Scientific Research

A Comparative Study of Medium Access Control Protocols for Wireless Sensor Networks

Meghan GUNN, Simon G. M. KOO
Department of Mathematics and Computer Science, University of San Diego, San Diego, USA
Email: {*meghangunn-09, koo*}*@sandiego.edu*

Abstract

One of the major constraints in wireless sensor networks (WSNs) is power consumption. In recent years, a lot of efforts have been put into the design of medium access control (MAC) protocols for WSN, in order to reduce energy consumption and enhance the network's lifetime. In this paper, we surveyed some MAC protocols for WSN and compared their design tradeoffs. The goal is to provide a foundation for future MAC design, and to identify important design issues that allow us to improve the overall performances.

1. Introduction: Wireless Sensor Networks

A wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices that use sensors to cooperatively monitor physical or environmental conditions such as temperature, sound, vibration, pressure, motion, or pollutants at different locations. The purpose of a WSN is to collect and process data from a target domain and transmit information back to specific sites. WSN technology is an emerging technology that can be utilized in a wide range of potential applications including but not limited to, biomedical treatment, military applications, traffic surveillance, fire detection, structural and earthquake monitoring, industrial control, and rescue operations.

Such a network usually consists of a number of wireless sensor nodes that arrange themselves into a multi-hop network. Each node consists of one or more sensors, a low power radio transceiver or other wireless communication device, an embedded processor, and an energy source, usually a battery. The size of a wireless node can vary from the size of a shoebox, down to size of a grain of dust, and cost varies depending on size. These size and cost constraints result in corresponding constraints on nodes resources, including energy, memory, computational speed and bandwidth.

2. Factors for Design of a MAC Layer Protocol

Considering that sensor nodes are likely to be battery powered, and because they are often implemented in environments where it proves to be difficult to change or recharge batteries, prolonging the lifetime of nodes is a critical issue for a successful wireless sensor network. Not only does the transmission of data cost energy, but receiving, and scanning for data also use a significant amount of energy. In addition to being energy efficient, WSN should be scalable and adaptable to change. Change can come in the form of network size, node density, or topology. Additionally, nodes may die over time, new nodes may join, or nodes may move to a different location. A good MAC protocol should gracefully accommodate such network changes. Lastly fairness, latency, throughput, bandwidth utilization are also concerns for WSN. However, these goals may be primary concerns in traditional wireless networks, but they prove to be secondary for WSN. This is due to the fact that in a traditional wireless network, usually a number of different applications may be competing for use of the communication channel; however in a WSN, the nodes are typically working for the same application.

3. Sources of Energy Waste

In a sensor, the Radio Frequency (RF) module, which consumes most of the energy, becomes the crucial entity to be optimized. Therefore, designing an energy-efficient Medium Access Control (MAC) protocol is significant factor in reducing energy consumption based on its direct control over RF module [2]. There are four distinctive sources of energy waste for wireless sensor nodes, colli-

sions, overhearing, control packet overhead, and idle listening. Collisions are caused by contention, when two nearby sensor nodes both attempt to access the communication channel at the same time. Overhearing is a result of a node picking up packets that are destined to other nodes. Common control packets used in WSN include Ready-to-Send (RTS), Clear-to-Send (CTS), and Acknowledge (ACK). The transmission of these packets contributes to energy consumption, therefore a minimal number of control packets should be used to make a data transmission. Idle listening has proved to be one of the major sources of energy waste. Given that a node does not know when it will be the receiver of a message from one of its neighbors, it must keep its radio in receive mode at all times, resulting in idle listening. Studies have shown that idle listening can consume 50-100% of the power required for receiving [1].

4. Proposed MAC Protocols

There are a considerable number of MAC protocols that have been designed and implemented for WSN. This section will discuss a few of these protocols and their essential behaviors.

4.1. Sensor MAC (S-MAC)

The key idea behind S-MAC [1] is the utilization of managed synchronized duty cycles. A duty cycle utilizes a periodic awake and sleep schedule, allowing nodes in sleep mode turn off their radio [1]. A duty cycle is represented as a ratio of wake time to total cycle time, S-MAC limits it duty cycles to about 10%, reducing energy waste by attempting to minimize idle listening. Sleep and listen periods are predefined and constant in S-MAC.

Additionally, nodes in S-MAC create virtual clusters by periodically exchanging sleep schedules with their neighboring nodes [1]. This exchange is implemented by sending a SYNC packet, which is very short, and includes the address of the sender and the time of its next sleep. Nodes that receive the SYNC packet will adjust their timers immediately after they receive the SYNC packet and will go to sleep when the timer fires. Thus the schedules are updates and the nodes are synchronized.

Nodes that reside in two virtual clusters wake up for the listen phases for both clusters. This however is one of the drawbacks of the S-MAC algorithm, the possibility of a node following two different schedules resulting in more energy consumption via idle listening and overhearing.

Lastly, S-MACs design includes the utilization of adaptive listen, overhearing avoidance techniques, and message passing. With adaptive listen, neighboring nodes wake up for a short period of time at the end of each transmission to listen for possible data transmissions. To avoid overhearing, all immediate neighbors of sender and receiver are put to sleep upon receiving RTS/CTS. Resultantly neighbors do not overhear data packets and following ACKS. The nodes use the duration field in the packet, which indicates how long to sleep. Message passing is a technique in which long messages are divided into frames and sent in a burst. With this method, nodes may achieve energy savings by minimizing communication overhead and latency at the expense of unfairness in medium access.

4.2. Timeout MAC (T-MAC)

T-MAC is similar to S-MAC in that it utilizes an active/sleep duty cycle. However, TMAC improves upon the design of S-MAC by introducing an adaptive duty cycle in which the active part is dynamically ended, increasing the efficiency of the algorithm for variable traffic loads. The idea behind the design of T-MAC is as follows. While latency requirements and buffers space are generally fixed, the message rate will usually vary. Therefore, the nodes must be implemented with an active time that can handle the highest expected load. Whenever the load is lower than that which is expected, the active time is not optimally used and energy will be wasted on idle listening. To solve this inefficiency, the T-MAC protocol implementation reduces idle listening by transmitting all messages in bursts of variable length, and sleeping between bursts. To maintain an optimal active time under variable load, the length of the active time is dynamically determined, ending in an intuitive way by timing out when the node hears nothing [4].

Every node periodically wakes up to communicate with its neighbors during active time periods. The nodes communicate using a modified RTS-CTS-DATA-ACK four-step exchange to deliver messages, which provides both collision avoidance and reliable transmission [4]. A node will keep listening and potentially transmitting, as long as it is in an active period. An active period ends when no activation event has occurred for a time TA. An activation event includes but is not limited to, the reception of any data on the radio, the sensing of communication of the radio, the end-of-transmission of a node's own data packet or an ACK data packet. If no activation event is sensed, the node then goes to sleep again until the next frame. During the sleep mode, new messages are queued.

An important aspect of T-MAC is determining TA, the time that a node must wait before it times out, and goes to sleep. The idea is that a node should not go to sleep while its neighbors are still communicating, since it may be the receiver of a subsequent message [4]. Receiving the start of the RTS or CTS packet from a neighbor is

enough to trigger a renewed interval TA. Additionally, a node may be out of range, and therefore may not hear the RTS that starts a communication with its neighbor, so the interval TA must be long enough to receive at least the start of the CTS packet.

4.3. S-MACL, a Global Sleeping Schedule

As mentioned the S-MAC protocol creates virtual clusters in which the clustered nodes follow a common sleeping schedule. In order to connect these virtual clusters, nodes residing between clusters have to adopt multiple schedules. These nodes, known as border nodes, constitute nearly 50 percent of the nodes in some networks and may have to adopt up to 4 different schedules. These border nodes have to stay in active mode longer than other node, which means that they waste more energy than non-border nodes. Resultantly, these nodes will die sooner, and the network coverage rate is reduced. A more serious problem happens in multi-hop sensor networks, in which border nodes have to act as intermediate outers to relay packets. The death of these border nodes may increase the routing difficulty, even segment a network. Some nodes will not be able to communicate to the rest of the network [6].

To resolve the problem of multiple sleeping schedules, S-MACL attempts to merge all the virtual synchronization clusters into one cluster to ensure that only one sleeping schedule will be used in a fully connected network. To do this, S-MACL utilizes the node id, a unique identifier that is mounted on each sensor node. More specifically it uses the id of the synchronizing sender node and applies it as a schedule id. The scheduling process in S-MACL is presented as follows. When a node does not receive any SYNC frame after its first listening period, it will arbitrarily choose one schedule and announce this schedule and assign its own id as the schedule id. We call such a node a synchronizer, since it chooses its schedule independently and other nodes will synchronize with it. Otherwise, the node will receives a schedule from a neighbor SYNC frame before having a chance to choose its own schedule, and will follow that schedule by setting its schedule as the same, and announcing this schedule to its neighbors. We call such a node a follower. When a node receives a different schedule from its neighbors' SYNC frame, it will compare the current schedule id and the new schedule id. Then it will start following the schedule with the higher id. If the new schedule in the incoming SYNC frame has a lower id, this node will announce its own schedule during the listening time of the new schedule. This operation ensures that nodes will always use the schedule with highest id. The authors show through various scenarios, with different numbers of nodes and different topologies that S-MACL performs better than S-MAC in most cases [6].

4.4. Patten MAC (P-MAC)

P-MAC [14] is unique in that instead of having fixed sleep and awake schedules as with S-MAC, the sleep-wakeup schedules of the sensor nodes are adaptively determined, based on a node's own traffic and that of its neighbors. This improves throughput under heavy traffic and reduces unwanted energy consumption while the networks is performing under light loads when compared to the performance S-MAC.

Similar to S-MAC, P-MAC is a time-slotted protocol, however unlike S-MAC in which a node sleeps for a duration of the time slot and is awake for the remainder, in P-MAC, the node must either be awake or asleep for the entire duration of the time slot. With P-MAC, a sensor node gets information about the activity in its neighborhood before sending communication packets through patterns. Based on these patterns, a sensor node can put itself into a long sleep for several time frames when there is no traffic in the network. If there is any activity in the neighborhood, a node will know this through the patterns and will wake up when required. Thus P-MAC saves more power than S-MAC as well as T-MAC, without compromising on the throughput.

A sleep-wakeup pattern is a stream of bits indicating the tentative sleep-wakeup plan for a sensor node over several slot times [14]. A 1 in the stream indicates that the node intends to stay awake during a slot time, while a 0 indicates that the node intends to sleep. Since the pattern is only a tentative plan, it is subject to change. This pattern stream of 1s and 0s is generated for each individual node. These patterns are used to convey activity from one node to its neighbors. Thus, the schedule for a node is derived from its own pattern and, as well as the patterns of its neighboring nodes, resulting in a schedule for the network.

Pattern generation based on the binary strings that are associated with a node over some number of time slots, this is referred to as a period [14]. The nodes' pattern is updated during each period using local traffic information available at the node and exchanged between the neighboring nodes at the end of each period. When the network is activated, the pattern at every node has just one bit for the first period, which is 1. If there is no data for a node to send at the first time slot of bit 1, then it indicates that the traffic load is light, and the node can afford to go to sleep. Consequently, the node updates its pattern to 01, and so on. If during the next time slot, the node still has no data to send, the node is encouraged to sleep longer by doubling the number of 0 bits, ie. 001. By exponentially increasing the sleep time during light traffic the node is able to save a considerable amount of energy. On the other hand, if a node has any data to transmit at any time slot, regardless of the pattern bit at that time slot, the next bit in the pattern becomes a 1.

These patterns are not the decisive sleep schedule for the nodes; they are only a tentative sleep-wakeup plan [14]. As mentioned P-MAC obtains it schedule based on the node's pattern, and the pattern of its neighbors. The nodes broadcast newly generated patterns at the end of the current period. As a result, the time is divided into time frames, referred to as super time frames (STF). Each STF has two sub-frames. In the first, the Pattern Repeat Time Frame (PRTF), each node repeats its current pattern. The second time frame, the Pattern Exchange Time Frame (PETF), is used for the exchange of new patterns between neighbors. To obtain the actual sleep-wakeup itinerary, the strings of bits are compared between neighbors at each time slot as well as looking for data packets in the buffer of the neighboring nodes at each time slot. Based on a series of rules, the bits are compared and a new pattern is created and followed. In addition to the use of 1s and 0s, 1- bit is introduced. A 1-bit will be used when the nodes pattern bit is 1 and either the receivers bit is 0, or the node has no packets to be sent. Therefore, 1- implies that the node should wakeup at the beginning of the time slot and listen for a short amount of time. If it hears no communication from its neighbors, then it goes back to sleep. The reason for this is that since the pattern bit for the node is 1, the node is a candidate to be a receiver of communication and its neighbors may try to send data to it. Thus, if the node goes to sleep, the packet destined to it will be lost, and energy is wasted.

4.5. Traffic-Adaptive MAC (TRAMA)

As a traffic load increase, the probability of collisions of control or data packets occurring in any contention-based scheme increases. This degrades channel utilization and further reduces battery life [7]. TRAMA implementation attempts to provide energy-efficient conflict free channel access in wireless sensor networks by creating transmission schedules that are adaptive to changes, prolongs the battery life of each node, and is robust to wireless loses [7]. The protocol consists of three components: the Neighbor Protocol (NP), the Schedule Exchange Protocol (SEP) and the Adaptive Election Algorithm (AEA). Additionally, TRAMA uses single, time-slotted channel access that is divided up into random and scheduled access periods.

The main function of the Neighbor Protocol is to gather two-hop neighborhood information by using signaling packets. This protocol operates periodically during random access periods. Schedule Exchange Protocol utilizes a schedule consisting of intended receivers for future transmission slots. Schedules are established based on the current traffic information at the node, and are periodically propagated to the neighboring node. SEP maintains consistent schedules for the one-hop neighbors

of each node. The Adaptive Election Algorithm uses the schedule information from SEP and the neighborhood information to elect a transmitter, receiver and stand-by nodes for the current time slot. Nodes that are not selected to transmit or receive data for a particular time slot are removed from the election process, allowing them to switch to sleep mode and improving the channel utilization. As a result, the sleep schedule of a node is a direct function of the traffic going through the node and its neighbors, and is synchronized automatically when nodes exchange information about their identifiers and their traffic [7].

TRAMA organizes access to the communication channel into time slots allowing random and scheduled access. Random Access periods are used for signaling, synchronization, and updating two-hop neighbor information. The scheduled access periods are used for contention free data exchange between nodes.

4.6. B-MAC, a Versatile Low Power MAC

B-MAC is a carrier sense media access (CSMA) protocol that utilizes low power listening and an extended preamble to achieve low power communication [10]. 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 [10].

4.7. X-MAC, a Short Preamble MAC

While being simple and improving energy efficiency, the low power listening approach used by B-MAC which employs a long preamble is suboptimal in terms of energy consumption, is subject to overhearing, as well as introducing excess latency at each hop [11]. This issue is threefold. First, the receiver typically has to wait the full period until the preamble is finished before the DATA/ACK exchange can begin, even if the receiver has woken up at the start of the preamble. Second, LPL suffers from the overhearing problem, where receivers who are not the target of the sender also wake up during the long preamble and have to stay awake until the end of the preamble to find out if the packet is destined for them. This wastes energy at all non-target receivers within

transmission range of the sender. Third, because the target receiver has to wait for the full preamble before receiving the data packet, the per-hop latency is lower bounded by the preamble length. Over a multi-hop path, latency can accumulate to become substantial [11].

X-MAC is a low power MAC protocol that strives to overcome these shortcomings by employing a shortened preamble approach. The ideas behind this approach is to embed address information of the target node in the preamble so that non-target receivers can realize that they are not the receiver and quickly go back to sleep. This solution addresses the overhearing problem. Furthermore, X-MAC introduces the strobed preamble. This approach allows the target receiver to interrupt the long preamble as soon as it wakes up and determines that it is the target receiver. This is accomplished by dividing the one long preamble into a series of short preamble packets, each containing the id of the target node. Accordingly, instead of sending a constant stream of preamble packets, the protocol inserts small pauses between the series of short preamble packets, during which time the transmitting node pauses to listen to the medium. These gaps enable the receiver to send an early ACK packet back to the sender by transmitting the ACK during the short pause between preamble packets. When a sender receives an ACK from the intended receiver, it stops sending preambles and sends the data packet. This allows the receiver to cut short the excessive preamble, which reduces per-hop latency and energy spent unnecessarily waiting and transmitting [11].
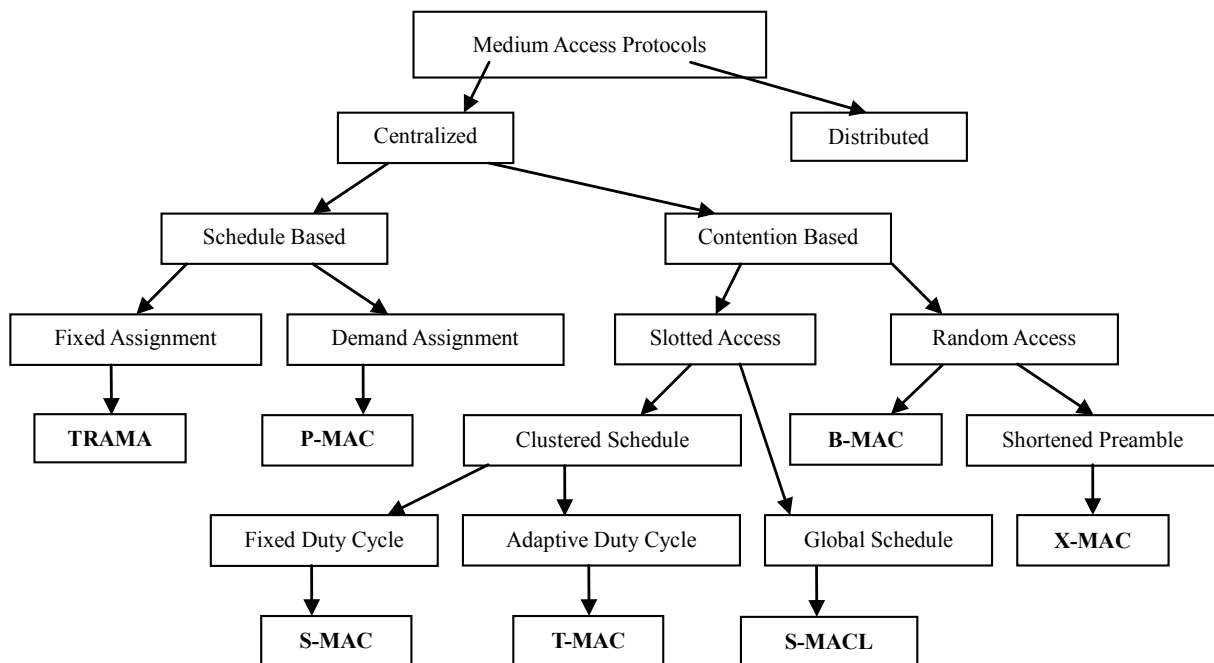

Figure 1. MAC design options.

Table 1. Tradeoff analysis.

	Energy	Fairness	Latency	Throughput
S-MAC	(+) Periodic Sleep (+) Message Passing (-) Idle Listening (-) Overhearing	(-) Message Passing	(-) Periodic Sleep (+) Adaptive listen (+) Message Passing	(-) Periodic Sleep
T-MAC	(+) Adaptive Duty Cycle		(-) Adaptive Duty Cycle	(-) Adaptive Duty Cycle
S-MACL	(+) Global Sleep Schedule			
P-MAC	(+) Adaptive Sleep Schedules			(+) Adaptive Sleep Schedules
TRAMA	(+) Transmission Schedules (-) Overhearing	(+) Transmitter Electron Algorithm		(+) Transmitter Electron Algorithm
B-MAC	(+) LPL (-) Long Preamble (-) Overhearing			
X-MAC	(+) Shortened Preamble		(+) Strobed Preamble	

5. A Comparison

5.1. Comparison of Design

Centralized MAC protocol design can be divided into two sub sections, schedule- based, and contention-based. A schedule based design schedules nodes into different sub-channels. Schedules protocols are successful in that they avoid collisions thus promoting energy efficiency. However, they tend to have poor scalability and adaptability. On the other hand in a contention-based schedule, nodes compete in a probabilistic coordination for access to the communication channel. Contention-based protocols have proved to be more scalable and flexible to topology change. However, when compared with schedule-based designs, they are not as energy efficient. Figure 1 illustrates which of the previous protocols discussed use each of the schemes and partitions the design of the protocols into more detailed subsets.

5.2. Analysis of Tradeoffs

Table 1 represents comparison of the tradeoffs in protocol design based on the statistics available. Due to resource constraints, the table is not complete. A (+) indicates a positive outcome of the subsequent design method. A (-) indicates a tradeoff of a network performance metric as a result of the design technique.

5.3. Protocol Comparison

This section offers a more detailed discussion of the advantages and disadvantages of the protocol design. It also offers a direct comparison between some of the protocols. Again, for some of the protocols, the information is minimal due to a lack of available information.

5.3.1. S-MAC
S-MAC reduces the amount of energy wasted by idle listening, which is accomplished by introducing sleep schedules. Its implementation is simple, and time synchronization overhead is prevented with sleep schedule announcements. Lastly, adaptive listening is used to reduce multi-hop latency due to periodic sleep modes and nodes waiting until the subsequent listen period of the intended receiver. Adaptive listen saves more energy for heavy loads by reducing latency by at least half.

On the other hand, the S-MAC protocol essentially trades energy efficiency for reduced throughput and increased latency. Throughput is reduced because only the active part of the frame is used for communication. Latency increases because a message-generating event may occur during sleep time. Additionally, adaptive listening incurs overhearing or idle listening if the packet is not destined to the listening node. Lastly, sleep and listen periods are predefined and constant, which decreases the efficiency of the algorithm under variable traffic load.

For light traffics loads S-MAC offers significant energy efficiency over always listening MAC protocols. Simulation experiments have shown that the S-MAC protocol reduces the energy used by the radio with up to 30%, after optimal tuning. The energy savings and increased throughput of S-MAC as compared with traditional protocols without sleep cycles such as CSMA and IEEE 802.11 without duty cycle control is shown in Figures 2 and 3.
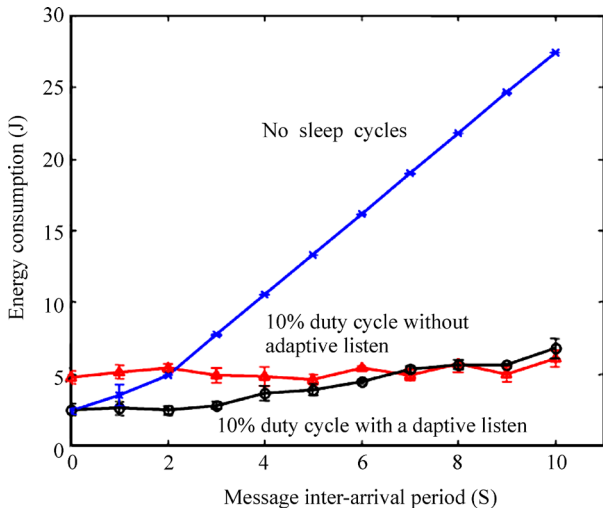
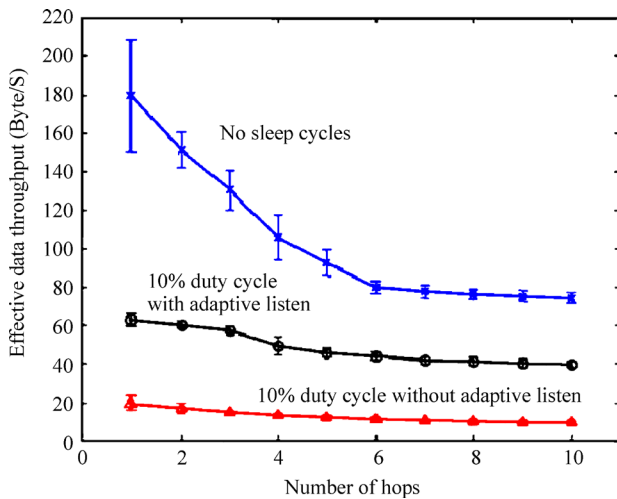**Figure 2. Energy consumption at different traffic loads.**

Figure 4. Energy consumption based on event triggered reporting.

Figure 3. Effective through put under highest traffic load.

Figure 2 shows that at light traffic load, periodic sleeping has significant energy savings over fully active mode and adaptive listen saves more at heavy load by reducing latency. In Figure 3 one can see that adaptive listen significantly increases throughput.

5.3.2. T-MAC
Simulation experiments have shown that the T-MAC protocol reduces the energy used by the radio with as much as 80% in a typical scenario when compared to classical protocols like CSMA. The S-MAC protocol saves only 30% in this scenario, after optimal tuning. Implementation of the T-MAC protocol on real wireless sensor hardware has shown that, in an idle situation, the radio can be turned off for as much as 97.5% of the time, reducing the total energy used more than 96%. In a situation with high message rates, the T-MAC protocol does
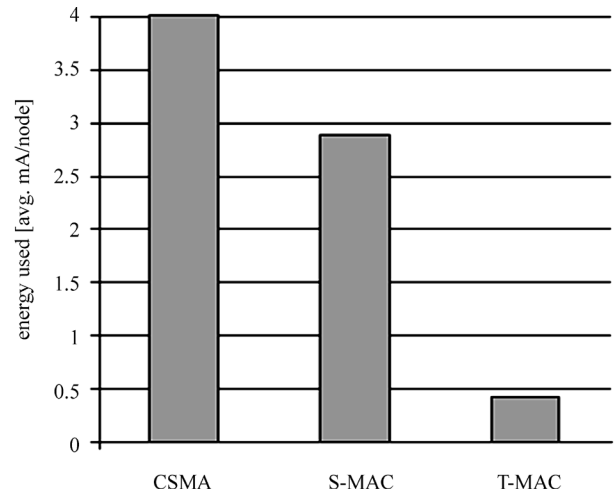
not increase the latency, since nodes do not sleep in that case. Furthermore, the authors show that, for variable workloads, T-MAC uses one fifth of the energy used by S-MAC. While this adaptive duty cycling reduces energy usage for variable workloads, these gains come at the cost of reduced throughput and increased latency. Results of simulations are illustrated in Figure 4, which compares the amount of energy used for CSMA, S-MAC, and T-MAC in a typical scenario.

5.3.3. S-MACL
With S-MACL, all nodes consume less energy, especially the border nodes that act as intermediate routers, greatly increasing the lifetime of these nodes. Additionally, as a result of the global synchronization schedule, the number of collisions is reduced, which also reduces the amount of energy wasted. The contrastive simulation of S-MAC with S-MACL results showed that S-MACL achieves a great level of energy efficiency compared with S-MAC.

5.3.4. P-MAC
Based on simulations done by the authors, in comparison to S-MAC under light traffic loads, P-MAC consumes less energy, though throughput remains the same. However, under heavy traffic loads, P-MAC consumes less energy and achieves a higher throughput. This is due to the fact that with S-MAC, sensor nodes must periodically go to sleep, even if the traffic load is high. On the other hand, the implementation of P-MAC allows the nodes to stay awake due to the varying schedule patterns. Because PMAC is able to adaptively schedule sleep and awake periods, it offers more energy savings under light loads, and higher throughput under heavy loads as compared to S-MAC.

5.3.5. TRAMA

TRAMA is able to improve energy efficiency by utilizing transmission schedules that avoid collisions of data packets at the receiving nodes. Additionally nodes switch to low power radio mode when there are no data packets intended for those nodes. Furthermore, TRAMA achieves conflict-free transmission by scheduling access among two-hop neighboring nodes during a particular time slot and by allowing nodes to switch to sleep mode if they are not selected to transmit or are not the intended receivers of traffic for a particular time slot. Lastly, adequate throughput and fairness is achieved based on the transmitter-election algorithm that is inherently fair and promotes channel reuse as a function of the competing traffic around any given source or receiver. On the other hand, TRAMAs efficiency is limited by its complex election algorithm and data structure. Moreover, it incurs overhead due to explicit schedule propagation as well as higher queuing delays [7].

TRAMA implementation results in a higher percentage of sleep time and less collision probability when compared to CSMA based protocols, which greatly improves energy savings. TRAMA has a higher delay but higher maximum throughput than contention-based S-MAC. Through extensive simulations, TRAMAs performance is compared against a number of contention and a scheduled based MACs. It is evident from the simulation results that significant energy savings can be achieved by TRAMA depending on the offered load. TRAMA also achieves higher throughput (around 40% over S-MAC and CSMA and around 20% over 802.11) when compared to contention-based protocols because it avoids collisions due to hidden terminals [7].

5.3.6. B-MAC

The authors have show that testing the communication channel for activity is about 10x less expensive than listening for a full contention period. Idle listening is reduced in the B-MAC protocols by shifting the burden of synchronization to the sender: when a sender has data, the sender transmits a preamble that is at least as long as the sleep period of the receiver; thus, the sender and receiver can be completely decoupled in their duty cycles [10]. This removes the need for, and the overhead introduced by, synchronized wake/sleep schedules.

The authors show that B-MAC surpasses existing protocols in terms of throughput, latency, and for most cases energy consumption. It is simple in both design and implementation. While B-MAC performs quite well, it suffers from the overhearing problem, and the long preamble dominates the energy usage. Additionally, while, unscheduled sleep reduces control overhead, consequentially, the sender incurs greater overhead to wake up the unsynchronized receiver from sleep.

The performance benchmark has shown that B-MAC outperforms S-MAC with greater energy savings and network performance [7].

5.3.7. X-MAC

B-MAC requires more time to transfer packets from the source to the destination. This is because the entire preamble has to be always sent, even though the receiver was already awake. X-MAC saves this time, thus conserving energy.

6. Conclusions

When developing a MAC protocol, prolonging lifetime for nodes is a critical issue to consider in order to promote for a successful wireless sensor network. Many of the developed protocols are developed with specific assumptions in mind and for specific applications. In this article, we surveyed wireless MAC protocols for wireless sensor networks, and we can conclude that no protocol is the "best" implementation. However, each of these protocols addresses different issues that arise from energy waste in sensor nodes.

7. References

[1] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor network," In Proceedings of the INFOCOM'02. IEEE Computer Society, San Francisco, 2002.

[2] J. Ai, J. F. Kong, and D. Turgut, "An adaptive coordinated medium access control for wireless sensor networks," In Computers and Communications Proceedings, ISCC'04, 2004.

[3] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," In Computer Networks, Vol. 38, No. 4, pp. 393–422, 2002.

[4] T. Zheng, S. Radhakrishnan, and V. Sarangan, "PMAC: An adaptive energy-efficient MAC protocol for Wireless Sensor Networks," In Parallel and Distributed Processing Symposium Proceedings, 19th IEEE International, 2005.

[5] LAN MAN Standards Committee of the IEEE Computer Society, IEEE Std 802.11-1999, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE, 1999.

[6] L. Zhang, G. Somnath, V. Prakash, and S. Samar, "An energy efficient wireless sensor MAC protocol with global sleeping schedule," In Computer Science and its Applications, CSA'08, International Symposium, pp. 303– 308, October 2008.

[7] V. Rajendran, K. Obraczka, and J. J. Garcia-Luna-Aceves "Energy-efficient, collision-free medium access control for wireless sensor networks," In Wireless Networks, Vol. 12, pp. 63, 2006.

[8] J. M. So and N. Vaidya, "Multi-channel MAC for ad hoc networks: Handling multi-channel hidden terminals using

a single transceiver," Talk at Workshop with Intl. School on WSN, Dagstuhl, Germany, August 30, 2005.

[9] C. L. Fullmer and J. J. Garcia-Luna-Aceves, "Solutions to hidden terminal problems in wireless networks," In Proceedings ACM SIGCOMM'97, Cannes, France, September 14-18, 1997.

[10] J. Polastre, J. Hill, and D. Culler, "Versatile low power media access for wireless sensor networks," In Second ACM Conference on Embedded Networked Sensor Systems, 2004.

[11] M. Buettner, G. V. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks," In Proceedings of 2nd ACM Conference on Embedded Networked Sensor Systems (SenSys'06), pp. 307–320, 2006.

[12] S. Singh and C. Raghavendra, "PAMAS: Power aware multi-access protocol with signalling for ad hoc networks," ACM SIGCOMM Computer Communication Review, Vol. 28, No. 3, pp. 5–26, July 1998.

[13] P. Karn, "MACA - a new channel access method for packet radio," In ARRL/CRRL Amateur Radio 9th Computer Networking Conference, pp. 134–140, 1990.

[14] LAN MAN Standards Committee of the IEEE Computer Society, IEEE Std 802.11-1999, Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, IEEE, 2000.