On the Medium Access Control Protocols Suitable for Wireless Sensor Networks – A Survey

Mohamed Guerroumi[1], Al-Sakib Khan Pathan[2], Nadjib Badache[3], and Samira Moussaoui[1]

[1]Electronic and Computing Department, USTHB University, Algiers, Algeria
[2]Computer Science Department, International Islamic University Malaysia, Kuala Lumpur, Malaysia
[3]DTISI, CERIST Research Center, Algiers, Algeria
mguerroumi@usthb.dz, sakib@iium.edu.my, badache@mail.cerist.dz, and smoussaoui@usthb.dz

Abstract**:** A MAC (Medium Access Control) protocol has direct impact on the energy efficiency and traffic characteristics of any Wireless Sensor Network (WSN). Due to the inherent differences in WSN's requirements and application scenarios, different kinds of MAC protocols have so far been designed especially targeted to WSNs, though the primary mode of communications is wireless like any other wireless network. This is the subject topic of this survey work to analyze various aspects of the MAC protocols proposed for WSNs. To avoid collision and ensure reliability, before any data transmission between neighboring nodes in MAC layer, sensor nodes may need sampling channel and synchronizing. Based on these needs, we categorize the major MAC protocols into three classes, analyze each protocol's relative advantages and disadvantages, and finally present a comparative summary which could give a snapshot of the state-of-the-art to guide other researchers find appropriate areas to work on. In spite of various existing survey works, we have tried to cover all necessary aspects with the latest advancements considering the major works in this area.

Keywords**:** Medium, Access, Energy, Latency, MAC protocol, Collision, Wireless, Sensor, Networks**.**

1. Introduction

Energy efficiency is one of the most critical issues for any Wireless Sensor Network (WSN). This type of network is often envisaged to be deployed in places where human beings may not get easy access (or, no physical access) and hence, the power sources of the sensor nodes could not be recharged or replaced. In this scenario, the gradual degradation of the batteries decreases the useable lifetime of WSNs. There are various ongoing attempts to find out new mechanisms from hardware perspectives, for energy harvesting or for prolonging battery lifetime of the sensors to allow the network the maximum possible longevity. Energy, being a very important resource for the network's operation for the intended period of time, there are in fact, several proposed solutions of energy conservation from the angle of communication protocols as well. In spite of the availability of such hardware and software or technical solutions, the choice of energy conservation protocol still remains difficult. Often, depending on the network characteristics or setting of the application environment, choices are to be made. As hardware solutions for battery technologies are still evolving to be more efficient, the researchers often focus on the operational or software techniques to address energy efficiency issue in WSN. These are separate areas of investigation and for our work; we would like to analyze this energy-efficiency issue from the perspective of software or communication protocols or more specifically, Medium Access Control (MAC) protocols.

The principal sources of energy consumption in WSN are collision, control packet overhead, idle listening, and the overhearing [1-2]. All these dominant parameters are related directly to the operating mode of MAC (Medium Access Control) protocols, which motivated us to study the various protocols proposed for this layer.

Designing power efficient MAC protocol is one of the ways to prolong the lifetime of the network. To find out the advancements, achievements, challenges, and issues in this topic, here, we present a study of the energy efficient MAC protocols for wireless sensor network. We present the basic concepts, the operating modes, and the characteristics of each protocol by scrutinizing the strong and the weak points of each one of them.

The rest of the paper is organized as follows: Section 2 presents the different functionalities provided by sensor MAC protocols, the parameters that have to be considered to design a good MAC protocol and shows some common metrics that need to be considered to evaluate its performances. Section 3 discusses the related surveys presented in this area. Based on the need of synchronization between neighboring nodes, this Section presents a taxonomy that is used to categorize the existing sensor MAC protocols. Sections 4, 5 and 6 present synchronous, asynchronous and hybrid sensor MAC protocols, respectively. The main points of medium access scheme of all the reviewed protocols are then summarized in Section 7 - this section discusses and compares these protocols based on the evaluation metrics presented in Section 2. Finally, Section 8 draws the conclusions with some open research directions.

2. MAC protocol Functionality, Design and Metrics

At the end of network deployment, communication links between sensor nodes have to be established. Moreover, communication medium needs to be shared fairly and efficiently. These main points constitute the objectives that any medium access protocol has to achieve.

A. Mac Protocol Functionality

Depending on the network requirements and device capability, MAC protocol provides different functionalities. As discussed in [3] and [4], these functions can be noted as below:

- Control medium access by determining the winner of the medium at any time. Medium access represents the main function of wireless MAC protocols since

broadcasts easily cause data corruption through collisions.
- Define the frame format, the time frame, and perform data encapsulation and decapsulation for communications between devices.
- Ensure successful and reliable transmission between devices using acknowledgement (ACK) messages and retransmissions when necessary.
- Prevent frame loss through overloaded recipient buffers.
- Use error detection or error correction codes to control the amount of errors present in frames delivered to upper layers.

B. Mac Protocol Design

In traditional wireless ad hoc network, MAC protocols attempt to provide high throughput, low latency, fairness, and mobility management, but often have little or no consideration for energy conservation. In wireless sensor networks, where sensor nodes are characterized by their the limited resources, multi-hop operation mode, and different application requirements, MAC protocols however, must provide the best performance at the smallest amount of energy consumption due to the limited energy resources available to each sensor node. Nevertheless, energy efficiency and throughput are the major aspects that need to be considered in MAC protocol design for wireless networks. According to [6], [26], for designing a good MAC protocol for these networks, the following parameters have to be considered:
- *Energy Efficiency*: sensor nodes are battery powered and it is often very difficult to change or recharge batteries for these sensor nodes. Sometimes it is beneficial to replace the sensor node rather than recharging them.
- *Latency*: this parameter basically depends on the application requirements. In some sensor network applications, the detected events must be reported to the sink node in real time so that the appropriate action could be taken immediately.
- *Throughput*: depends on the application requirements. Some sensor network applications require sampling the information with fine temporal resolution. In such sensor applications it is better that sink node receives more data.
- *Fairness*: related to the limited bandwidth, it is necessary to ensure that the sink node receives information from all sensor nodes fairly.

C. Mac Protocol Metrics

To evaluate the performance of MAC protocols, the research community considers some common metrics that need to be considered [5-6]. However, each protocol has some other specific metrics related to its design that also need to be evaluated. The common metrics are:
- *Energy consumption per bit (joules/bit):* can be defined as the total energy consumed divided per the total bits transmitted. Energy consumption is affected by all the major sources of energy waste in wireless sensor network such as idle listening, collisions, control packet overhead and overhearing.
- *Average delivery ratio*: is the number of packets received by the sink to the number of packets sensed by each node and sent over the network towards the sink node.
- *Average Packet Latency*: is the average time taken by the packets to reach to the sink node.
- *Network Throughput*: is defined as the total number of packets delivered at the sink node per time unit.

D. Medium Access Methods

In wireless sensor networks, controlling access to the channel, generally known as multiple access control, plays a key role in determining channel capacity utilization, network delays and more important, power consumption. It also influences congestion and fairness in channel usage. CSMA (Carrier Sense Multiple Access) and TDMA (Time-Division Multiple Access) are the most controlling channel access methods in wireless sensor networks.

CSMA (Carrier Sense Multiple Access) is the simplest form of medium access control in which nodes can transmit at any time as long as there is no contention [11]. CSMA can be non-persistent or *p*-persistent. In non-persistent CSMA, a wireless channel has to sample before any data transmission to determine if another device has already started transmitting. If the channel is busy, a backoff operation has to perform before attempting to transmit again. When the channel is free, sensor node transmits its data immediately. In *p*-persistent CSMA, sensor node continues to sense the channel when the channel is busy instead of delaying and checking again later. When the channel becomes free, sensor node transmits its data with probability *p* and delays the transmission with probability (1-*p*). An extended version of CSMA, called CSMA with collision avoidance (CSMA/CA) attempts to avoid collisions by using a control message exchange to reserve the wireless channel before each data message transmission using the RTS/CTS (Request to Send / Clear to Send) mechanism. This method is usually more used. It does not require clock synchronization and global topology information. Dynamic node joining and leaving are handled gracefully without extra operations. However, RTS/CTS mechanism incurs high overhead of the channel capacity in sensor networks [9], [34] because, data packets are typically very small in sensor networks.

TDMA (Time-Division Multiple Access) is a common scheduling method which schedules transmission times of neighboring nodes to occur at different times. Each sensor node transmits data during its own time slot [28]. Thus, it can solve the hidden terminal problem without extra message overhead. However, TDMA has many disadvantages [27] like clock synchronization and scalability problem.

3. Classifications of Sensor MAC Protocols

MAC protocols for wireless sensor networks can be classified into several categories based on the medium access mechanism. In [2], two classes have been provided: contention-based protocols and schedule-based protocols. In [3], the authors classify the MAC protocols with the same manner as was presented in [2] and also they provide one more sub-class under the two broad categories. Based on how neighboring nodes organize access to the shared medium, the MAC protocols are classified in [22] into random access, slotted access, frame-based, and hybrid protocols. Another classification is given in [6], where the authors provide a thematic taxonomy and classify MAC protocols according to the problems dealt with: scheduled protocols, protocols with a common active period, preamble sampling protocols, and hybrid protocols. In [7], the authors broadly classify the MAC protocols for wireless sensor networks into contention-based protocols, contention-free (scheduled-based) protocols, hybrid protocols and preamble sampling protocols.

Another comprehensive state-of-the-art study of WSN MAC protocols is provided in [5]. In this study, the authors provide a thematic taxonomy in which sensor MAC protocols are classified according to the dealt problems. The studied protocols are classified into three categories: scheduled protocols, protocols with common active periods, and hybrid protocols. The survey presented in [45] explores the extent to which existing MAC protocols for WSNs can serve for mission-critical applications. The analyzed protocols are classified according to data transport performance and suitability for mission-critical applications. Therefore, the following two main categories are used: delay-aware with four sub-classes (node-to-node decrease, node-to-node guarantee, end-to-end decrease and end-to-end guarantee) and reliability-aware with node-to-node increase, node-to-node guarantee, end-to-end increase and end-to-end. In [46], the authors detail the evolution of WSN MAC protocols with four categories: asynchronous, synchronous, frame-slotted, and multichannel. These protocols have been evaluated in terms of energy efficiency, data delivery performance, and overhead needed to maintain a protocol's mechanisms. MAC strategies for cognitive radio networks have also been surveyed in [48]. This survey shows the fundamental role of the MAC layer and identifies its functionalities in a cognitive radio network. Classification of the cognitive MAC protocols is proposed with two main categories: Direct Access Based (DAB) and Dynamic Spectrum Allocation (DSA). This work also discusses the advantages, drawbacks, and further design challenges of cognitive MAC protocols.

In [43], the authors focused their study on timeliness issues of slotted contention-based MAC protocols and provide a comprehensive review and taxonomy of synchronous MAC protocols. Based on the delay efficiency, the authors classify these protocols into two main categories: static schedule and adaptive schedule. Dealing with mobility can pose many challenges in protocol design; especially, at the MAC layer. These barriers require mobility adaptation algorithms to localize mobile nodes and predict the quality of link that can be established with them. In this context, the authors in [47] survey the current state-of-art in handling mobility. They describe the existing mobility models and patterns; and analyze the challenges caused by mobility at the MAC layer. In [49], the authors outline the sensor network properties that are crucial for the design of MAC layer protocols and study some MAC protocols without giving any classification.

Now, after having some background, here we present a discussion on several representative MAC protocols proposed in the previous works. As shown in Figure 1, three general classes for sensor network MAC protocols can exist. These classes are principally based on one key parameter. Some MAC protocols require that sensor nodes have to be synchronized to perform their functions. However, some other protocols do not need this requirement and let sensor nodes have medium access without synchronization. Moreover, other MAC protocols allow sensor nodes to switch between these two previous modes according to the traffic behavior and the sent packet type.

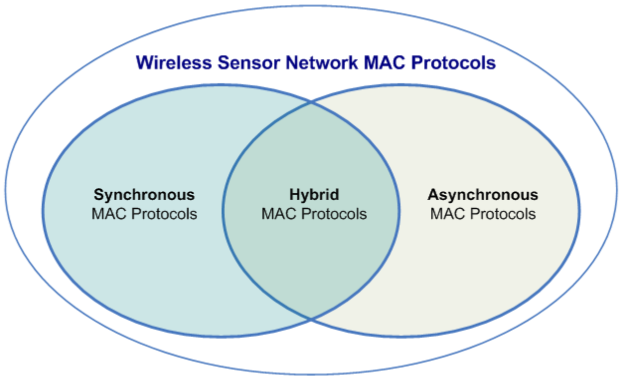

Figure 1. Classification of Sensor MAC protocols.

Synchronous MAC protocols attempt to organize nearby sensor nodes so their communications occur in an ordered fashion. The most common synchronization method organizes sensor nodes using time TDMA, where a single sensor node utilizes a time slot. Organizing sensor nodes provides the capability to reduce collisions and message retransmissions at the cost of synchronization and state distribution.

Asynchronous protocols attempt to conserve energy by allowing sensor nodes to operate independently with a minimum of complexity and without clock synchronization. While collisions and idle listening may occur and cause energy loss, these kinds of MAC protocols typically do not share information or maintain state.

The hybrid MAC protocols combine the two previous classes by allowing sensor nodes to use both synchronous and asynchronous mode.

4. Synchronous Sensor MAC protocols

Synchronous MAC protocols attempt to reduce energy consumption by coordinating sensor nodes with a common program. This can be done by establishing transmission schedules statically or dynamically to allow nodes to transmit data packets without collisions [50]. Most of the synchronous sensor MAC protocols use some form of TDMA because the other forms of multiple access, such as frequency or code division, would increase the cost and power requirements of the sensor nodes [55], [92]. By using a common program, the MAC protocol specifies which sensor nodes should utilize the channel at any time and thus, limits or eliminates collisions, idle listening, and overhearing. Nodes not participating in communication with its neighbors may enter in the sleep mode until they have a message to transmit or to receive and thus, can optimize energy consumption. In the existing literature, several synchronous MAC protocols for WSNs have been proposed like [1], [23], [26], [29], [50-61]. Some of these solutions have been broadly surveyed in [3] and are divided into four subclasses, Priority-based, Traffic-based, Clustering-based, and Slotted TDMA. Synchronous sensor MAC protocols have one common aspect. Before any data packet transmission, the peer neighboring sender and receiver nodes have to be synchronized [93]. Based on this aspect, we regard to these protocols as only one category and we survey in this section, the most significant of these protocols.

A. Channel Access Protocols

In [23], the authors proposed three protocols using the priority of nodes or links calculated from a random function to permit the channel access. The random function uses sensor node IDs and time slot numbers as input values to establish the priority within a two-hop neighborhood. The sensor nodes share their neighbor information and each sensor node maintains information about its two-hop neighborhood.

The first protocol proposed is called Node Activation Multiple Access (NAMA). NAMA uses distributed time division, time is divided into blocks of S_b sections and each section is divided into P_S parts. The parts contain T_P time slots. A node i chooses only one part p_i, during which to contend for a time slot to transmit data packets. The choice of a part is dependent on the density of neighbors already using that part, usually decided when the node joins a network. In this protocol, the last section of each block is reserved for signaling messages that allow sensor nodes to join the network. Each sensor node calculates its priority, compares it with the priority of its neighbors and determines who has access to the current time slot within the sensor node's chosen part. If a sensor node has the highest priority among its two hop neighbors for the given time slot, then the sensor node may transmit.

The second protocol called Link Activation Multiple Access (LAMA) is a time-slotted code division medium access scheme using Direct Sequence Spread Spectrum (DSSS) [24-25] code assigned to the receiver and the priority of the transmitter. Each sensor node gets a code assigned from a finite set of pseudo-noise codes. During each time slot, the sensor node with the highest priority in a two-hop neighborhood, calculated based on sensor node ID as in NAMA protocol, may activate a link by using the code assigned to the receiver. Using orthogonal codes allows sensor nodes to communicate when they would normally interfere and using the neighborhood information prevents collisions at the receiver.

The third protocol called Pairwise-link Activation Multiple Access (PAMA) is also a time-slotted link activation protocol based on a code division multiplexing scheme using DSSS code [24-25]. The links between sensor nodes can be activated by assigning priorities to the links and by varying the codes and priorities of links based on the current time slot. A communication link between two sensor nodes can be established if the link between the source (*s*) and the destination (*d*) node has the highest priority among all links of nodes *s* and *d*, and node source has the highest priority of its two-hop neighbors using the code assigned to link (*s*, *d*). Using DSSS allows nodes to communicate on different codes without interruption and the protocol algorithm prevents collisions on the same code by using the neighborhood information.

The main advantage of these protocols is the collision avoidance and the sensor nodes need only local information for channel access decision. But, the major drawback of these protocols is in the resources required. All the protocols require a sensor node to calculate the priorities of each neighboring sensor node and for each time slot; which consume more energy resources and decrease the network lifetime. Also, in LAMA and PAMA, the sensor nodes need to have radios with spread spectrum capabilities, which increases sensor node cost.

B. Sensor MAC protocols (S-MAC)

This protocol is specifically designed for wireless sensor networks. The protocol S-MAC [26-27] aims to reduce energy consumption, while supporting good scalability and collision avoidance. S-MAC tries to reduce energy consumption from all the sources that cause energy waste, like idle listening, collision, overhearing and control overhead. S-MAC consists of three major components: periodic listen and sleep, collision and overhearing avoidance, and message passing.

The basic scheme of S-MAC is shown in Figure 2. Each node goes to *sleep* for some time, and then wakes up and listens to find if any other node wants to talk to it. During sleep, the node turns off its radio, and sets a timer to awake itself later. The listening and sleeping time duration can be selected according to different application scenarios. For simplicity, S-MAC uses the same values for all the nodes.

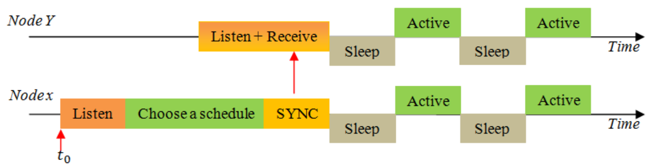

Figure 2. S-MAC basic scheme.

Before starting its periodic listen and sleep, the sensor node has to choose a schedule and exchange it with its neighbors. First, the sensor node listens for a certain amount of time. If it does not hear a schedule from another node, it randomly chooses a time to go to sleep and immediately broadcasts its schedule in a SYNC message, indicating that it will go to sleep after *t* seconds.

If the node receives a schedule from a neighbor before choosing its own schedule, it follows that schedule by setting its schedule to be the same. It then waits for a random delay t_d and rebroadcasts this schedule, indicating that it will sleep in $(t - t_d)$ seconds. Neighboring nodes form virtual clusters to set up a common sleep/active schedule. If two neighboring nodes reside in two different virtual clusters, they wake up at listen periods of both clusters.

To maintain synchronization among neighboring nodes, the sensor nodes periodically transmit SYNC messages at the beginning of the active period. The SYNC messages allow sensor nodes to learn their neighbors' schedules so they can wake up at the proper time to transmit a message. To improve performance, however, sensor nodes adopt the schedule of their neighbors in several cases. If a node currently does not have a schedule and hears a SYNC message, it adopts the schedule and joins the virtual cluster. If a sensor node hears multiple, sufficiently different schedules, it adopts them all so as to allow communications between different virtual clusters. A sensor node that does not hear any SYNC messages from neighbors chooses its own schedule. In order to detect new schedules, sensor nodes periodically listen for a longer time period that enables them to detect neighboring schedules with high probability. Each sensor node performs a simple contention avoidance algorithm based on a random backoff to limit the number of SYNC message collisions.

To receive both SYNC packets and data packets, the listen period is divided into two parts. The first part is reserved to send or receive SYNC packets, and the second one for sending or receiving RTS/CTS packets, as shown in Figure

3. If a sensor node wants to send a SYNC packet, it starts carrier sense (CS) [27] 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 the time slot, it sends its SYNC packet. The sensor node follows the same procedure when sending RTS, CTS, DATA and ACK packets.

The RTS and CTS packets contain the message transmission time, including time for the ACK packet, which permits the other neighboring nodes that are not concerned with this communication to sleep until the end of the transmission. S-MAC has been improved by the same authors in [27]. The authors introduce the adaptive listening technique, where nodes, in the same virtual cluster, that overhear a CTS, can wake up at the end of the data transmission to possibly act as the next hop. By doing this, the sensor nodes may transfer a message across two hops per frame time and decrease the latency. The authors also introduce a message fragmentation option, called message passing that allows sensor nodes to transmit relatively larger messages as smaller fragments using a single RTS/CTS exchange. Thus, if one fragment becomes corrupt due to collision or channel error, only the small fragment needs to be retransmitted instead of the entire data message.

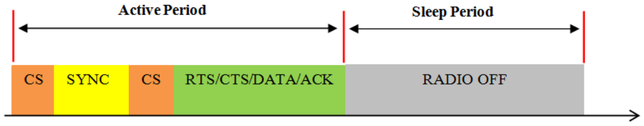

Figure 3. S-MAC frame format.

S-MAC offers several advantages like introducing the active /sleep period allows optimization of energy consumption. The concept of message-passing, where long messages are divided into small frames also decreases the energy consumption. The free synchronization method minimizes the problem of coordinating sensor nodes for communication and may provide adequate synchronization and clustering functionality for other protocols. We find also that S-MAC algorithm requires modest resources, such as memory for schedule offsets and timers for wakeup. Moreover, S-MAC can scale easily since the sensor nodes do not require any scalability coordination. S-MAC only coordinates neighbors using beacon messages, so sensor nodes do not have to forward or share large amount of state information.

However, S-MAC has some disadvantages. Sensor node can follow multiple schedules, which results in more energy consumption via idle listening and overhearing; the border nodes may die faster and cause segmentation along the borders of the virtual clusters. The static duty cycle of S-MAC can consume more energy and limit the protocol's performance. The duty cycle can be set based on expected application requirements, but S-MAC does not adapt to environment changing. Also, S-MAC does not expect to control virtual cluster size throughout the network. Varying cluster sizes have several impacts on the protocol's performance and large clusters can increase the message latency.

C. Timeout MAC protocol (T-MAC)

T-MAC protocol [1] extends the protocol S-MAC. T-MAC tries to reduce the idle listening by using a variable active period instead of using a fixed duty cycle schedule. To maintain an optimal active time under variable load, T-MAC dynamically determines its duration. Every node periodically wakes up to communicate with its neighbors and then, goes to sleep again until the next frame. Meanwhile, new messages are queued. Nodes communicate with each other using a RTS/CTS/DATA/ACK mechanism, which provides both collision avoidance and reliable transmission. In T-MAC, 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 an additional period or timeout (TA).

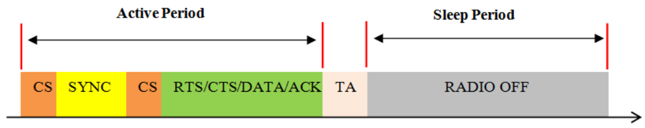

Figure 4. T-MAC frame format.

Figure 4 shows a T-MAC frame in which each node starts its frame by waiting and listening. If it hears nothing for a certain amount of time (CS), it chooses a frame schedule and transmits a SYNC packet, which contains the time until the next frame starts. If the node, during the CS time, hears a SYNC packet from another node, it follows the schedule in that SYNC packet and transmits its own SYNC accordingly. Nodes retransmit their SYNC once in a while. This allows new and mobile nodes to adopt an existing schedule.

If a node has a schedule and hears a SYNC packet with a different schedule from another node, it must adopt both schedules. It must also transmit a SYNC with its own schedule to the other node, to let the other node know about the presence of another schedule. After synchronization, sensor node starts its data transmission if the channel is still free during CS time. The active period ends in each case if any event occurs during the TA time.

To improve message latency, T-MAC introduces a new term called Future Request To send (FRTS) message to solve the same problem addressed by the adaptive listening technique of S-MAC. Sensor nodes can use an FRTS packet to inform the next hop that it has a future message to transfer. If a node receives a CTS packet destined for another node, it sends immediately an FRTS packet. The FRTS packet contains the length of the data that will be sent. A node must not send an FRTS packet if it senses communication right after the CTS. A node that receives an FRTS packet knows it will be the future destination of an RTS packet and must be awake by that time. The node can determine this from the timing information included previously in the FRTS packet.

To avoid collision between the FRTS and the data packet that follows the CTS, the data packet must be postponed for the duration of the FRTS packet. The initial sender of RTS should send a small Data-Send (DS) packet to preserve the channel during the FRTS duration. After the DS packet, it must immediately send the normal data packet. T-MAC also considers the buffer size priority of the sensor node and gives the possibility to control the channel to the sensor node that has a full buffer. This sensor may immediately send an RTS message after receiving an RTS message from another sensor node, which allows it to can limit buffer overflow.

D. Adaptive Coordinated Medium Access Control (AC-MAC)

AC-MAC [29] introduces the adaptive duty cycle scheme within the framework of S-MAC. This protocol tries to improve latency and throughput in high traffic loads situation while remaining as energy-efficient as S-MAC. As shown in Figure 5, AC-MAC based on the number of packets queued at the MAC layer, allows sensor nodes that have queued packets to introduce multiple data exchange periods using one SYNC frame.

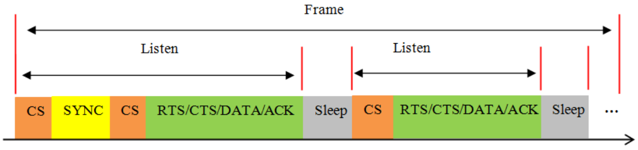

Figure 5. AC-MAC frame format.

In the beginning of each duty cycle, each sensor node calculates the number of the message queued in its MAC layer and announces this value in the first RTS packet sent within the SYNC frame. Sensor nodes that receive this RTS message can then calculate the duty cycle to use within the virtual cluster for the current SYNC period.

To optimized latency and throughput, AC-MAC provides sensor nodes with many buffered messages a priority; each sensor node calculates its random backoff value from a contention window whose size varies inversely proportional to the amount of traffic it has buffered. To simplify the protocol, sensor nodes only adopt one schedule per SYNC period.

E. Pattern MAC (PMAC)

P-MAC [30] is a '*time slotted*' protocol like S-MAC. P-MAC adjusts its duty cycle based on traffic conditions allowing sensor nodes with more data to utilize more slots than sensor nodes that have no data to transmit. In S-MAC, a node can stay awake for certain duration of a time slot, and go to sleep in the remaining duration. In P-MAC, a node can either be awake or asleep during a time slot.

In this protocol, sensor nodes share their proposed sleep and awake times for the next frame through a pattern sharing procedure. A sensor node gets information about the activity in its neighborhood beforehand 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 tries to save more power than that of SMAC and TMAC, without compromising the throughput.

A *sleep-wakeup* pattern is a string of bits (zero or one) indicating the tentative sleep-wakeup schedule for a sensor node over several slot times. Bit 1 in the string indicates that the node intends to stay awake during a slot time, while 0 indicates that the node intends to sleep. For example, a pattern of 0010 for a node indicates that the sensor node tentatively plans to be asleep for two consecutive slot times, stay awake in the third and go to sleep in the forth. This pattern is only a tentative plan and it can be changed according to the patterns of its neighboring nodes. Consequently, sleep-wakeup schedule for a node is derived from its own pattern and, the patterns of its neighboring nodes. Also, the pattern is defined as string of N bits indicating the tentative sleep-wakeup schedule during the M time slots of the upcoming frame. If $(N < M)$ then the pattern has to expand to fill the entire frame. For example, if N= 0010 and M= 10 time slots, the tentative pattern will be then, 0010001000.

In order to adapt to the traffic load, a node's pattern is updated during each period using the local traffic information available at the node and exchanged at the end of each period. P-MAC uses pattern technical update similar to TCP (Transmission Control Protocol) window growth and each node generates its update pattern according to the following sequence:

$$\mathbf{1, 01, 001, 0001, \ldots\ldots 0^{ß}1, 0^{ß}\,01, 0^{ß}001, \ldots\ldots\ldots 0^{N-1}1}$$

In the first period, the working pattern of each node is 1, which expects that the traffic load is high at the beginning and each node should be awake. If there is no data to send during the first time slot of bit 1, then it indicates that the traffic load is potentially low and sensor node should update its pattern to 01. With the same manner, if the sensor node has no data to send during the second time slot, it updates its pattern to 001. This update continues by increasing the number of 0 until the number of 0 bits in the updated pattern reaches a predefined threshold ß. After this threshold, the number of 0 bits could increase if there is no data to send during period 1 until the number of 0 bits reaches N time slots. A sensor node's pattern immediately increases to 1 whenever it has messages to send. Sensor nodes constantly update their pattern based on current conditions, but remain in operation according to the previously shared schedule. The sensor node shares its current pattern in the pattern exchange slots at the end of a frame using CSMA.

Node's pattern is performed and exchanged according to PMAC frame presented in Figure 6. PMAC frame consists of two sub-frames. The first is called Pattern Repeat Time Frame (PRTF), during which each node repeats its current pattern and during N time slots, these time slots are reserved to send data. At the end of these N slots, PRTF has one additional time slot during which all the sensor nodes stay awake. This special time slot is used to speed up communication and to broadcast messages, which occurs after the regular data slots.

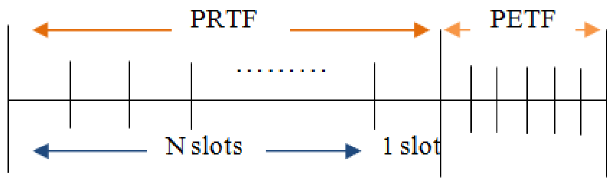

Figure 6. PMAC frame format.

The second sub-frame is called Pattern Exchange Time Frame (PETF), during which new patterns are exchanged between neighbors. PETF is also divided into various time slots reserved to exchange the new patterns generated during PRTF at each node to reflect the latest traffic information. The last generated pattern during a particular PRTF becomes the pattern for the next PRTF, and will be advertised to the neighbors during the PETF. The pattern is cyclically repeated during PRTF such that each time slot has one pattern bit assigned. Patterns received from its neighbors during the preceding PETF are also repeated in the same way. If a node receives no new patterns from some of its neighbors during the preceding PETF (probably due to collisions), it then repeats its old patterns.

PMAC offers a simple way to advertise messages and form schedules between sensor nodes in a neighborhood. The capability to quickly adapt to changing traffic conditions may also make PMAC an attractive choice for a sensor network deployment. However, the schedule generation algorithm has several possible disadvantages. First, some sensor nodes may not receive an updated pattern due to channel errors while others correctly receive the update. This may lead to different schedules present in the same neighborhood and cause collisions, idle listening, and wasted transmissions. Also, the functionality of the protocol relates directly to the traffic intensity. Each time the sensor node operates in an active time slot, it performs the pattern update algorithm. During times of high traffic intensity, the processing requirements may become large as the sensor node operates in many active time slots.

5. Asynchronous Sensor MAC protocols

Unscheduled MAC protocols offer the advantage of simplicity, without having to maintain and share state of neighboring nodes. Protocols in this category wake up the next hop node by continuously sending preambles or packets, and thus eliminate the synchronization overhead. A good number of asynchronous protocols are proposed as MAC layer solutions for WSNs. Some of these protocols have been surveyed in [43] and discussed under six categories. Static wake-up preamble [34-35], [37], [61-70], [72], Adaptive wake-up preamble [32-73], [74-79], Collaborative schedule setting [80-82], Collision resolution [83-84], Receiver initiated [65], [85-87], and Anticipation based [39], [88-90]. Earlier, in [3], these protocols were classified into four categories: Multiple transceiver [10], Multiple path [8], Event-centered [38], and Encounter-based [65]. Irrespective of the subclass to which a protocol belongs, synchronous MAC protocols use the same techniques, such as channel sensing and channel reservation messages to mitigate the effects of the common problems like a higher rate of collision, idle listening, and overhearing. For this reason, we prefer to simplify the classification and consider these protocols under only one main category. Now, let us know about the most import asynchronous MAC protocols.

A. Berkeley MAC Protocol (B-MAC)

B-MAC [34] is a contention based MAC protocol. Like [35] and [36], B-MAC uses a preamble to wakeup sleeping neighbors. Sensor nodes, in this protocol, independently follow a sleeping schedule based on the target duty cycle for the sensor network. Since the sensor nodes operate on independent schedules, B-MAC uses very long preambles for message transmission. The preamble length is provided as parameter to the upper layer. The source node transmits a long enough preamble causing the destination node wake up and sensing it. Sensor nodes that sense this signal remain awake to receive the data following the preamble or return to sleep if they do not detect activity on the channel. Before transmitting, sensor nodes wait a random period of time to prevent any collision.
Figure 7 shows the communication mechanism of B-MAC. If a source node wishes to transmit, it precedes the data packet with a preamble that is slightly longer than the sleep period of the destination node. So, the destination node, at some point during the transmission of the preamble, will wake up and detect the preamble; it has to remain awake to receive the data packet.

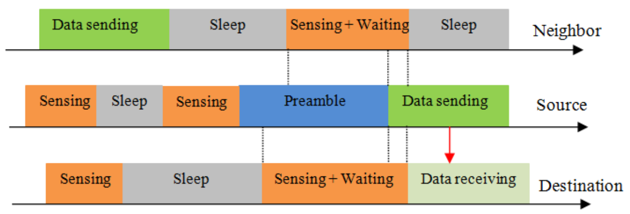

Figure 7. B-MAC communication mechanism.

B-MAC 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 differentiate between a noise and 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. This mechanism permits to increase the reliability of channel assessment and provides a great deal of flexibility through a protocol interface that allows the sensor node to change many operating variables in the protocol, such as delay and backoff values.
A key challenge of B-MAC is implementing check intervals that are very short, which ensures 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.
The Low Power Listening (LPL) approach used by B-MAC which employs a long preamble is suboptimal in terms of energy consumption, is subject to overhearing, as well as it introduces excess latency at each hop [37]. 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 which 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.

B. WiseMAC Protocol

WiseMAC [32-33] is CSMA-based medium access control protocol. It uses the preamble sampling technique [31] to minimize power consumption when listening to an idle medium. In this 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 preamble transmission time (T_p) is initially set to be equal to the sampling period (S_p).

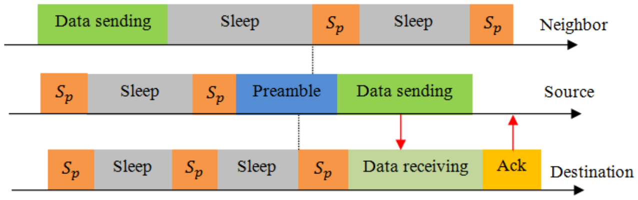

Figure 8. WiseMAC communication mechanism.

However, the receiver may not be ready at the end of the preamble, due to reason like interference, which causes the possibility of over-emitting type energy waste. Moreover, over-emitting 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. This method uses the knowledge of the sleep schedules of the neighboring nodes. The nodes learn and refresh their neighbor's sleep schedule during every data exchange as part of the acknowledgement message. So, each node keeps a table of sleep schedules of its neighbors. Based on this table's information, the sender node schedules transmissions by choosing the minimum requirement preamble. To decrease the possibility of collisions caused by that specific start time of wake-up preamble, a random wake-up preamble is advised. To avoid the clock drift between the source and the destination, a lower bound for the preamble length (T_p) is calculated as the minimum of destination's sampling period (S_p) and the potential clock drift with the destination which is a multiple of the time since the last ACK packet arrival. Figure 8 shows the communication mechanism of this protocol. WiseMAC has been extended a bit in [13]. This improvement allows a common destination node to automatically stay awake at the end of the wake-up period, when more traffic has to be handled, which improves the delay cases.
According to the simulation results [32], WiseMAC performs better than one of the S-MAC variants. Moreover, 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.
However, the decentralized sleep-listen scheduling which results in different sleep and wake-up times for each neighboring node represents the main drawback of WiseMAC. This is especially an important problem for broadcast type of communication, since broadcasted packet will be buffered for neighbors in sleep mode and delivered many times as each neighbor wakes up. However, this redundant transmission will result in higher latency and more power consumption. In addition, WiseMAC may suffer more from the hidden terminal problem. That is because WiseMAC is 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 same communication range.

C. A Short Preamble MAC (X-MAC)

X-MAC [37] is a low-power MAC protocol that strives to overcome the shortcomings of the long preamble used by B-MAC [34]. X-MAC uses a shortened preamble approach and includes the ID of the target sensor node in the preamble. So, non-target receivers can realize that they are not concerned by this transmission and quickly go back to sleep. This solution addresses the overhearing problem. However, 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 (S_p) packets, each containing the ID of the target node (Figure 9). 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.

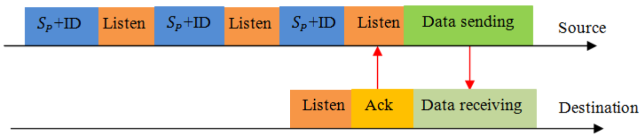

Figure 9. X-MAC communication mechanism.

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 the short excessive preamble, which reduces per-hop latency and energy spent unnecessarily waiting and transmitting. However, the non-target receivers, after going back to sleep, may wake up and sense the medium for several periods while the data transmission is not yet achieved which wastes energy for these nodes.

D. Spatial Correlation-based Collaborative MAC protocol (CC-MAC)

CC-MAC [38] attempts to conserve energy, while fulfilling application requirements, by utilizing the knowledge that sensor nodes located near each other generate correlated measurements. To achieve energy savings, CC-MAC filters measurements from highly correlated sensor nodes in an effort to reduce the number of messages the sensor network must handle.
The authors introduce an analytical framework to investigate the relation between the positions of sensor nodes in the event area and the event estimation reliability. Based on analysis within the framework, the authors introduce the Iterative Node Selection (INS) algorithm that creates a sample topology for the sensor network to exploit spatial correlation and filter correlation between the nodes. Thus, INS creates a correlation region defined by its correlation radius, based on statistical information about the sensor network deployment. Sensor nodes closer than the correlation radius produce correlated data. Therefore, if a node transmits data, the nodes in its correlation region are not required to send data. This algorithm is executed by the sink during the network initialization to calculate and distribute the correlation radius throughout the network.

CC-MAC consists of two components: the Event MAC (E-MAC), which filters sensor node measurements to reduce traffic and the Network MAC (N-MAC), which forwards the filtered measurements to the sensor network sink. More specifically, E-MAC is executed when sensor node wants to transmit its sensed data to the sink, while N-MAC is performed when a node receives data from another node and tries to forward it to the next hop.

E-MAC reduces the traffic generated in an area by having only sensor nodes separated by at least the correlation distance measurements. Other nodes periodically sleep to save energy and awake to forward messages. Correlated sensor nodes rotate the role of generating measurements to balance energy consumption throughout the network. Sensor nodes get elected as active nodes and to represent the correlated sensor nodes by winning contention for the wireless medium. E-MAC modifies the standard RTS/CTS/DATA/ACK scheme by introducing a First Hop (FH) bit into the control packet headers. The sensor node actively reporting measurements sets the FH bit when it transmits messages so that other nodes can decide to generate measurements or not. If a sensor node does not belong to any correlation region, it will then begin to also generate measurements. Once the originating sensor node has transmitted its sensed data, the FH bit gets cleared and the message will be forwarded using N-MAC protocol.

After removing the redundant data present in multiple measurements by E-MAC, N-MAC forwards it from source nodes to the sink. However, the forwarded traffic may become more important. To compensate for this, N-MAC protocol transmissions take preference over E-MAC transmissions through the use of smaller backoff windows and inter-packet times in same way that the PCF (Point Coordination Function) in IEEE 802.11 receives preferential access to the wireless channel over the DCF (Distributed Coordination Function).

The simulation results show that CC-MAC can achieve a good balance of low energy consumption and favorable traffic performance compared to the other protocols. Additionally, the analytical framework proposed in this protocol allows users to apply the CC-MAC protocol to applications with various data fidelity requirements. CC-MAC, however, requires that sensor nodes possess or obtain ranging information about their neighbors in order for N-MAC to filter data from correlated sensor nodes.

The computational resources required by the INS algorithm may also limit the application of the protocol. For example, if the number of sensing events increases, the overhead associated with computing the correlation radius and distributing throughout the network increases. For large networks, this overhead may become significant.

E. Convergent MAC protocol (CMAC)

CMAC [39] uses unsynchronized wakeup scheduling with a predefined idle duty-cycle. In this wake-up scheduling scheme, the sleep period is fixed according to the duty cycle and active period. Instead of a long preamble to activate the receiver, CMAC uses aggressive RTS. To detect an RTS, nodes periodically wake up and double check the channel for activities. CMAC initially uses anycast to transmit packets to a potential forwarder that wakes up first. *Awake* candidate receivers will contend to be the anycast receiver by prioritizing their CTS transmissions according to their routing metrics to the sink. After receiving CTS, the data packet will be sent to the sender of the CTS immediately. To overcome the disadvantages of anycast, such as higher RTS/CTS overhead and route stretch, CMAC converges from anycast to unicast once it establishes contact with a receiver having a sufficiently good routing metric.

As discussed above, CMAC has three main components: Aggressive RTS equipped with double channel check for channel assessment, anycast to quickly discover a forwarder, and convergent packet forwarding to reduce the anycast overhead.

In the aggressive RTS, CMAC uses multiple RTS packets separated by fixed short gaps instead of a long preamble. The short gap allows receivers to send back CTS packets. CMAC sends all RTS packets without clear channel assessment (CCA) except the first one. In very low duty cycle, nodes must assess the channel very quickly each time they wake up. However, if the receiver wakes up during the gap between two RTS transmissions, it may miss the RTS burst. So the authors propose to use double channel check which works by assessing the channel twice with a fix short separation between them each time a node wakes up.

The anycast mechanism is used to send the RTS burst, where more than one node in the forwarding set may try to reply to the same RTS, and the one closest to the destination should be elected to receive the data packet. The CTS transmissions are prioritized according to the routing metrics of contending nodes. Nodes with better routing metrics sends CTS packets earlier, while other overhearing nodes cancel their CTS transmissions accordingly, and nodes that can make little progress are excluded.

To overcome the shortcomings of anycast like overhead of anycast RTS/CTS exchange, the authors propose convergent packet forwarding. In such mechanism, the node will remain awake for a short duration after receiving a data packet. If the latest anycast receiver has a routing metric close to the best, CMAC will use unicast and send the data directly to this node without using RTS/CTS packets.

The experiment and simulation results show at low duty cycles that CMAC achieves the throughput and latency performance and outperforms other energy efficient protocols like BMAC [34], SMAC [26] and GeRaF [18-20]. The issue here is that a lower duty cycle MAC protocol can save energy, but low activity levels place a limit on the protocol’s complexity, the possible network capacity, and the message latency.

6. Hybrid Sensor MAC protocols

Hybrid MAC protocols aim to leverage advantages and mitigate the disadvantages of the synchronous and asynchronous protocols two. These protocols try to adapt their behaviors according to the traffic loads and patterns. Several protocols of this category [12], [14], [21], [44] can be found in [7].

A. Zebra MAC Protocol (Z-MAC)

Z-MAC [17] combines TDMA and CSMA. It adapts to the level of contention in the network. Under low contention, it behaves like CSMA, and under high contention, like TDMA. Z-MAC uses CSMA as the baseline MAC scheme and uses a TDMA schedule to enhance contention resolution and assign time slot during the network deployment phase. The authors adopt a centralized channel reuse scheduling algorithm [16]. After the time slot assignment, each node reuses its assigned slot periodically in every predetermined period, called *frame*. The owner node is a node that is assigned to a time slot. The others are non-owners of that slot. In the adopted algorithm [16], more than one owner per slot may exist and two nodes further than two-hop neighborhoods can own the same time slot.

Z-MAC uses priority scheme to switch between CSMA and TDMA depending on the level of contention. A sensor node may transmit during any time slot. It samples the channel and transmits a packet when the channel is clear. However, an owner of that slot always has higher priority over its non-owners in accessing the channel. The owners are given earlier chances to transmit and their slots are scheduled a priori to avoid collision, but when a slot is not in use by its owners, non-owners can steal the slot.

During the deployment phase, each sensor node learns its two-hop neighbors, assigned to a time slot, chooses its time frame and forwards its frame size and slot number to its two-hop neighbors. In transmission phase, sensor node can be in one of two modes: Low Contention Level (LCL) or High Contention Level (HCL). A node is in HCL only when it receives an Explicit Contention Notification (ECN) message from a two-hop neighbor within the last period. Otherwise, the node is in LCL. A node sends an ECN when it experiences high contention based on the packet loss rate. ECN permits to avoid the problem of hidden host. Z-MAC uses backoff, CCA and LPL interfaces of B-MAC [34] to perform its LCL and HCL.

Z-MAC requires local clock synchronization only among neighboring senders and when they are under high contention (HLC). Z-MAC adopts a technique from RTP/RTCP (Real-Time Transport Protocol) [15] to implement this synchronization. The performance results show that Z-MAC outperform B-MAC under medium to high contention while it shows competitive, but slightly less performance than B-MAC under low contention (especially, in terms of energy efficiency). According to the authors, Z-MAC finds its utility in applications where expected data rates and two-hop contention are medium to high.

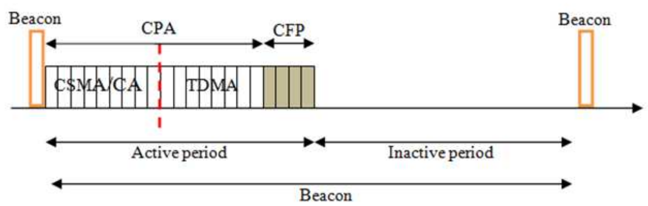

Figure 10. Hybrid-MAC communication mechanism.

B. CSMA/TDMA hybrid MAC Protocol (hybrid-MAC)

In this protocol [41], the authors propose a hybrid MAC protocol based on IEEE 802.15.4 standard [42]. Their main idea consists of adding a dynamic TDMA period into contention access period of this standard. IEEE 802.15.4 standard has two operational modes, non-beacon mode and beacon-enabled mode. In non-beacon mode, channel access is only based on CSMA/CA. But in beacon-enabled mode, a beacon frame has to transmit in specified time intervals by the coordinator. The beacon frame is divided into an active and an inactive period. Devices are in sleep mode during the second period for energy saving purposes. The active period also consists of a Contention Access Period (CAP) and a Contention Free Period (CFP).This beacon permits the devices synchronize themselves for accessing the channel. The authors use the beacon-enabled mode and propose a modified beacon frame by assigning TDMA slots to sensor nodes during the CAP period (Figure 10).

To determine the limit between TDMA and CSMA in CAP period, the authors consider two parameters: channel utilization level in CAP and the amount of pending data in nodes' queues. To determine the queue state, eight different values have been utilized by using three reserved bits of the standard data packet header. These values define the queue state level meter which indicates the fraction of queue being occupied. Thus, each node gives the coordinator a more accurate description of its queue state. The coordinator maintains the queue state of network nodes in local array structure. Each array cell belongs to one network node and has the initial value of 0. Each time a data packet is received the coordinator checks the queue state of the sending node and updates its corresponding array cell. The channel utilization is evaluated as a simple function of number of used slots, number of unused slots and number of slots having collision.

Two cases explain the decreasing of channel utilization, increasing collisions and reduction of used slots. In the first case, the coordinator checks the queue state array values and assigns TDMA slots to the nodes in descending order of their queue state values. In the second case, when the low channel utilization is caused by the decreasing of the used slots number and low collisions, the TDMA period length should be shortened.

In this protocol, when a TDMA slot is assigned to a node, it will not be authorized to send data in the CSMA/CA period in the same beacon frame. Thus, number of nodes participating in the contention is decreased and fewer collisions occur. Moreover, TDMA slots are only assigned to nodes having queued data, thus permitting to avoid the common problem of under-utilization of slots in TDMA methods. However, the main advantage of this protocol resides in the use of coordinator node.

7. Critical Issues to Know About Performance

In this study, we have noticed that each MAC protocol proposed for sensor networks tries to minimize energy consumption by reducing collisions, limiting idle listening, and overhearing. Synchronous MAC protocols require that sensor nodes expend some considerable part of their energy to share state and maintain synchronization.

Additionally, the extent and frequency to which the sensor network undergoes organization and reorganization can greatly affect its performance. However, the protocols of this class may allow sensor nodes to remain asleep for longer periods of time and forward messages with less effort than those using unscheduled MAC protocols since the sensor node has some indication of its neighbor's active/sleep schedules. Nevertheless, Synchronous MAC protocols have obvious disadvantage in scalability and adaptability. This is because, sensor nodes need to strictly follow the assigned communication time slots in case of TDMA scheme and require frame synchronization which involves the complicated tasks of slot allocation and schedule maintenance. Figure 11, Figure 12, and Figure 13 provide qualitative performances of MAC protocols. Each curve qualitatively represents the tendency of different performance metrics (energy, latency and throughput) as the traffic load changes. The dashed horizontal line qualitatively represents the medium level of each metric. Figure 11 presents the qualitative performances of synchronous MAC protocols. This kind of protocol has a good performance in terms of energy consumption when the network traffic load is low. The energy-saving performance decreases with an increasing traffic load and reaches an unacceptable level when the network's lifetime becomes too short.

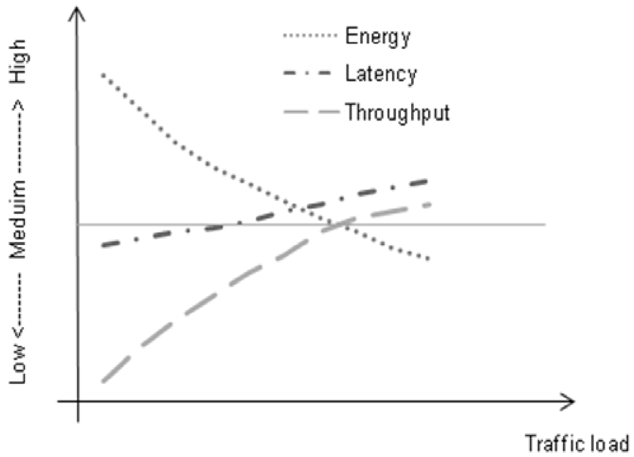

Figure 11. General performance understanding of Synchronous MAC protocols performances.

With the increasing traffic load, on the other hand, the latency is increasing due to a heavy traffic load [91], and end-to-end delay will be prolonged. Thus, the throughput curve is moving up and eventually stabilized at certain level when the traffic load becomes high due to the queuing traffic.

Asynchronous MAC protocols are characterized by their simplicity. Neighboring sensors nodes do not need to schedule common active/sleep plan; thus, minimizing resource utilization within a sensor node and eliminating the protocol overhead. However, coordinating neighboring sensor nodes for communication becomes a primary function of these protocols. Using the preamble to establish the communication between neighboring nodes eliminates the synchronization need but may increase the overhearing and thus, energy is wasted due to the long preamble. For some protocol, this energy waste can be mitigated using short preamble.

Hybrid MAC protocols combine the qualities of the two previous classes. The first class presents collision-free, high channel utilization and throughput, which are suitable for high traffic load situations. The second one possesses simplicity, flexibility and robustness; hence, enabling nodes to be adaptable to network topology change. Hybrid MAC protocol may switch its process between the synchronous and asynchronous mode according to the traffic loads. According to the experiment done in [40], hybrid protocol can help the monitoring system to give an almost real-time answer to the emergencies. Figure13 shows the qualitative performances of the hybrid MAC protocols. These protocols give an intermediate performance level comparing with the pervious protocols.

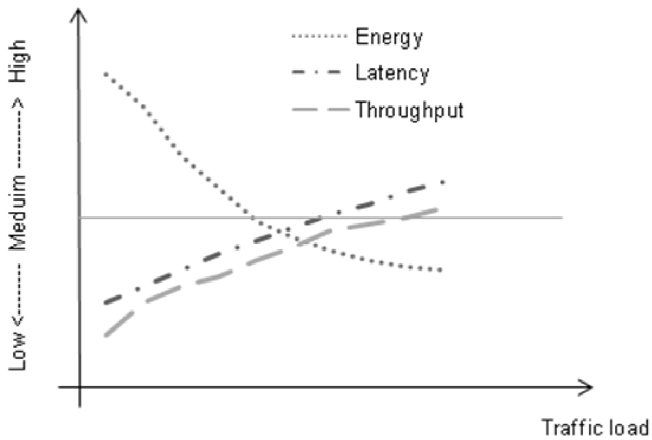

Figure 12. General performance understanding of Asynchronous MAC protocols performances.

Nevertheless, the short preamble is not a complete solution to make the protocol adaptive to the traffic load, especially in case of low duty cycle where the number of wasted short preambles can be increased. Figure 12 shows the qualitative performances of these protocols. Comparing with synchronous protocols, the energy curve reaches the dashed horizontal line early. The energy-saving is more influenced by the Overhearing and the hidden host problem. On the other hand, the latency and the throughput are ameliorated due to the adaptive active /sleep period.

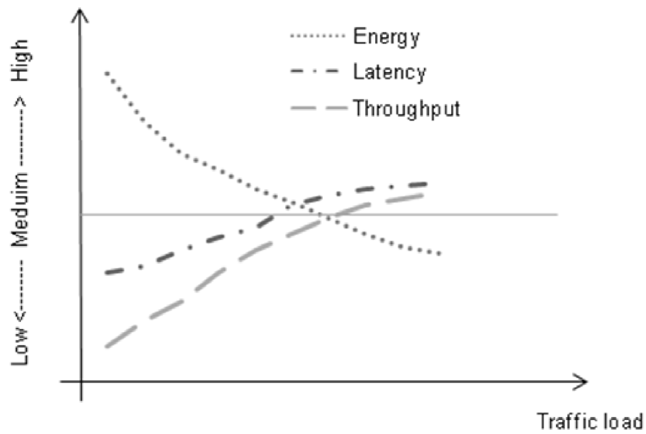

Figure 13. Hybrid MAC protocols performances.

The requirements for the MAC protocol in WSNs are clearly different from those for traditional wireless networks. WSNs are often application-specific. While there are typical applications over WSNs, various applications still exhibit their own peculiarities in the usage pattern of networks. So, MAC layer protocols for WSNs need to be adaptable enough to accommodate a variety of traffic patterns and to coordinate with the network and application layers to adapt radio usage to application communication patterns and achieve higher energy savings.

8. Conclusions

In this study, we have covered many MAC protocols proposed so far for sensor networks, but many more exist. Based on synchronization mode, the studied protocols have been classified on three classes: synchronous, asynchronous and hybrid MAC protocols. In the first class, sensor nodes

change the radio state periodically and need clock synchronization to maintain common active/sleep schedule. Moreover, node may have pre-assigned time slots to transmit the data but each node has to listen to the time slots of its neighbors in order to synchronize. In the second class, sensor node has its own active/sleep schedule that can be established based on the medium activity. Sensor node may adjust its active/sleep time according to the medium state. Hybrid protocols combine both synchronous and asynchronous modes - senor node may switch its communication mode according to the traffic behavior.
Throughout this work, energy consumption and latency have been considered as the main studied parameters. Each protocol provides benefits for certain applications or under certain conditions according to the chosen design. But, there are still many more challenges that need to be solved for MAC protocols. It is still needed to find out a suitable solution for real-time support. Optimal or the best achievable energy efficiency also remains an open issue of great interest. It would surely be welcome if a general, flexible MAC protocol could be designed that supports various applications and operating environments while consuming minimal power and offering acceptable traffic characteristics.

Acknowledgement

This work was supported in part by NDC Lab, KICT, IIUM.

References

[1] T.v. Dam, K. Langendoen, “An adaptive energy-efficient MAC protocol for wireless sensor networks,” in Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys), 2003, pp. 171-180.
[2] B. Krishnamachari, “Medium access and sleep scheduling,” Book chapter in Networking wireless sensors, ISBN-13: 978-0-521-83847-4, Cambridge University Press, December 22, 2005, pp. 82-102.
[3] K.I. Kredo, P. Mohapatra, “Medium access control in wireless sensor networks,” Computer Networks, vol. 51, no. 4, March 2007, pp. 961-994.
[4] J.F. Kurose and K.W. Ross. Computer Networking: A Top-Down Approach Featuring the Internet third edition. ISBN-13: 978-0321269768, Addison Wesley, 2005.
[5] A. Bachir, M. Dohler, T.Watteyne, and K. Leung, “MAC essentials for wireless sensor networks,” IEEE Commun. Surv. Tutorials, vol.12, no. 2, Second Quarter 2010, pp. 222–248.
[6] R. Yadav, S. Varma, and N. Malaviya, “A Survey of MAC Protocols for Wireless Sensor Networks,” UbiCC Journal, vol.4, no.3, August 2009, pp.827-833.
[7] Y. Z. Zhao, C. Miao, M. Ma, J.B. Zhang, and C. Leung, “A survey and projection on medium access control protocols for wireless sensor networks,” ACM Computing Surveys, vol. 45, no. 1, November 2012, pp. 1-37.
[8] I. Chatzigiannakis, A. Kinalis, S. Nikoletseas, “Wireless sensor networks protocols for efficient collision avoidance in multi-path data propagation,” In Proceedings of the ACM International Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, 2004, pp. 8–16.
[9] A. Woo and D. Culler, “A transmission control scheme for media access in sensor networks,” In ACM MobiCom, 2001, pp. 221-235.
[10] S. Singh, C. Raghavendra, “PAMAS – power aware multi-access protocol with signaling for ad hoc networks,” SIGCOMM Computer Communications Review, vol. 28, issue. 3, 1998, pp. 5-26.
[11] L. Kleinrock, F.A. Tobagi, “Packet switching in radio channels: Part I- Carrier sense multiple-access modes and their throughput-delay characteristics,” IEEE Transactions on Communications, vol. 23, no.12, 1975, pp.1400-1416.
[12] Y.Z. Zhao, M. Ma, C.Y. Miao, and T. N. Nguyen, “An energy-efficient and low-latency MAC protocol with adaptive scheduling for multi-hop wireless sensor networks,” Computer Communications, Vol. 33, 2010, pp. 1452-1461.
[13] P. Hurni, T. Braun, and M. Anwander, “Evaluation of WiseMAC and Extensions on Wireless Sensor Nodes,” Springer Telecommunication Systems Journal, vol.43, no.1-2, February 2010, pp. 49-58.
[14] G.S. Ahn, E. Miluzzo, A.T. Campbell, S.G. Hong, and F. Cuomo, “Funneling-MAC: A localized, sink-oriented MAC for boosting fidelity in sensor networks,” In ACM SenSys, 2006, pp. 293-306.
[15] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A transport protocol for real-time applications,” Audio-Video Transport Working Group, RFC 1889, Jan. 1996. Available at: http://www.ietf.org/rfc/rfc1889.txt [last accessed: 15 August, 2013]
[16] I. Rhee, A.Warrier, J.Min, and L. Xu, “DRAND: Distributed randomized TDMA scheduling for wireless ad hoc networks,” In Proc. ACM MobiHoc, New York, 2006, pp. 190–201.
[17] I. Rhee, A. Warrier, M. Aia, J. Min, and M. Sichitiu, “Z-MAC: A Hybrid MAC for Wireless Sensor Networks,” IEEE/ACM Transactions on Networking, vol. 16, no. 3, June 2008, pp. 551-524.
[18] P. Casari, A. Marcucci, M. Nati, C. Petrioli, and M. Zorzi, “A detailed simulation study of geographic random forwarding (GeRaF) in wireless sensor networks,” In Proceedings of the Military Communications Conference (MILCOM), 2005, pp. 17-20.
[19] M. Zorzi, and R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Energy and latency performance,” IEEE Transactions on Mobile Computing, vol. 2, no. 4, 2003, pp 349-365.
[20] M. Zorzi, and R. Rao, “Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Multihop performance,” IEEE Trans, Mobile Comput, Vol. 2, no. 4, pp. 337-348, 2003.
[21] G.P. Halkes, T. Van Dam, and K.G. Langendoen, “Comparing energy-saving MAC protocols for wireless sensor networks,” Mobile Networks and Applications Journal, Volume 10 Issue 5, October 2005, pp. 783–791.
[22] K. Langendoen and G. Halkes, “Energy-Efficient Medium Access Control,” Book chapter in Embedded Systems Handbook (R. Zurawski Ed.), ISBN-13: 978-1420074109, CRC Press, 2009, pp. 34.1-34.29.

[23] L. Bao and J. Garcia-Luna-Aceves, “A new approach to channel access scheduling for ad hoc networks,” in Proceedings of the International Conference on Mobile Computing and Networking (MobiCom), 2001, pp. 210–221.
[24] J. Garcia-Luna-Aceves and J. Raju, “Distributed assignment of codes for multihop packet-radio networks,” In MILCOM Proceedings, Monterey, CA, USA, 2-5 Nov. 1997, pp. 450-461.
[25] M. Joa-Ng and I. Lu, “Spread spectrum medium access protocol with collision avoidance in mobile ad-hoc wireless network,” In IEEE INFOCOM '99, New York, NY, USA, 21-25 March 1999, pp. 776-783.
[26] W.Ye, J. Heidemann, and D. Estrin, “An energy-efficient MAC protocol for wireless sensor networks,” Proceedings of the Joint Conference of the IEEE Computer and Communications Societies (INFOCOM), vol. 3, 2002, pp. 1567–1576.
[27] W. Ye, J. Heidemann, and D. Estrin, “Medium access control with coordinated adaptive sleeping for wireless sensor networks,” IEEE/ACM Transactions on Networking vol.12, no. 3, 2004, pp 493–506.
[28] Z. Li, B. Zhao, Y. Qu, and K. Chen , “An adaptive and distributed STDMA Scheme for Ad Hoc and Sensor Networks,” International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, 2006. ICN/ICONS/MCL, 23-29 April 2006, p. 121.
[29] J. Ai, J. Kong, and D. Turgut, “An adaptive coordinated medium access control for wireless sensor networks,” in Proceedings of the International Symposium on Computers and Communications, vol. 1, 2004, pp. 214–219.
[30] T. Zheng, S. Radhakrishnan, and V. Sarangan, “PMAC: An adaptive energy-efficient MAC protocol for wireless sensor networks,” in Proceedings of the IEEE International Parallel and Distributed Processing Symposium, 2005, pp. 65–72.
[31] A. El-Hoiydi, “Spatial TDMA and CSMA with preamble sampling for low power ad hoc wireless sensor networks,” Proceedings of ISCC 2002, Seventh International Symposium on Computers and Communications, 1-4 July 2002, pp. 685 - 692.
[32] C. C. Enz, A. El-Hoiydi, J-D. Decotignie, and V. Peiris, “WiseNET: An Ultralow-Power Wireless Sensor Network Solution,” IEEE Computer, vol. 37, Issue. 8, August 2004, pp. 62-70.
[33] A. El-Hoiydi, and J.-D. Decotignie, “WiseMAC: an ultra low power MAC protocol for multi-hop wireless sensor networks,” In Proceedings of the International Workshop on Algorithmic Aspects of Wireless Sensor Networks (Algosensors), Lecture Notes in Computer Science, 2004, pp. 18–31.
[34] J. Polastre, J. Hill, and D. Culler, “Versatile low power media access for wireless sensor networks," In Proceedings of the International Conference on Embedded Networked Sensor Systems (SenSys’04), 2004, pp. 95-107.
[35] C. Schurgers, V. Tsiatsis, S. Ganeriwal, and M. Srivastava, “Optimizing sensor networks in the energy-latency-density design space,” IEEE Transactions on Mobile Computing, vol. 1, no.1, 2002, pp. 70-80.
[36] A. El-Hoiydi, “Aloha with Preamble Sampling for Sporadic Traffic in Ad-hoc Wireless Sensor Networks,” In Proceedings of IEEE International Conference on Communications, vol.5, April 2002, 3418 - 3423.
[37] 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), 2006, pp. 307-320.
[38] M.C. Vuran and I.F. Akyildiz, “Spatial correlation-based collaborative medium access control in wireless sensor networks,” IEEE/ACM Transactions on Networking vol. 14, no. 2, 2006, pp. 316-329.
[39] S. Liu, K. Fan, and P. Sinha, “CMAC: An Energy-Efficient MAC Layer Protocol Using Convergent Packet Forwarding for Wireless Sensor Networks,” ACM Transactions on Sensor Networks, vol. 5, no. 4, Article 29, November 2009.
[40] Q. Zhang, Y. Liua, H. Guoa, Q. Zhang, “The Design of Hybrid MAC Protocol for Industry Monitoring System Based on WSN,” Procedia Engineering, Elsevier, Volume 23, 2011, Pages 290–295.
[41] M.H.S. Gilani, I. Sarrafi, M. Abbaspour, “An adaptive CSMA/TDMA hybrid MAC for energy and throughput improvement of wireless sensor networks,” Ad Hoc Networks, vol. 11, issue. 4, 2011, pp. 1297-1304.
[42] IEEE Standard 802.15, “Wireless Medium Access Control and Physical Layer Specification for Low Rate Wireless Personal Area Networks,” IEEE Std. 802.15, Part 15.4, 2006.
[43] M. Doudou, D. Djenouri, and N. Badache, “Survey on Latency Issues of Asynchronous MAC Protocols in Delay-Sensitive Wireless Sensor Networks,” IEEE Communications Surveys & Tutorials, Volume: 15, Issue: 2, Second Quarter 2013, pp. 528-550.
[44] M. Salajegheh, H. Soroush, and A. Kalis, “HYMAC: Hybrid TDMA/FDMA medium access control protocol for wireless sensor networks,” In Proceedings of the 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC’07), IEEE, Washington, D.C., 2007, pp.1–5.
[45] P. Suriyachai, U. Roedig, and A. Scott, “A Survey of MAC Protocols for Mission-Critical Applications in Wireless Sensor Networks,” IEEE Communications Surveys & Tutorials, Volume: 14, Issue: 2, Second Quarter 2012, pp. 240-264.
[46] P. Huang, L. Xiao, S. Soltani, M.W. Mutka, and N. Xi, “The Evolution of MAC Protocols in Wireless Sensor Networks: A Survey,” IEEE Communications Surveys & Tutorials, Volume:15, Issue: 1, First Quarter 2013 , pp. 101-120.
[47] Q. Dong, and W. Dargie, “A Survey on Mobility and Mobility-Aware MAC Protocols in Wireless Sensor Networks,” IEEE Communications Surveys & Tutorials, Volume: 15, Issue: 1, First Quarter 2013, pp. 88-100.
[48] A.D. Domenico, E.C. Strinati, and M.G.D. Benedetto, “A Survey on MAC Strategies for Cognitive Radio Networks,” IEEE Communications Surveys & Tutorials, Vol. 14, No. 1, First Quarter 2012, pp. 21-44.
[49] I. Demirkol, C. Ersoy, and F. Alagoz, "MAC protocols for wireless sensor networks: A survey," IEEE Communications Magazine, Vol. 44, issue. 4, September 2006, pp. 115-121.

[50] V. Rajendran, K. Obraczka, and J.J. Garcia-Luna-Aceves, "Energy-efficient, collision-free medium access control for wireless sensor networks," Wireless Networks, Vol. 12, issue. 1, Feb 2006, pp. 63–78.
[51] Y.E. Sagduyu and A. Ephremides, "The problem of medium access control in wireless sensor networks," IEEE Wireless Communications, vol. 11, issue. 6, 2004, pp. 44-53.
[52] P. Lin, C. Qiao, and X. Wang, "Medium access control with a dynamic duty cycle for sensor networks," In Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), vol. 3, 2004, pp. 1534-1539.
[53] L.F.W. van Hoesel and P.J.M. Havinga, "A lightweight medium access protocol (LMAC) for wireless sensor networks: reducing preamble transmissions and transceiver state switches," In: 1st International Workshop on Networked Sensing Systems (INSS'04), Tokyo, Japan, pp. 205-208.
[54] S. Chatterjea, L. van Hoesel, and P. Havinga, "AI-LMAC: an adaptive, information-centric and lightweight MAC protocol for wireless sensor networks," In Proceedings of the Intelligent Sensors, Sensor Networks, and Information Processing Conference, 2004, pp. 381-388.
[55] C. K. Nguyen, and A. Kumar, "Medium access control with adjustable sleeps for wireless sensor networks," In Proceedings of the 11th IEEE Symposium on Computers and Communication (ISCC '06), IEEE, Washington, D.C., 2006, pp. 270-276.
[56] C. K. Nguyen, and A. Kumar, "An energy-aware medium-access-control protocol with frequent sleeps for wireless sensor networks," In Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC '05), IEEE, Washington, D.C., 2005, pp. 386–391.
[57] S. Du, A.K. Saha, and D.B. Johnson, "RMAC: A routing-enhanced duty-cycle MAC protocol for wireless sensor networks," In Proceedings of the 26th Annual IEEE Conference on Computer Communications, 2007, pp. 1478-1486.
[58] Y. Sun, S. Du, O. Gurewitz, and D. B. Johnson, "DW-MAC: A low latency, energy efficient demand wakeup MAC protocol for wireless sensor networks," In Proceedings of the 9th ACM International Symposium on Mobile Ad Hoc Networking and Computing (MobiHoc '08), ACM, New York, 2008, pp. 53-62.
[59] S. Yessad, F. Nait-Abdesselam, T. Taleb, and B. Bensaou, "R-MAC: Reservation medium access control protocol for wireless sensor networks," In Proceedings of the 32nd IEEE Conference on Local Computer Networks, IEEE, Washington, D.C., 2007, pp. 719-724.
[60] V. Rajendran, J.J. Garcia-Luna-Aceves, and K. Obraczka, "Energy-efficient, application-aware medium access for sensor networks," In Proceedings of the 2005 International Conference on Mobile Ad Hoc and Sensor Systems Conference (MASS 2005), 2005, pp. 623-630.
[61] T. Wu, and S. Biswas, "A self-reorganizing slot allocation protocol for multi-cluster sensor networks," In Proceedings of the 4th International Symposium on Information Processing in Sensor Networks, IEEE, April 2005, pp. 309-316.
[62] H. Wang, X. Zhang, F. Nat-Abdesselam, and A. A. Khokhar, "DPSMAC: An Asynchronous MAC Protocol for Wireless Sensor Networks," In Proc. 14th International Conference of High Performance Computing (HiPC'07), Springer-Verlag Berlin, Heidelberg, 2007, pp. 393-404.
[63] P. Kumar, M. Gunes, Q.Mushtaq, and B.Blywis, "A real-time and energy-efficient MAC protocol for wireless sensor networks," International Conference on Wireless and Optical Communications Networks (WOCN'09), Cairo , Egypt, April 28-30, 2009, pp. 1-5.
[64] S. Mahlknecht and M. Boeck, "CSMA-MPS: A Minimum Preamble Sampling MAC Protocol for Low Power Wireless Sensor Networks," in IEEE Workshop on Factory Communication Systems, Vienna, Austria, Sept. 22-24, 2004, pp. 73-80.
[65] E.-Y. Lin, J. Rabaey, and A. Wolisz, "Power-Efficient Rendez-vous Schemes for Dense Wireless Sensor Networks," in IEEE International Conference on Communications, Vol. 7, June 20-24, 2004, Paris, France, pp. 3769-3776.
[66] R. Braynard, A. Silberstein, and C. Ellis, "Extending Network Lifetime Using an Automatically Tuned Energy-Aware MAC Protocol," Proc. European Workshop on Wireless Sensor Networks (EWSN), Zurich, Switzerland, 2006, pp. 244-259.
[67] L. Bernardo, R. Oliveira, M. Pereira, M. Macedo, and P. Pinto, "A Wireless Sensor MAC Protocol for Bursty Data Traffic," In 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Athens, Greece, Sept. 3-7, 2007, pp. 1-5.
[68] D. Moss and P. Levis. BoX-MAC: Exploiting Physical and Link Layer Boundaries in Low-Power Networking. Technical Report SING-08-00, 2008.
[69] K.J. Wong and D. Arvind, "Speckmac: Low-power decentralized MAC protocol low data rate transmissions in Specknets," Proc. 2nd International Workshop on Multi-hop Ad Hoc Networks: from Theory to Reality (REALMAN'06), ACM SIGMOBILE, May 2006, Italy, pp. 71-78.
[70] C.J. Merlin and W.B. Heinzelman, "Schedule Adaptation of LowPower-Listening Protocols for Wireless Sensor Networks," IEEE Transactions on Mobile Computing, May 2010, Vol. 9, Issue: 5, pp. 672-685.
[71] D. Lee and K. Chung, "RA-MAC: An Energy Efficient and Low Latency MAC Protocol Using RTS Aggregation for Wireless Sensor Networks," International Conference on Advanced Technologies for Communications (ATC), Hanoi, Oct. 6-9, 2008, pp 150-153.
[72] I. Joe and H. Ryu, "A Patterned Preamble MAC Protocol for Wireless Sensor Networks," IEEE International Conference on Computer Communications and Networks (ICCCN07), Honolulu, HI, 2007, pp. 1285-1290.
[73] P. Hurni, and T. Braun, "MaxMAC: A Maximally Traffic-Adaptive MAC Protocol for Wireless Sensor Networks," European Conference on Wireless Sensor Networks (EWSN), Coimbra, Portugal, February 17-19, 2010, pp 289-305.
[74] J. Kim, J. On, S. Kim, and J. Lee, "Performance Evaluation of Synchronous and Asynchronous MAC

Protocols for Wireless Sensor Networks," Second International Conference on Sensor Technologies and Applications (SENSORCOMM'08), Cap Esterel, France, Aug. 2531, 2008, pp. 500-506.

[75] Y.C. Chang, J.-R. Jiang, J.-P.Sheu, and H.Y. Shih, "ADCA: An Asynchronous Duty Cycle Adjustment MAC Protocol for Wireless Sensor Networks," Global Telecommunications Conference IEEE GLOBECOM 08, New Orleans, LO, Nov.30-Dec. 4, 2008, pp. 383-3870.

[76] J. Polastre. A Unifying Link Abstraction for Wireless Sensor Networks. Ph.D. dissertation, University of California, Berkeley, October 2005.

[77] P. Hurni, T. Braun, and M. Anwander, "Evaluation of WiseMAC and Extensions on Wireless Sensor Nodes," Telecommunication Systems Journal, Springer, February 2010, Vol.43, No.1-2, pp. 49-58.

[78] X. Shi and G. Stromberg, "SyncWUF: An Ultra Low-Power MAC Protocol for Wireless Sensor Networks," IEEE Transactions on Mobile Computing,, Vol. 6, Issue 1, Jan. 2007, pp. 115-125.

[79] M. Miller and N. Vaidya, "A MAC Protocol to Reduce Sensor Network Energy Consumption Using a Wakeup Radio," IEEE Transactions on Mobile Computing,, Vol. 4, No. 3, May-June 2005, pp. 228-242.

[80] Q. Yu, C. Tan, and H. Zhou, "A Low-Latency MAC Protocol for Wireless Sensor Networks," International Conference on Wireless Communications, Networking and Mobile Computing (WiCom07), Shanghai, Sept. 21-25, 2007, pp. 2816-2820.

[81] I. Gragopoulos, I.Tsetsinas, E.Karapistoli, and F.-N.Paulidou, "Performance Study of a New MAC Algorithm to Reduce Energy and Delay in Wireless Sensor Networks," New Technologies, Mobility and Security, Springer, 2007, pp. 39-51.

[82] W.-H. Liao and H.-H. Wang, "An asynchronous MAC protocol for wireless sensor networks," Journal of Network and Computer Applications, 2008, Vol. 31, No. 4, pp. 807-820.

[83] K. Jamieson, H. Balakrishnan, and Y. C. Tay, "Sift: A MAC Protocol for Event-Driven Wireless Sensor Networks," Third European Conference on Wireless Sensor Networks (EWSN06), Zurich, Switzerland, February 2006, Vol. 3868, pp. 260-275.

[84] I. Demirkol and C. Ersoy, "Energy and delay optimized contention for wireless sensor networks," Computer Networks: The International Journal of Computer and Telecommunications Networking, August 2009, Vol. 53,Issue 12, pp. 2106-2119.

[85] Y. Sun, O. Gurewitz, and D.B. Johnson, "RI-MAC: a receiver-initiated asynchronous duty cycle mac protocol for dynamic traffic loads in wireless sensor networks," Proc. 6th ACM conference on Embedded network sensor systems (SenSys'08), 2008, pp. 1-14.

[86] P. Dutta, S. Dawson-Haggerty, Y. Chen, C. Liang, and A. Terzis, "Design and Evaluation of a Versatile and Efficient Receiver-Initiated Link Layer for Low-Power Wireless," In Proc. 8th ACM Conference on Embedded Networked Sensor Systems (Sensys'10), Zurich, Switzerland, Nov. 3-5, 2010, pp. 1-14.

[87] P.G. Park, C. Fischione, A. Bonivento, K.H. Johansson, and A. Sangiovanni-Vincentelli, "Breath: a self-Adapting protocol for wireless sensor networks in control and automation," IEEE SECON, San Francisco, CA, USA, 2008, pp. 323-331.

[88] S. Rashwand, J.V. Misic, V.B. Misic, S. Biswas, and Md.M. Haque, "A Novel Asynchronous, Energy Efficient, Low Transmission Delay MAC Protocol for Wireless Sensor Networks," ICDCS Workshops Proceedings, 2009, pp. 186-193.

[89] J. Kim, X. Lin, N.B. Shroff, and P. Sinha, "On Maximizing the Lifetime of Delay-Sensitive Wireless Sensor Networks with Anycast," in Proc. IEEE INFOCOM, Pheonix, AZ, April 2008, pp. 807-815.

[90] J. Kim, X. Lin, and N.B. Shroff, "Optimal Anycast Technique for Delay-Sensitive Energy-Constrained Asynchronous Sensor Networks," IEEE INFOCOM, 2009, pp. 612-620.

[91] M. Guerroumi, A.-S.K. Pathan, N. Badache, and S. Moussaoui, "Strengths and Weaknesses of Prominent Data Dissemination Techniques in Wireless Sensor Networks", International Journal of Communication Networks and Information Security, Vol. 5, No. 3, pp. 158-177.

[92] M. Hussaini, H. Bello-Salau, A. F. Salami, F. Anwar, A. H. Abdalla, Md. Rafiqul Islam, "Enhanced Clustering Routing Protocol for Power-Efficient Gathering in Wireless Sensor Network," International Journal of Communication Networks and Information Security, Vol. 4, No. 1, April 2012, pp. 18-28.

[93] N. Meghanathan, "Link Expiration Time and Minimum Distance Spanning Trees based Distributed Data Gathering Algorithms for Wireless Mobile Sensor Networks," International Journal of Communication Networks and Information Security, Vol. 4, No. 3, December 2012, pp. 196-206.