node. This is especially an important problem for broadcast type of communication, since broadcasted packet will be buffered for neighbours in sleep mode and delivered many times as each neighbour wakes up. However, this redundant transmission will result in higher latency and power consumption. In addition, the hidden terminal problem comes along with WiseMAC model as in the Spatial TDMA and CSMA with Preamble Sampling algorithm. That is because WiseMAC is also based on non-persistent CSMA. This problem will result in collisions when one node starts to transmit the preamble to a node that is already receiving another node's transmission where the preamble sender is not within the range.

3.4 X-MAC [10]:

The X-MAC [10] protocol is designed for asynchronous low-power duty-cycled WSNs. It uses strobed preambles to achieve a better performance than ordinary Low-Power-Listening (LPL) based protocols. The short strobed preambles are used instead of a single large preamble. Moreover, the short preambles contain the address of the destination. Thus, a destination node may recognize its own address immediately and transmit an acknowledgment in the next gap after the preamble which reduces the medium access delay since the originator does not need to transmit all short preambles. Figure 4 shows the difference between the medium access of LPL and X-MAC.

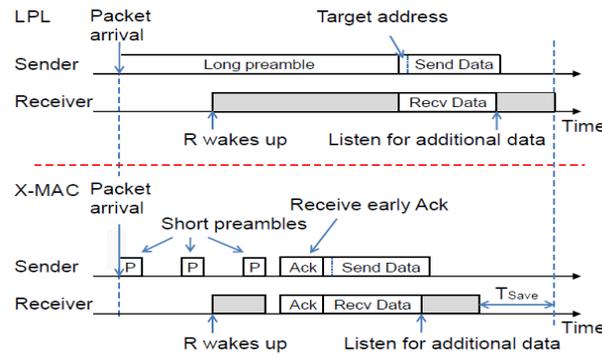


Fig. 4: LPL and X-MAC concept

The advantage of X-MAC over LPL is that the destination node can respond immediately instead of listening to the whole preamble. The originating node stops the preamble transmission and starts its data transmission after receiving the early acknowledgment from the destination node during one of the gaps.

As a result, the medium access delay is reduced by approximately 50% even in the case that there is no contention on the radio channel. The difference may become larger depending on the preamble duration, the traffic load, and the packet size. The efficiency of the protocol depends on the CCA delay and the switching time of the transceiver between rx and tx mode since these hardware limitation are responsible for the length of the short preamble and the duration of the gaps. In addition, the medium access delay is strongly affected by the hardware limitations due to the fact that they also limit the length of the duty-cycle.

The protocol takes advantage from data sniffing. A destination node stays awake a short time after it has received a data transmission. Therefore, it can respond quickly with an early acknowledgment if another node wants to send packets to it. This feature may look unimportant at first glance.

However, traffic patterns in WSNs are typically data-centric and event-driven. For this reason, data sniffing significantly affects the performance of the X-MAC [10] protocol. Moreover, the acknowledgment covers the function of a CTS message if received by a node which is not the originator of the preamble. Thus, it reduces the collision probability in multi-hop networks caused by the hidden-node problem.

The protocol is able to improve its energy efficiency depending on the traffic load since a node switches off its transceiver if it receives a preamble or an acknowledgment which is not dedicated for it. As a result, the corresponding node saves energy which prolongs its lifetime.

IV. Analysis

In this section, we present analytical expressions for our main performance metrics of interest for downlink traffic that is, power consumption and delay. We derive expression for above protocols in case of downlink communication. In the downlink direction, the challenge is to transmit data from the access point to some sensor node, without requiring that the sensor node continuously listen to the channel. A trade-off must be made between power consumption and transmission delay. Notations considered for analysis of power consumption and delay as in [11] are given in Table I.

Table I: Notations

Symbol	Description
P_{rx}	Power consumed in receive mode
P_{rxwuc}	Power consumed in receiving wakeup packet
L_w	Length of wake up packet
R_{wuc}	Data rate of the wake up radio
T_{cca}	The time for clear channel assessment or carrier sensing
P_{sl}	Power consumed in sleep state
L_{wack}	Length of immediate ACK
L_b	Length of beacon
L_{ack}	Length of ACK Packet
L_{data}	Length of data packet
T_{set}	Time required to wakeup main transceiver
P_{set}	Power consumed in waking process
T_{tt}	Switching time from Rx to Tx and vice versa
N	Total number of endpoints in the network
T_b	The time required for beacon transmission
P_{tt}	Power consumed in switching
Batcap	Battery capacity
T_{sl}	Sleep duration
R	Data rate of main radio

T_{wuc}	Duration of wake up call
T_{ack}	Time required to transmit ACK packet
T_{data}	Time required to transmit data packet
T_{hdr}	Time needed to transmit a message header
T_{cts}	Clear to send packet duration
T_{wi}	Duration of the wake-up Interval
Θ	Crystal tolerance of a wake up timer
T_p	Time required to send preamble
T_{poll}	Channel sampling duration
T_{rts}	Ready to send packet duration
T_{sync}	Sync frames interval
T	Interval between two communications
$T_{otenerg}$	Total energy consumed by the endpoint
T_{cw}	Length of contention window

4.1 Power consumption and delay of T-MAC

A beacon frame of length (Lb) is transmitted at T_{sync} intervals for the synchronization of endpoints. Each endpoint polls the channel for RTS messages at T_{wi} intervals. A data frame transmission consists of a random delay due to random back off within contention window (CW) followed by RTS, CTS, Data, and ACK frames. After receiving the RTS frame the non-targeted endpoints go to sleep mode again. Equation (1) describes the average power consumption ($PTMAC_{avg}$) and equation (4) defines the latency (DelayTMAC) involved in receiving the data packet from the coordinator. Equation (2) shows the polling interval:

$$PTMAC_{avg} = \left(Prx * \frac{L_{data}}{R} + Ptx * \frac{L_{cts} + L_{ack}}{R} + \left(\frac{T_{cw}}{2} + \frac{L_{rts}}{R} \right) * N * Prx * T + T_{poll} * Prx * T + Psl * Tsl + \left(T_{cw} - \frac{L_b}{R} \right) * \frac{1}{T_{sync}} * Prx + Pset * 4Tset + Ptt * Ttt \right) / T \quad (1)$$

Where

$$T_{poll} = \frac{2 * Tset + T_{cw} + T_{rts}}{T_{wi}} \quad (2)$$

$$\text{And } Tsl = T - (T_{rts} + T_{cts} + \frac{T_{cw} + T_b}{4} + T_{data} + T_{ack} + 4Tset + Ttt) \quad (3)$$

$$\text{DelayTMAC} = T_{data} + \frac{T_b + T_{wi}}{4} + T_{ack} + T_{rts} + T_{cts} + T_{cw} + Ttt + 3Tset \quad (4)$$

4.2 Power consumption and delay of BMAC

We model the power consumption of BMAC which is one of the most commonly used unsynchronized low duty cycle MAC protocol. Here, endpoints sleep, wake up and poll the channels periodically at T_{wi} intervals. The average power consumption ($PBMAC_{avg}$) of an endpoint is given by (5). As on average, the receiver catches the preamble half way through transmission, therefore, power consumption includes listening to the channel about half of the preamble, data reception, ACK transmission, and clear channel assessments during time (T). Other power consumption factors are overhearing of preamble, reception of packet headers by the nontargeted endpoints, sleeping, and switching from one state to another. Equation (6) shows the average delay (DelayBMAC) required for receiving the downlink packet:

$$PBMAC_{avg} = \left(Prx * T_{data} + \frac{T_p}{2} \right) + Ptx * \frac{L_{ack}}{R} + Prx * \left(\frac{T_{cca}}{T_{wi}} * T \right) + N * (T_{hdr} + \frac{T_p}{2} * Prx + (psl * Tsl + Pset * Tset + Ptt * 2Ttt)) / T \quad (5)$$

$$\text{Where } Tsl = T - (T_{cca} + T_{data} + T_{ack} + Tset + Ttt + \frac{T_p}{2}) \quad (6)$$

$$\text{DelayBMAC} = T_{data} + T_{cca} + T_{ack} + \frac{T_{avgp}}{2} + Ttt + Tset \quad (7)$$

$$\text{Where } T_{avgp} = \frac{\text{Delay}_{max} + \text{Delay}_{min}}{2},$$

$$\text{Delay}_{max} = T_{wi} + T_{data},$$

$$\text{Delay}_{min} = T_{data}$$

4.3 Power consumption and delay of WiseMAC

Equations (8) and (9) model the power consumption ($PWMAC_{avg}$) and latency (DelayWiseMAC) of WiseMAC. The average power consumption includes power consumption on receiving the data packet, transmission of ACK, and other overheads:

$$PWMAC_{avg} = \left(Prx * \left(X + \frac{L_{data}}{R} \right) + Ptx * \frac{L_{ack}}{R} + (N - 1) * Y * Prx + Psl * Tsl + Prx * \frac{T * T_p}{T_{wi}} + Pset * Tset + Ptt * Ttt \right) / T \quad (8)$$

$$\text{Where } X = 2\theta L(1 - e^{-T_{data}/4\theta L}), Y = (T_{data} + 12\theta L * T_{data}) / 2T_{wi}(1 - e^{-\frac{T_{data}}{4\theta L}}),$$

$$\text{and } Tsl = T - \left(T_{data} + T_{ack} + Tset + Ttt + \frac{T_{wi}}{2} \right);$$

$$\text{DelayWiseMAC} = T_{data} + T_p + Ttt + Tset + T_{wi}/2 \quad (9)$$

4.4 Power consumption and delay of XMAC

XMAC uses small strobe preambles. The polling time for the receiver is equal to $2TP + T_{ack}$, where TP represents preamble time and T_{ack} represents ACK timings. Equations (10) and (11) describe the average

power consumption ($PXMAC_{avg}$) and latency ($Delay_{XMAC}$) for receiving a downlink packet from the coordinator:

$$PXMAC_{avg} = (Prx * \frac{L_{data}}{R} + Ptx * \frac{L_{ack}}{R} + T_{poll} * Prx * T * N + P_{sl} * T_{sl} + P_{set} * T_{set} + P_{tt} * T_{tt})/T, \quad (10)$$

Where $T_{poll} = (2 * T_{set} + 2 * L_p + L_{ack})/T_{wi}$

And $T_{sl} = T - (T_{data} + 2T_{ack} + T_p + 3T_{set} + T_{tt} + \frac{T_{wi}}{2})$;

$$Delay_{XMAC} = T_{data} + T_b * \frac{T_{wi}}{T_p + T_{ack}} * 2 + 2T_{ack} + T_{tt} + 3T_{set} \quad (11)$$

V. Numerical Analysis Results And Discussion

The notations values used in numerical analysis are shown in Table II and Table III based on values used in [11].

Table II. Common Input Parameters

Symbol	Value
P_{tx}	26mW
P_{rx}	13.5mW
T_{data}	8ms
T_{ack}	0.6ms
P_{tt}	13.5mW
T_{tt}	0.4ms
R	40 Kbps
L_{data}	40 Bytes
P_{set}	0.0043mW
T_{set}	0.25 ms
T_{cca}	3ms
P_{sl}	0.0015mW
L_b	20 Bytes
L_{ack}	5Bytes
L_{wup}	7Bytes
N	20
Θ	30 ppm
P_{wur}	470nW

Table III. Parameters specific to each MAC

TMAC	
T_{cw}	4ms
T_{cts}	1.2ms
T_{rts}	1.2ms
T_{sync}	900 ms
T_{wi}	90 ms
T_b	4ms
BMAC	
T_{wi}	50 ms
T_p	86.7ms
Wise MAC	
T_{wi}	400 ms
T_p	20 ms
X-MAC	
T_{wi}	100 ms
T_p	2.41 ms

5.1 Power Consumption

From figure 5 we can say that X-MAC is having low energy consumption in comparison with other protocols. In TMAC, energy savings at high traffic is due to Overhearing avoidance. More neighbour nodes sleep for more time on hearing RTS and CTS when traffic is high, at low traffic they do not hear RTS/CTS thus wasting energy in idle listening during fixed duty cycle. Such observation is not seen in BMAC, XMAC, WiseMAC. In figure we can see that energy consumption in WiseMAC is less than B-MAC protocol but greater than XMAC. Reason for this is that even though XMAC does not use minimized preamble but in XMAC all neighbouring nodes turn off their radio after receiving any single preamble while in WiseMAC, all neighbouring nodes also receive full preamble with actual receiver.

5.2 Delay:

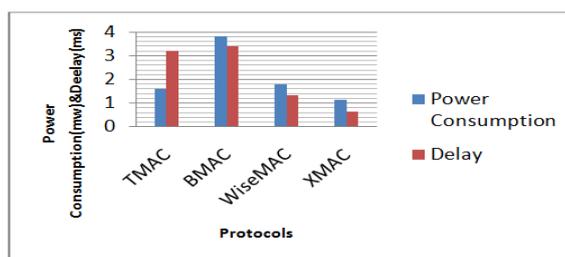


Fig. 5: Numerical result for power consumption and delay

Fig.5 shows that the latency value for X-MAC protocol is very low in comparison with other protocols. B-MAC use long preamble before data to wake up receiver. In B-MAC receiver turns its radio off after getting preamble but still sender continue to send remaining part of preamble which results in waste of energy and also introduces excess latency. While in case of WiseMAC though it manages dynamic preamble length, its latency is more than X-MAC protocol. As X-MAC uses short preamble packets and randomised back-off technique, its latency is reduced in comparison with others.

VI. Conclusion

In this paper, different Wireless Sensor Network MAC protocols such as Timeout T-MAC protocol, Wise-MAC protocol, Berkeley Media Access Control (B-MAC) protocol, X-MAC have been discussed. We have drawn the conclusion that the MAC protocol influences network lifetime. However, different MAC protocols can be efficient depending on the given environment and applications. During this work, we realized that the MAC protocols for the wireless sensor networks are a hard and extensive area. As protocols are compared with its numerical result, we found that X-MAC performs well in terms of both power consumption and delay. As X-MAC uses short preamble packets and randomised back-off technique. Some of the planned work has to be streamlined away for future work. We see clear paths for future work:

- Verification through employment and wide ranging simulations.
- Formal descriptions to address other type of MAC protocols and addition of components.
- Cross layer optimization is an area that needs to be explored more extensively.
- Extension of components and formal descriptions to address the other type of WSN MAC Protocols.

Acknowledgement

The authors would like to thank the anonymous reviewers for their helpful feedback, suggestions and comments, if any, to improve the presentation of this paper.

References

- [1] I. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci: A Survey on Sensor Networks, IEEE Communication Magazine, pp. 102-114, August 2002.
- [2] Wei Ye, J.Heidemann and D. Estrin: An Energy-Efficient MAC Protocol for Wireless Sensor Networks, IEEE INFOCOM, New York, Vol. 2, pp. 1567-1576, June 2002
- [3] Tijs van Dam, Koen Langendoen: An Adaptive Energy Efficient MAC Protocol for Wireless Networks, in Proceedings of the First ACM Conference on Embedded Networked Sensor Systems November 2003.
- [4] J. Polastre, J. Hill, D. Culler: Versatile low Power Media Access for Wireless Sensor Networks, Proceedings of the 2nd ACM Conference on Embedded Networked Sensor Systems (SenSys'04), Baltimore, MD, November 2004.
- [5] C.C. Enz, A. El-Hoiydi, J.-D. Decotignie, V. Peiris: WiseNET: An Ultralow-Power Wireless Sensor Network Solution, IEEE Computer, Vol. 37, Issue 8, August 2004.
- [6] A. El-Hoiydi: Spatial TDMA and CSMA with Preamble Sampling for Low Power Ad-hoc Wireless Sensor Network, Proceedings of ISCC'02, Seventh International Symposium on Computers and Communications, pp. 685-692, July 2002.
- [7] A. El-Hoiydi and J. D. Decotignie, WiseMAC: An ultra low power MAC protocol for multihop wireless sensor networks, in First Int. Workshop on Algorithmic Aspects of Wireless Sensor Networks (ALGOSENSORS 2004), Lecture Notes in Computer Science, LNCS 3121, Springer-Verlag, pp. 18-31, July 2004.
- [8] S. Coleri-Ergen and P. Varaiya, Pedamacs: Power efficient and delay aware medium access protocol for sensor networks, IEEE Trans. on Mobile Computing, pp. 920-930, May 2006.
- [9] S.S. Kulkarni, "TDMA Services for Sensor Networks", proceedings of 24th International Conference on Distributed Computing Systems Workshops, Pages: 604-609, 23-24 March 2004
- [10] M. Buettner, G. Yee, E. Anderson, and R. Han, X-MAC: A short preamble MAC protocol for duty-cycled wireless networks, in 4th ACM Conf. on Embedded Networked Sensor Systems (SenSys 2006), Boulder, CO, pp. 307-320, Nov 2006.
- [11] Niamat Ullah, M. Sanaullah Chowdhury, Mosaddique Al Ameen, and Kyung Sup Kwak, "Energy Efficient MAC Protocol for Low-Energy Critical Infrastructure Monitoring Networks Using Wakeup Radio" International Journal of Distributed Sensor Networks Volume 2012, Article ID 504946, 15 pages, doi:10.1155/2012/50494