Multi-channel Medium Access Control Protocols for Wireless Sensor Networks: A Survey

GholamHossein EkbataniFard
Department of Computer Engineering, Lahijan Branch, Islamic Azad University, Guilan, Iran
ekbatanifard@iau-lahijan.ac.ir

Received: 2011/03/23 ;Accepted: 2011/04/29 Pages: 21-45

Abstract

Extensive researches on Wireless Sensor Networks (WSNs) have been performed and many techniques have been developed for the data link (MAC) layer. Most of them assume single-channel MAC protocols. In the usual dense deployment of the sensor networks, single-channel MAC protocols may be deficient because of radio collisions and limited bandwidth. Hence, using multiple channels can significantly improve the performance of WSN. Recent developments in sensor technology, as seen in Crossbow's MICAz Mote, Rockwell's WINS nodes and IEEE 802.15.4 Zigbee, have enabled support for multi-channel communications. Several multi-channel MAC protocols with different objectives have been proposed for WSNs in literature. This paper surveys and classifies the state-of-the-art multi-channel MAC protocols that are proposed for WSNs. It first outlines the sensor network properties that are crucial for designing a MAC protocol. It subsequently reviews the existent challenges to design a good multi-channel MAC protocol for the sensor networks. Then, several multi-channel MAC protocols specifically proposed for the WSNs are inspected in detail and compared with each other. Finally, some open issues in this area are outlined for future research.

Keywords: *Wireless Sensor Networks (WSNs), Medium Access Control (MAC), Multi-Channel*

1. Introduction

Wireless Sensor Networks (WSNs) [1] consist of resource constrained devices, called sensor node, that communicate wirelessly in a multihop nature. Sensor nodes are usually scattered in an ad hoc fashion over a geographic area. These can be uniformly or non-uniformly distributed with various densities depending on an application. Data are taken from elements sensed such as temperature, light, sound, motion, etc., and forwarded to a base station, named sink.

Popularity of WSNs is the result of technological advancements in Micro-Electro Mechanical Systems (MEMS) for sensing, microelectronics for computation and communication, and wireless networking techniques for efficient transmission [2]. Sensor nodes usually are equipped with a single transceiver utilizing a single channel. Therefore, WSNs cannot provide reliable and timely communication with high data rate requirements due to radio collisions and limited bandwidth. Here, the term "channel" is used upon a logical level, where it can be implemented as a frequency band under FDMA or an orthogonal code under CDMA.

With improvement in sensor hardware, support for multi-channel communication is already in an advanced stage of implementation. For example, Berkeley's third generation Mica2 Mote has an 868/916 MHz multi-channel transceiver [3]. In Rockwell's WINS nodes, the radio operates on one of 40 channels in the ISM frequency band, selectable by the controller [4]. The new IEEE 802.15.4 standard, popularly called as Zigbee, defines 11 channels at 868/915 MHz and 16 channels at 2.4 GHz [5] (between 2400 and 2483.5 MHz). MICAz and Telos use CC2420 radio that provides multiple frequencies. It provides 16 non-overlapping channels, with 5MHz spacing [3]. Therefore, it is necessary to design multi-channel MAC protocols to take full advantage of parallel transmission to enhance throughput and performance of WSNs.

Extensive researches on WSNs have been performed and many techniques have been developed for the data link (MAC) layer. Various aspects of MAC protocols for WSNs are discussed in several surveys [1-2, 6-12]. However, there is not any survey on multi-channel MAC protocols for WSNs. This motivates us to investigate the state-of-the-art multi-channel MAC protocols specially proposed for WSNs to provide this paper as a road map for future research.

There are various important parameters for designing a multi-channel MAC protocol for WSNs, among which is the limitation on the whole number of channels that can practically be used, due to interference with neighboring networks where some channels cannot be used. As well as, *channel assignment* method, which involves assigning channels to node interface(s), and providing *rendezvous point(s)* on which two communicating nodes should be on the same channel at the same time. The other parameters for designing a multi-channel MAC protocol for WSNs are discussed in detail in Section 3.

The rest of the paper is organized as follows: in Section 2, some MAC related characteristics of WSNs are introduced. Section 3 describes the existing challenges for designing a good multi-channel MAC protocol for WSNs. Section 4 surveys existing multi-channel MAC protocols that have been especially proposed for WSNs. These multi-channel MAC protocols are compared in section 5. Finally, in Section 6, open issues and future research directions are pointed out and concluding remarks are given.

2. MAC Related WSNs Characteristics

Here, various MAC layer related features of WSNs are given in detail. These features are: 1) Various methods used to access channels, 2) Different types of channels, 3) Single-radio channel assignment methods, and 4) Special attributes of a good MAC protocol for WSNs.

2.1. Source of Energy Wastage

One of the main concerns in WSNs is network lifetime where *energy loss* in each node is vital. The major sources of energy loss in a MAC protocol for WSNs are as: *Collision, Control Packet Overhead, Idle Listening, Overhearing, Overemitting.*

Energy consumption model also has a given energy wastage for transitions from sleep to active state and vice versa.

2.2. Channel Access Methods

Channel access methods employed in multi-channel MAC protocols can be divided into 1) *Schedule-based*, 2) *Contention-based,* and 3) *Contention-schedule-based* (*Hybrid*) protocols.

1. In *schedule-based* protocols, nodes switch between different channels, and coordination of channel switching is required between sender and receiver in order to be on the same channel at the same time. Schedule-based protocols have the natural advantage of collision free access to channel, which causes energy saving. However, there are some factors that limit the use of schedule-based protocols, such as scalability and adaptability to network dynamics. The main technique used in schedule-based protocols is *Time Division Multiple Access* (TDMA). In the TDMA-based protocols, nodes are able to communicate without packet collision or overhearing within each slot. Nevertheless, pure TDMA protocols are scarcely feasible in reality, because of the strict requirement of global synchronization. Burri et al. [14], to overcome the disadvantage of pure (global) TDMA, propose the local TDMA scheme such that synchronization is only strictly required in one hop.
2. In *contention-based* protocols, the most common channel-access approach is *Carrier Sense Multiple Access* (CSMA) paradigm. CSMA is described by its simplicity, flexibility, and adaptability to changes in the number of active nodes. No clock synchronization or global topology information are needed. However, contention-based protocols may fail to allocate the medium successfully and result in collision when the number of sources or the source transmitting rates increases.
3. In *hybrid* protocols, similar to schedule-based methods, the time is divided into slots. Here, nodes contend for accessing the time slots in spite of schedule-based protocols where time slots are assigned in advance. After completion of the contention phase and successful access to a time slot, node can transmit without any collisions.

2.3. Radio Types

Some MAC protocols are designed for a simple radio providing just a single channel. An alternative is to use a second, extremely low power radio that can be utilized for signaling an intended receiver to wake-up and turn on its primary radio to receive a data packet. In the most energy-efficient case, the second radio is capable of emitting a tone waking up all the neighboring nodes including the intended receiver (or waking up the intended receiver only in some advanced cases).

2.3.1. Single-Radio Channel Assignment Methods

In single-radio sensor nodes, the radio can divide the available bandwidth into multiple channels. Two common ways for doing so are *Frequency Division Multiple Access* (FDMA) and *Code Division Multiple Access* (CDMA). FDMA partitions the total bandwidth of the channel into a number of frequency bands, on which multiple nodes can transmit concurrently without collision. CDMA on the other hand, uses a single carrier in combination with a set of orthogonal codes. Data packets are XOR-ed with a specific code by the sender before transmission, and then XOR-ed again by the receiver with the same code to retrieve the original data. Receivers using another code see the transmission as (pseudo) random noise. This allows the concurrent and collision-free transmission of multiple messages.

Single-radio, multi-channel protocols can be classified according to the following channel assignment methods: i) *Fixed*, ii) *Semi-dynamic*, and iii) *Dynamic.*

i. In *fixed* assignment, radios are assigned permanent channels. The operating frequencies are not changed during communication, although the assignment of channels can be renewed, for example, due to changing interference circumstances.
ii. In *semi-dynamic* approach, channels are assigned to radios for constant use; however, it is possible to change the channel for communicating with radios that are assigned different channels.
iii. In *dynamic* channel assignment method, nodes can dynamically switch their interfaces from one channel to another between successive data transmissions. Dynamic channel assignment is further classified into three categories based on the methods of coordination as follows: a) *Dedicated Control Channel*, b) *Split Phase*, and c) *Frequency Hopping* [15].
    a. In *dedicated* control channel protocols, nodes synchronize by exchanging control packets on the dedicated control channel and negotiate for the channel to be used for data transmission.
    b. With the *split phase* approach, nodes access the medium in two phases: a control phase and a data transmission phase. During the control phase, all the nodes switch to a common control channel and negotiate with their intended receivers for the channel(s) to be used during the data transmission phase. Usually, during the control phase, access to the medium is contention-based.
    c. In *frequency-hopping* approaches, nodes hop, or in other words switch, between different channels. A pair of devices stop hopping when they make an agreement for transmission and return to the frequency hopping subsequently after transmission completes.

Note that in wireless mobile sensor networks, where devices are highly mobile, pair-wise frequency negotiation is required. Hence, it involves unnecessary overhead and is too costly if applied to static WSN.

2.4. A Good MAC Protocol Attributes

A very good MAC protocol for the WSNs should have attributes such as, 1) energy efficiency, 2) scalability, adaptability to changes in network topology, 3) low latency, 4) high throughput, 5) high bandwidth utilization, 6) fairness, 7) reliability, 8) collision free, 9) self-stabilizing and 10) do not require common global time reference. Moreover, the main goals for using multiple channels in WSN protocols are as follows:

- **Minimization of intra-network interference**. Radio is a shared medium, so concurrent transmissions in the same broadcast domain result in collisions and possibly packet losses. Hence, using multiple channels can reduce interference among sensor nodes.
- **Avoidance of external interference.** There are multiple radio standards operating in the same unlicensed frequency bands (e.g., 802.11, 802.15.4, and 802.15.1 all operate

in the 2.4 GHz band). Channel diversity is one method to mitigate inter-network interference among collocated networks.

- **Improvement of network throughput.** Utilizing parallel transmission in multi-channel protocols increases packet delivery rate and reduces the latency of network traffic.

3. Challenges for Multi-channel MAC Protocols

Alongside the traditional challenges of WSNs, such as energy efficiency and scalability, there are some other challenges in multi-channel WSNs. The nodes' transceivers can switch among several channels to gain more performance. It is necessary for senders and receivers to be on the same channel to communicate at a time. It is challenging to coordinate the distributed nodes to which channel they switch and in which state they are. As follows, we will show the challenging problems in designing multi-channel MAC protocols:

- **Multi-channel hidden terminal.** The problem occurs when a node (like C in Figure 1) is busy transmitting or receiving on a data channel then a neighboring node (like A in Figure 1) initiates a channel reservation handshake on the control channel. Because the node (C) is active on a data channel, it is unable to realize its neighboring selected channel and may inadvertently choose the same channel when it starts its next data exchange. Therefore, some protocols assign different available physical frequencies within two interference (communication) hops to avoid this problem.
- **Missing receiver or Deafness.** When a transmitter wants to send a packet to a receiver, which its transceiver happens to be tuned to another channel, the missing receiver problem occurs. In the missing receiver problem as shown in Figure 2, node A tries to communicate with node B by sending the RTS packet, while B is busy in data transfer on a different channel.

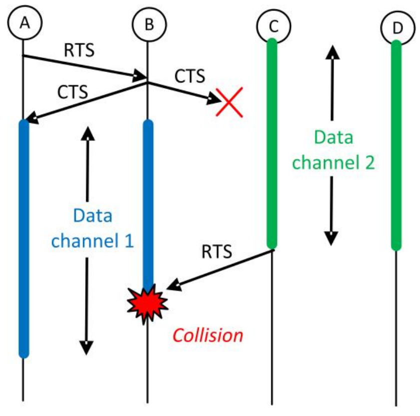

Figure 1. Multi-channel hidden terminal problem

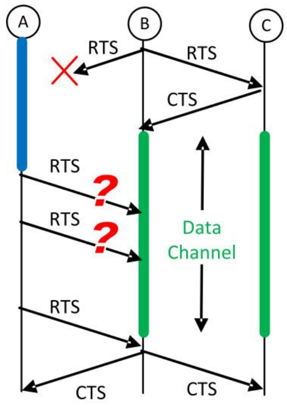

Figure 2. Missing receiver problem

- **Broadcast support.** The missing receiver problem causes another problem namely broadcast support. In single channel environment, the broadcast packet can be heard by all neighbor nodes. But, in multi-channel mode, when nodes in neighborhood are assigned different channels for packet reception, some nodes may miss the broadcast messages. Broadcast is important for WSN traffic, for instance, sensor nodes may require in-network processing before they transmit the data toward the sink or use local broadcast for route discovery.

- **Channel switching delay.** The radio cannot switch between the channels immediately where it takes some time, for instance, around 200 μs on CC2420 [3] radios. During that time, packets cannot be sent or received. In comparison with the shortness of the WSNs usual packet length, the overhead of the channel switching delay cannot be ignored.
- **Channel assignment method.** The task of channel assignment with minimum interference, sometime named as the 2-hop coloring problem, allows repetition of colors to occur only if the nodes are separated by more than two hops. This is a NP-complete problem and development of efficient heuristics for this problem is an open research area.
- **Interference among different channels.** Adjacent channel interferences have great impact on radio reception and hence cannot be neglected. The existence of adjacent channel interference can cause unexpected collisions hence, losing packets. The safe way is to use only non-adjacent channels in multi-channel protocols. IEEE 802.15.4 specification shows that one IEEE 802.11 channel can potentially collide with four IEEE 802.15.4 channels. Moreover, multi-channel protocols must have capabilities to work well with a small number of available channels. Otherwise, their performance may greatly degrade in some indoor scenarios.
- **Synchronization.** In a single-radio, multi-channel communication, synchronization is a big challenge. If the channel assignment is done dynamically, i.e., the radios are switching between channels instead of being fixed on one channel, a detailed coordination of channel switching is required between senders and receivers in order to communicate on the same channel at the same time.
- **Joining the network.** A new node joining the network may disrupt the channel organization or may be required to scan all the channels to find a suitable one for transmission. It is another challenge for multi-channel communication in WSNs.
- **Avoiding network partitioning.** If transceivers of nearby nodes are fixed on different frequencies, they cannot communicate with each other, therefore, it may cause network partitioning. Moreover, there are also other factors for network partitioning such as node failure, flat batteries, and the presence of obstacles.

4. Multi-channel MAC Protocols for WSNs

In this section, a range of multi-channel MAC protocols in WSN literature is described and their main shortcomings are outlined. Then, these protocols are classified according to some parameters and are compared with each other in the following section.

4.1. IEEE 802.15.4 Protocol

The IEEE 802.15.4 [16] protocol, which is originally designed for Wireless low-rate Personal Area Networks (WPAN), can be used for WSN applications. The protocol makes use of multi-channel communication to reduce the effects of interference due to co-existing networks that share same parts of the spectrum. The protocol has two modes of operation: *Beacon-enabled* and *Beaconless* modes. In the beacon-enabled mode, a coordinator node, which is a Full Function Device (FFD), is responsible for adjusting the channel, on which its end-devices, i.e., Reduced Function Device (RFD), should communicate, according to the interference experienced by the connected nodes to this coordinator. In this mode, communication can take place in a slotted mode of operation, (i.e., guaranteed timeslots can be allocated by the coordinator) and nodes should directly communicate with the coordinator to get the slot allocations. In this case, communication takes place on a single-hop network, such as a star. Even if a node intends to communicate with a peer in its communication range, all communication flows via the coordinator. When the protocol operates in a beaconless mode, it uses CSMA/CA and nodes operate on a fixed channel. A multi-hop network can be constructed by linking groups of star formations in the beacon-enabled mode. In this case, the beacon message should contain the device depth on the tree and the timing offset, such that a node selects a receiving schedule different from its parent.

Due to the hierarchy in the IEEE802.15.4 networks, the WPAN coordinator is responsible for binding of new nodes, scheduling and routing in the network. Additionally, in IEEE 802.15.4, since all the nodes in a WPAN, communicates on the same channel; contention within the network is not resolved.

4.2. CMAC Protocol

CMAC [17] is a desynchronized multi-channel MAC protocol that uses two radios. A Low power wakeup Radio (LR), in the lines of the Berkeley pico-radio that can only emit and receive a short train of wakeup pulses, and a Main half-duplex Radio (MR). LR is always on, used to monitor a node's default channel while MR is placed in the sleep mode thus conserving energy. LR radio is used only for wake up and not to be used as a second interface to send data. It plays two roles, first, when a node wishes to transmit, the receiver is woken up through a series of pulses and, second, channel negotiation is undertaken before MR is switched on. MR is able to tune dynamically to any of the pre-decided set of channels. It transmits at a constant power level in the selected band although; it can be switched off too. MR handles all data transmission/reception.

In [17] a Dynamic Channel Allocation (DCA) algorithm that uses coloring techniques for channel assignment has been proposed such that nodes have been allocated channels that overlap at 3 hops or more. DCA assigns channels in a distributed manner in order to make subsequent communication free from interference.

CMAC relies on three types of control messages for its operation: Request (REQ), Confirm (CONF), and Wait (WAIT), that all are sent and received through the LR. When a packet has to be transmitted, LR of the sender is tuned to the receiver's channel R. It monitors R for DIFS period to avoid collision. If R is found unused, the sender transmits an REQ over channel R. REQ includes a set of pulses that identifies the sender's default channel S. If the receiver is idle, it replies with a CON message that is sent by the receiver's LR over channel R after SIFS. It consists of a sequence of pulses with the encoded representation of S. The encoding of S helps to distinguish the node that has won the contention. After transmission of CON, the receiver immediately switches its MR in the active state and tunes it to the channel described in the REQ (i.e., S). After the transmission of the data packet, the sender switches its MR to the channel of the receiver to receive ACK. When a node receiving a data packet on its MR hears a REQ on its LR, it informs the interested sender the time at which it will complete its current transaction. CMAC accomplishes this through WAIT.

There are two possible deaf periods in CMAC. 1) $T_{deaf}(s)$ on the sender side is the time between sending the REQ, switching back to the default channel, and after the reception of CON/WAIT. 2) The other deaf period is at the receiver, as $T_{deaf}(r)$, which begins at the time the first REQ has been heard and extends until information of the header in data frame is read.

However, there are some open questions in CMAC. 1) How does the sender initially know the receiver's channel R, to send REQ message? 2) DCA assumes that each cluster head knows the weights of the cluster heads of all the nodes (that are within 2-hop range of the nodes in its own cluster). But, it does not describe how it can be done. 3) What is the role of the CHs in the network operation after channel assignment phase (DCA assumes that the topology of the network is clustered)?

One of the major shortcomings of CMAC is that it needs to switch twice per packet at MR, one for data packet and the other for ACK. This in turn increases end-to-end delay. CMAC does not require any synchronization, although it requires two transceivers for each sensor node. This increases the hardware complexity and cost of the whole network. Moreover, channel negotiation is needed before each packet transmission, which will incur significant overhead. Meanwhile, the control channel might also become a bottleneck when many motes initialize channel negotiation and request data transmission simultaneously.

Generally, using multiple radio transceivers has some shortcomings. Radio transceivers constantly consume energy, even while asleep, which increases the energy consumption of the nodes. In addition, multiple radio transceivers system needs higher performance communication mechanisms and processor capabilities to receive and process data (or signals) from multiple channels.

4.3. Multi-Channel Lightweight MAC Protocol (MC-LMAC)

MC-LMAC [18] is a schedule-based single-radio multi-channel MAC protocol designed with the objective of maximizing the throughput of WSNs by coordinating transmissions over multiple frequency channels. It is based on the single-channel LMAC [19] protocol. MC-LMAC is based on scheduled access that nodes switch their interfaces between channels dynamically. Time is slotted and each node is assigned the control over a time slot to transmit on a particular channel. In fact, a node selects a time slot and a channel on which it is allowed to transmit.

The state transition of a node in MC-LMAC has been shown in Figure 3. As can be seen, in the *initialization* state, nodes sample the medium for an incoming packet to enter the *synchronization* state. Synchronization is achieved by a hierarchical scheme started by the sink node and continues hop-by-hop as each node synchronizes with its parent node. If a time difference is detected, (e.g. due to possible clock drifts) the nodes roll back to the initialization state. Moreover, guard intervals have also been used for minor drift to ensure that receivers are ready to listen before the senders start transmitting. Time slot *discovery* is an improved version of time slot discovery employed in LMAC, where each node gets a list of free slots and channels from the neighbors. A time slot and channel pair is occupied if the received signal level of the transmissions during the time slot on the channel is above a threshold or if a neighbor node is already receiving from another node during the time slot on a channel. In *time slot and channel selection* state, nodes keep a bit vector for storing the information about the slots occupied by neighbors. A node performs an 'OR' operation over each bit vector per channel and discovers the free slots on different channels. At the last stage, nodes stay in *medium access* state that follows the schedule according to the state diagram shown in Figure 4.

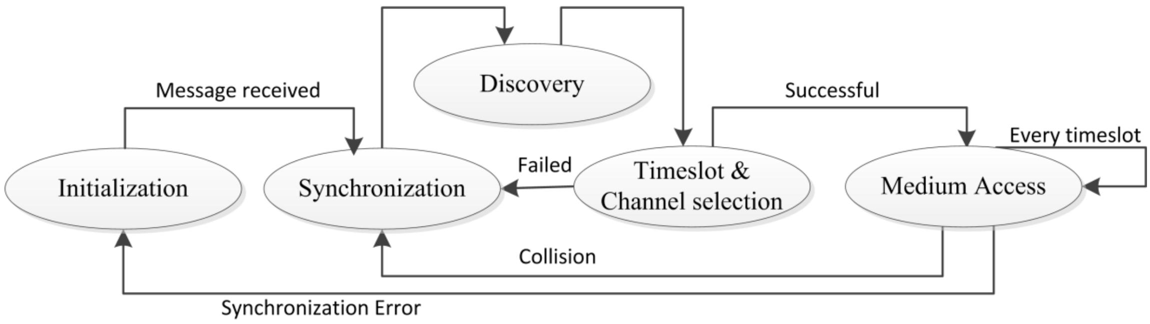

Figure 3. State diagram of a node while executing the MC-LMAC protocol [18]

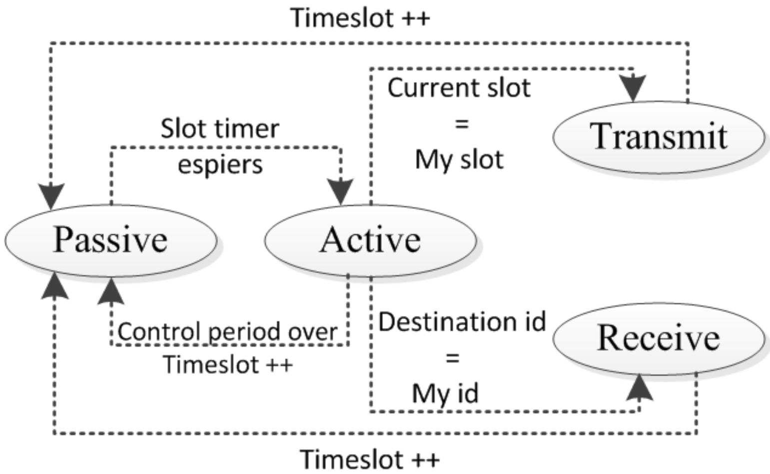

Figure 4. State diagram of a node while following the schedule in MC-LMAC [18]

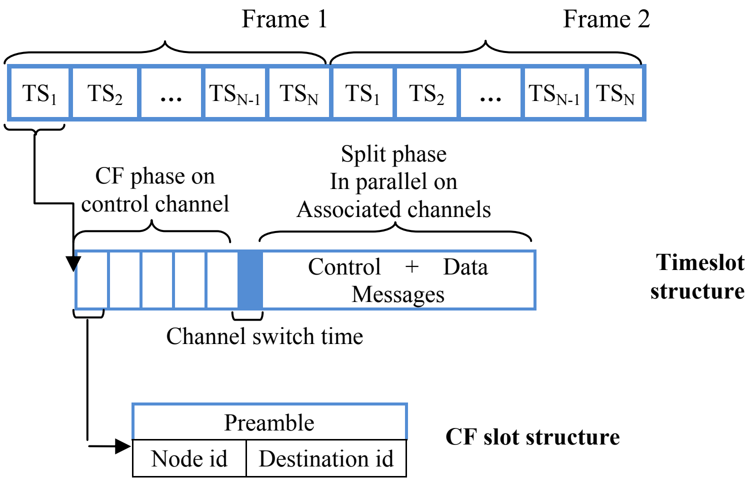

Figure 5. MC-LMAC timeslot structure [18]

The required number of time slots depends on the expected node density of the network. A time slot consists of a Common Frequency (CF) phase and a Split Phase (SP). In the CF phase, all nodes switch to the common control channel to address their destinations and to be informed whether they are addressed in the current slot. In the SP, senders and intended receivers or the nodes that are in the discovery state switch to the channel on which data and the control message transmission will take place. Figure 5 shows the time slot structure of the MC-LMAC. The number of CF slots is equal to the number of channels and it is determined at the initialization of the network and does not change. The main problem of the MC-LMAC is the overhead of the control messages used for synchronization, and channel and time slot selection. It gets worse as network density increases.

4.4. Multi-frequency MAC for WSN (MMSN)

MMSN [20] is the first multi-frequency MAC protocol designed especially for WSNs. It is based on slotted CSMA where at the beginning of each timeslot, nodes need to contend for the medium before they can transmit. The beginning of each timeslot is reserved for broadcasts. When a node intends to transmit a packet it has to listen for the incoming packets on both its own frequency and the destination's frequency.

MMSN allows users to choose one of the four available frequency assignment strategies, such as, 1) *Exclusive Frequency, 2) Implicit-consensus,* 3) *Eavesdropping* and 4) *Even-selection.* The first scheme, *exclusive frequency* assignment guarantees to assign different frequencies to different nodes within any two-hop neighborhood, to reduce communication interference and therefore reduce hidden terminal problem. The communication overhead in this scheme is relatively high due to several broadcasts. The second scheme, *implicit-consensus* provides the mentioned guarantee with smaller overhead but it assumes that physical frequencies are infinite which is not true regarding to current real-world WSNs platforms. The two other schemes, *eavesdropping,* and *even-selection* do not guarantee the assignment of different frequencies to two-hop neighbors and therefore, do not avoid potential conflicts. *Even selection* is a better choice if the sensor network system is more often static and the network congestion is a big issue. *Eavesdropping* can be used for energy saving if the system topology varies a lot with time and the network is lightly loaded. After frequency assignment, nodes cooperate in media access to maximize parallel transmission among neighboring space.

The time slot size depends on the number of nodes that compete for the same frequency and the data packet size. The usual time slot size is 3~5 ms. A time slot in MMSN consists of a broadcast contention period (T_{bc}) and a transmission period (T_{tran}). During the T_{bc}, nodes compete for the same broadcast frequency (f_0) and during the T_{tran}, nodes compete for shared unicast frequencies. The T_{tran} also provides enough time to actually transmit or receive a broadcast or unicast data packet. Consequently, each node initially checks the broadcast frequency for receiving or transmitting a broadcast packet. If there is no broadcast packet to transmit or receive, unicast packet transmission and reception are considered. MMSN uses a snooping mechanism to detect the packets on different frequencies, which cause the nodes to switch between channels frequently. **Error! Reference source not found.**Figure 6 shows the behaviors a node may take, if it has a unicast packet for transmission. The node first listens to the f_0 during T_{bc}. If it senses any broadcast signal, the node spends the rest of the time slot to receive the broadcast packet, as illustrated in Figure 6(a).

If the node does not sense any broadcast signal during the T_{bc}, it takes a random backoff within the time T_{tran} minus $T_{Packet\text{-}Transmission}$ where $T_{Packet\text{-}Transmission}$ is the time it takes to transmit a packet from sender to receiver. During the backoff time, called *toggle snooping* time, node snoop on two frequencies, f_{self} and f_{dest}, where f_{self} is assigned to the node for data reception and f_{dest} is assigned to the intended destination node. Each node snoops on frequency f_{self} to get prepared for a possible incoming unicast packet and also snoops on frequency f_{dest} to be aware if the destination node is busy or not. During backoff time, if the node senses any signal on f_{self}, it receives the data packet, as shown in Figure 6(b). If f_{dest} is sensed busy, this means that another node is trying to send a unicast packet to the same destination. In such a situation, it snoops on f_{self} only to receive a possible unicast packet, as depicted in Figure 6(c). In former state if the remaining time for the current time slot is shorter than $T_{Packet\text{-}Transmission}$, the node turns off carrier sensing to save energy as shown in Figure 6(d). Finally, if there is not any signal in both frequencies during the backoff period, the node sends out a unicast packet as illustrated in Figure 6(e). It sends the packet with the toggle transmission technique, which is illustrated in Figure 7. In this technique the preamble bytes of the physical layer data unit is transmitted with two frequencies, f_{self} and f_{dest}, in an alternating way to reduce collisions.

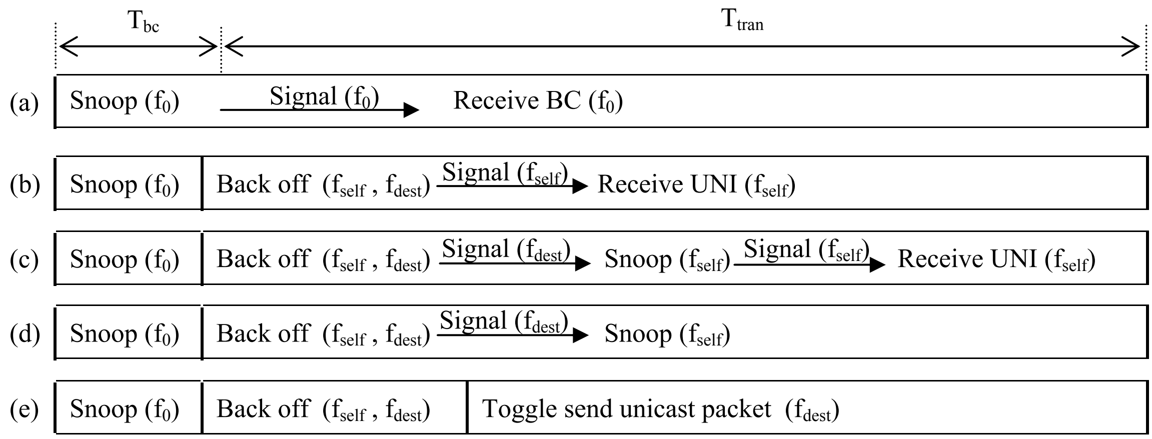

Figure 6. When a node has a unicast packet to transmit in MMSN. (a) receiving a broadcast packet. (b) receiving a data packet. (c) f_{dest} is busy, snooping on f_{self} only to receive the possible unicast packet (d) the node turns off carry sensing to save energy because remaining time for the current time slot is shorter than $T_{Packet\text{-}Transmission}$.(e) sending a unicast packet.(Source: [20]).

However, MMSN has some disadvantages. When a node wants to send a data unit, it has to switch between self-frequency and destination frequency at preamble sending time, which increases the message delay, and protocol overhead. MMSN has a fixed backoff time allocated within each time slot, which it is a shortcoming of this protocol. It also has a lower performance than CSMA even when there are two physical frequencies available. Note that, although MMSN needs time synchronization during media access to provide efficient broadcast support, it does not take advantage of the synchronization service to resolve the conflicts and/or improve its scheme.

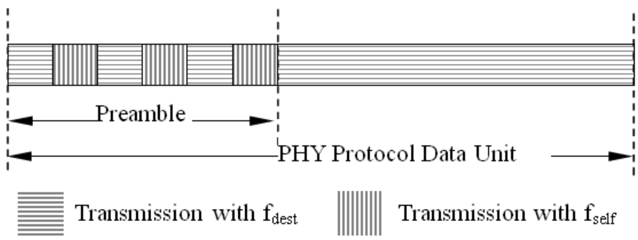

Figure 7. Toggle transmission technique used in MMSN [20]

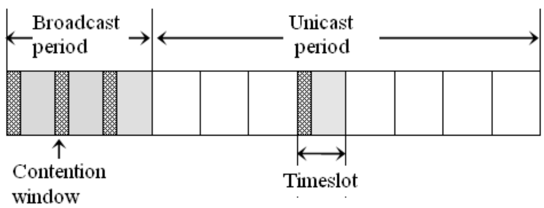

Figure 8. Frame structure of Y-MAC [21].

4.5. Tree-based Multi-Channel Protocol (TMCP)

TMCP [22] is a tree-based multi-channel protocol for data collection applications in WSNs. The main idea of TMCP protocol is to partition the whole network into multiple vertex-disjoint sub-trees all rooted at the base station. Then allocates different channels to each sub-tree and forward each flow only along its corresponding sub-tree. TMCP tries to keep away from complex coordination methods by reducing channel switching and communication among nodes with different channels.

TMCP has three modules, Channel Detection (CD), Channel Assignment (CA), and Data Communication (DC). The CD module discovers existing orthogonal channels, which can be utilized in the current environment. This component uses two nodes to sample the link quality of each channel by sending packets to one another, and then among all channels with good link qualities, it selects non-adjacent channels. Assume the number of non-adjacent channels is represented by k. With these k orthogonal channels, the CA module partitions the whole network into k sub-trees and assigns one unique channel to each of them. Partitioning is the main task of TMCP. The goal of partitioning is to reduce potential interference insofar as possible. After the channel assignment phase, the DC module manages the data collection through each sub-tree. When a node wants to send information to the base station, it just uploads packets along to the sub-tree it belongs. Here, authors assume that the base station is equipped with multiple radio transceivers, each of which works on a different channel to receive data packets simultaneously. In TMCP, the base station can also use the network structure to perform data dissemination. When the base station wants to send commands or update the code, it can transmit packets through all transceivers, and then packets will go through every sub-tree and reach all nodes in networks.

However, TMCP has some shortcomings. TMCP assumes that WSN is static therefore it is not applicable to dynamic WSNs, where new nodes may join the network after initial deployment or nodes die out. TMCP is designed to support data collection traffic and it is difficult to have successful broadcasts due to the partitions. Interference inside the branches of the sub-trees is not resolved because the nodes communicate on the same channel. The bandwidth is not fully utilized because only non-adjacent channels are used. Aggregation cannot be employed since communication among nodes in different sub-trees is blocked.

4.6. Y-MAC Protocol

Y-MAC [21] is an energy-efficient multi-channel MAC protocol for dense WSNs that utilizes a hybrid access method. It assigns time slots to receivers instead of senders. For time slot assignment, every node broadcasts a slot allocation vector, which contains occupied time slots within its own one-hop vicinity as well as itself. Then, each node collects information about occupied time slots in their two-hop vicinity by OR-ing the vectors. To assign time slots, one time slot is randomly selected by node if there are several time slots available in the vector; else, nodes should share the same time slot.

The time slot length is long enough only to receive one message. Figure 8 illustrates the frame structure of Y-MAC. A frame consists of broadcast and unicast periods. All nodes have to wake up at the beginning of the broadcast period to transmit and/or receive broadcast messages. At the beginning of each time slot, potential senders for the same receiver compete to access the medium. To guarantee per node fairness, Y-MAC gives a penalty to the contention winner by limiting the range of back-off timer value for the next transmission. Each receiver wakes up at the end of the contention window, if there is not any coming message; it turns off its radio to save energy. In order to synchronize the starting points for the next frame period, nodes exchange the time remaining in the current frame period. The timing information is included in control messages that every node periodically broadcasts to maintain network connectivity.

In Y-MAC, at light traffic condition, packets are exchanged on a base channel. In heavy traffic conditions, when multiple packets need to be transmitted, sender and receiver hop to a new channel according to a predetermined sequence. Other potential senders also follow the hopping sequence of the receiver. If a node receives a unicast message on the base channel (here F1), it hops to the next channel F2 to receive the following message. Sensor nodes hop to the next radio channel if they have additional pending messages for the receiver. Y-MAC attempts to increase network throughput and decrease message delivery latency through using this mechanism.

In Y-MAC when a node detects a network partition, it goes into the sleep mode to save energy. To re-associate with the network, the radio is periodically turned on to execute bootstrap phase until it receives a control messages. At first, the sleep interval is short. Whenever a node wakes but fails to rejoin the network, the interval doubles.

For evaluation purposes, Y-MAC has been implemented in RETOS operating system on the TmoteSky nodes. This protocol is energy efficient under light traffic conditions. Under heavy traffic, when a node loses the contention on the base channel, it cannot transmit even though there are some channels free and therefore it is one of the shortcomings of the protocol. The other drawback of Y-MAC is that the increased contention especially around the sink node with high data rate scenarios is hard to solve.

4.7. Hybrid MAC Protocol (HyMAC)

HyMAC [23] is a combination of TDMA and FDMA protocols proposed for WSN applications in which data gathered by sensor nodes has to be delivered to at least one sink node in a timely manner. HyMAC is designed to provide high throughput and small bounded end-to-end delay for the packets exchanged between each node and the sink. The communication period in HyMAC is a fixed length TDMA cycle composed of a number of frames. Each frame is equally divided into several fixed time slots where slot duration is the time required to transmit a maximum sized packet. A fixed number of consecutive slots in each cycle starting from its beginning form the *scheduled slots* while the remaining slots of that cycle are its *contention slots*. The base station is responsible to assign an appropriate frequency as well as specific time slot(s) to each node on a tree topology through applying a heuristic utilizing Breadth First Search (BFS) algorithm. *Scheduled node* can communicate in a collision free manner and turn off its radio when it is not necessary to save energy. All scheduled nodes employ the Low Power Listening (LPL) on contention slots during which they randomly select one slot to send a *HELLO* message to the base station. (Rephrase) On the other hand, all of the *unscheduled nodes* like the ones, which have just joined the network, only operate in contention slots to send the *HELLO* message to the base station. If a node receives a

HELLO message from any neighbor node in its one-hop distance, it adds the sender address to its neighbor list. The updated list will be included in the next *HELLO* messages sent by that node. When the base station receives the *HELLO* messages sent by the nodes, it can create the schedule and send it to each node in the network in a *SCHEDULE* message. Therefore, every node will be able to send *DATA* messages to its parent node using its assigned slot and frequency in a way that maximizes the network throughput and minimizes the overall uplink delay.

However, many open questions are not answered in HyMAC such as, 1) how to maintain time synchronized communication, 2) how a new node joins the network, 3) how to resolve collisions, 4) how to implement the protocol in a distributed way, and 5) how the authors simulated and/or emulated the protocol for the performance evaluation.

4.8. Fair Channel Assignment Protocol

A fair channel assignment method to level down the buffer wall in WSNs is proposed in [24]. The buffer wall phenomenon indicates that the nodes nearer to the sink are more likely to have empty buffers due to the lack of data to transfer, while the buffers of the nodes that are further away to the sink are filled with data. Therefore, the major challenge to achieve greater fairness and higher throughput simultaneously is to level down the buffer wall to get an ideal buffer usage distribution.

Main objective of [24] is trying to maximize the minimal achievable data sending rate generated from each individual sensor node. It proposes a multi-channel assignment algorithm called double-plate that is a layer-based approach, which allows the nodes in the same layer of the same sub-routing tree to choose the same sending channel. A sub-routing tree is a routing tree rooted at one node that is one hop away from the sink plus the sink node itself (e.g., there are three sub-routing trees in Figure9). The layer for a node is defined as the hop number from that node to the sink, where sink is layer 0.

In the channel assignment protocol, first, each node needs to find an appropriate parent to form a sub-routing tree. For each sub-routing tree, a series of channels is assigned to it; the assignment makes sure that channels for different sub-routing trees do not interfere with each other. For adjacent layers in the same sub-routing tree, channel interval of no less than 3 MHz is guaranteed. Experiments done in [24] shows that when the two channels are 3 MHz away or more, no obvious interference is observed.

Every node switches channel in an asynchronous slot-based manner. Each node to communicate with the parent chooses a sending channel and to receive should switch to its children's sending channel. Y. Yang et al. [24] propose an intra sub-routing tree channel switching scheme to make sure that nodes with higher buffer usages can get more chances to transmit. It proposes two kinds of decisions for each node, *receiving decision* and *sending decision*, where each of them consists of three transmission times. A receiving decision includes two slots of time being in receiving channel and one slot of time being in sending channel. Similar to that a sending decision consists of two slots of time being in sending channel and one slot of time staying in receiving channel. Each node chooses the next decision in accordance with the buffer usage distribution nearby it. Y. Yang et al. [24] also propose an inter sub-routing tree rate adaptation algorithm to achieve fairness among different sub-routing trees such that nodes in one sub-routing tree can switch to another for better sending rate. As shown in Figure 10, if a node A finds out that its data-sending rate is stable but much lower than that of the potential parent C, A chooses to switch to the sub-routing tree, to which C belongs.

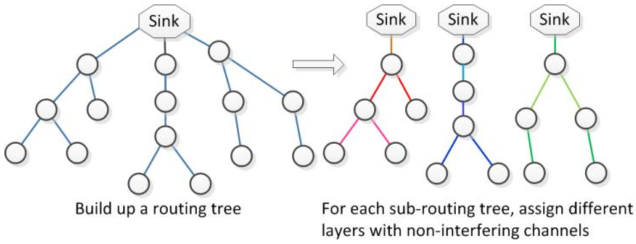

Figure 9. Sub-routing tree building in double-plate [25]

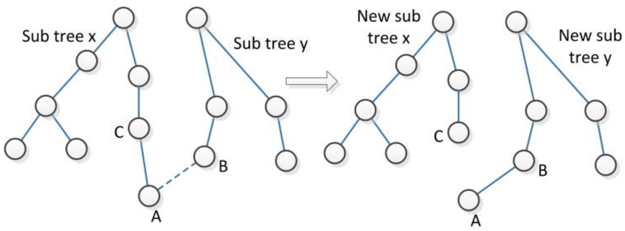

Figure 10. Rate adaptation between two different sub-routing trees [24]

However, in [24] the authors claim that performance of one sub-routing tree is independent to the others, but it is not always true in real world applications. Besides, there is a problem the same as in TMCP, such that aggregation cannot be done, since communication among nodes in different sub-trees is generally blocked. In addition, there is still interference among the nodes in a layer of the sub-routing tree because of the common frequency has shared among them.

4.9. A Practical Multi-Channel MAC Protocol

The MAC protocol proposed in [25] is a general-purpose multi-channel MAC protocol that is designed and implemented on sensor motes with no specific assumptions on the application. The main idea of the protocol is to assign a home frequency to each node such that network throughput is maximized.

In the proposed protocol, channels are organized as a ladder, starting with the lowest channel, F_0, up to the highest channel F_N, where N is the number of channels available in the network. All nodes in the network start at channel F_0. When this channel becomes overloaded, some nodes migrate to higher channels to spread the communication load across non-interfering frequencies; this phenomenon is called *channel expansion*. When a channel is no longer congested, nodes on this channel invite those from the next (higher) channel in the ladder to switch to the underutilized frequency; this phase is named *channel shrinking*. The switching decision is based on how serious are the collisions and interference and also on the role of the node in contributing traffic to the network.

Nodes that communicate frequently are clustered into the same channel, whereas those that do not communicate much (but are within each other's interference range) are separated into different channels. However, there is not any suggestion in the paper for detecting nodes that communicate frequently. Moreover, since nodes in a cluster using the same channel still suffers from the collision.

In this algorithm, nodes that behave predominantly as sinks have preference to switch channels first (i.e., initiate the cluster split). Such nodes do not send much traffic, and hence have a low-cost outgoing link, making it appropriate to split. Nodes that communicate heavily with those sinks come together to create a new cluster. To communicate across clusters, a sender on one home channel switches to the home channel of the receiver to send messages to it.

There are special messages used by this protocol. When a node first joins the network, it broadcasts a *HELLO* message at the home channel to inform its neighborhood that it has joined the channel. Node sends out *WHERE IS* message when the node wants to send a message to a neighbor but does not know its home channel. The message, which is sent periodically by nodes, is *CHANNEL UPDATE* message. When a node wants to leave its current channel, it sends a *BYE* message. Nodes send out *INVITATION* messages to the above channel to invite nodes to join its home channel. The other type of messages is *DATA*, which comprises any messages passed to the MAC by the upper layer.

However, this protocol considers channels as a ladder; therefore, it cannot take advantages of spatial reuse of the radio channels. The other drawback of the protocol is the overhead of *CHANNEL UPDATE* message that are sent out periodically by nodes.

4.10. Cluster based On-demand Multi-channel MAC Protocol (COM-MAC)

COM-MAC [26] is a cluster based on-demand multi-channel MAC protocol for Wireless Multimedia Sensor Networks (WMSNs) [27]. In this protocol, a clustering technique is used in order that each sensor node is associated with one Cluster Head (CH). COM-MAC assumes that each

CH is equipped with a number of half-duplex transceivers that can transmit or receive on some channels simultaneously and each sensor node is equipped with a single half-duplex transceiver. In addition, it supposes that each CH has sufficient power supply and better processing capacity. A scheduled multi-channel medium access technique is applied within each cluster, such that CH is responsible for dynamically allocating time slots and channels for its members according to the QoS requirements and traffic status. To enhance transmission reliability, this protocol uses an ARQ method to utilize the vacant spectrum.

Two protocols, a contention-based and a TDMA/FDMA-based protocol, are designed for each sensor node in support of different application scenarios to send request to CH. Control channel(s) for the contention-based protocol and the time slots for the TDMA/FDMA-based protocol are both assigned by CH(s). The frame structure of COM-MAC is composed of three sessions, request, scheduling, and data transmission session.

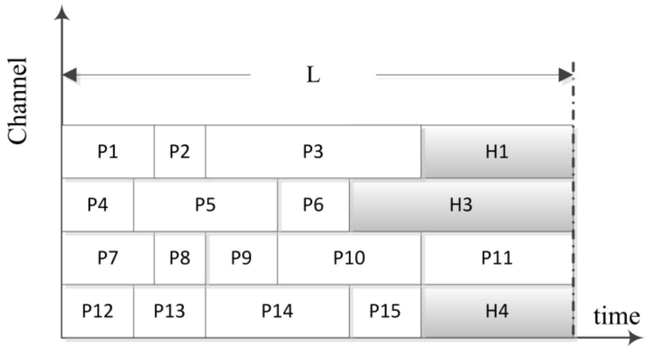

Figure 11. An example of request schedule in COM-MAC [26]

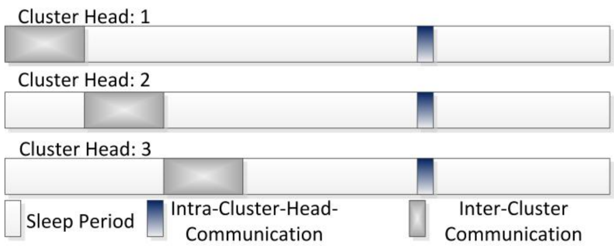

Figure 12. Inter-cluster negotiation in MCMAC [28]

A node will send a request message to its CH at the request session when it has data to send. Based on the information obtained in the request session, CH creates a schedule and broadcast it throughout the scheduling session to coordinate the data transmissions of the cluster members. The created schedule comprises of time slots and radio channels for data transmission as depicted in Figure 11. P_i indicates the request sent by node *i* and H_i indicates the unoccupied space in some channels. To enhance network throughput, H_i is used to retransmit the lost packets. At data transmission session, each sensor node will transmit its data at the assigned schedules and will wait for ACK if it is requested in transmitted packets. Hence, each time slot is divided into data transmission and *ACK* sections.

There are some critical questions unanswered in COM-MAC such as, 1) how does it avoid interference among nodes that belongs to different clusters but using the same channel (inter-cluster interference), 2) what happens to the nodes that want to send data after request session, 3) is the size of the frame fixed or not, and 4) on what parameters does it depend.

4.11. Multi-Channel MAC Protocol (MCMAC)

MCMAC [28] is a coordinator-based multi-channel MAC protocol for WSNs. It assumes that the nodes within the same cluster are synchronized. In addition, a cluster comprises not more than 64 sensor nodes. CHs can communicate with each other at stronger power, where one of the available channels is used as control channel (the control channel can be operated to exchange the control packets and data packets).

MCMAC also assumes that there are many sink nodes in the WSN in order to avoid from the bottleneck problem of single sink node.

In MCMAC, the CH is responsible for synchronization. It broadcasts the synchronous information within the cluster. Sensor nodes work in listening mode at synchronization phase. After receiving the synchronous information, they setup their wake up clock and get into sleep. In MCMAC, every frame consists of an active period and a sleep period. The active period is consists of four phases, (1) *synchronous beacon*, (2) *transmission request*, (3) *channel schedule*, and (4) *data convey*.

The CHs negotiate to reserve a contact-time for inter-CH communication in the sleep period of the frame structure. To avoid inter-cluster interference the neighboring CHs negotiate about sleep times with each other as illustrated in Figure 12. This negotiation is done on the control channel.

CHs send the *synchronous beacon* signal on the control channel at the beginning and end of the active period with the intention that the sensor nodes can adjust their wakeup clocks. The CHs switch to the listening mode at *transmission request* phase, which is divided into several time slots. The number of time slots is equal to the number of the cluster member nodes. The time slots are assigned in accordance with the sensor node ID such that the first slot belongs to the smallest node ID and the last slot belongs to the node ID that is biggest. Then, if a sensor node wants to transmit data, it sends the request packet on the control channel in its chosen time slot.

The CHs collect all the request information from the member nodes at the transmission request phase. At *channel schedule* phase, CHs broadcast the channel assignment information on which the channels for the source node and destination node are determined. The schedule packet includes the source node ID, destination node ID and the chosen channel for data exchange. Member nodes switch their transceiver to the agreed channel according to the content of the channel assignment packets. In the *data convey* phase, if a node is assigned the channel resource it wakes up to transmit on the agreed channel, then returns to sleep mode to save energy.

However, MCMAC has some disadvantages. In order to avoid inter-cluster interference, all neighboring clusters have to go to sleep mode that decreases concurrency and increases latency of the packets in the network. Meanwhile, the main defect of the MCMAC is the high communication burden of the CHs. There are also some open questions in MCMAC such as, how the sink nodes communicate and synchronize with each other. Why the number of nodes in a cluster is limited to 64 nodes since this restriction will decrease the scalability of the protocol.

4.12. Reliable Data Collecting MAC Protocol (Rainbow)

Rainbow [29] is a tree-based MAC protocol designed for reliable data collection in WSNs in scenarios with Radio Frequency (RF) interference. It uses local TDMA [14] and Frequency Hopping Spread Spectrum (FHSS) together. The FHSS scheme is used to decrease collisions, enhance throughput and avoid RF interference. In this scheme every hopping sequence is obtained by cyclic shifting a standard hopping sequence that is generated by interleaving a normal sequence with a *S*-random interleaver [30].

Rainbow uses some messages to construct its tree topology, these messages are *invitation*, *application*, *approve* and *bu*sy tone message that all transmitted at a pilot frequency. In this protocol, each node keeps at most one slot open at the pilot frequency for applying to build tree. If node A wants to join the network, it keeps listening for a super frame time to the pilot frequency to collect *invitation* messages. After that, node A elects the best neighbor node (node B) as the potential parent node according to the routing scheme. When there comes another *invitation* message from node B, node A will send a *busy* tone instantly, and transmit an *application* message to node B after a randomized backoff. Then, node A waits for *approve* message from node B. If there arrives the *approve* message, node A successfully joins the network, and will synchronize to B. It takes node B as parent node, and marks the applied slot as uplink. Moreover, node B marks the open slot as downlink slot, and its frequency starts to hop.

Punch [29] is the name of the distributed time-frequency channel allocation algorithm proposed by Rainbow to obtain collision free communication by allocating exclusive resource in two-hop. Frequency allocation results from Hopping Sequence Start Point (HSSP) allocation. The HSSP of a node is specified by *invitation* message at joining phase. In Rainbow, every node deduces an *index* value from the number of current super frame to index the current frequency in the hopping sequence. This value is the same for the whole nodes. Each node switches its frequency based on its own hopping sequence, the *index* value and of its parent node hopping sequence. The TDMA-based frequency hopping communication is achieved when two nodes turn their radio to the same frequency in the same time slot. When link failure occurs, Rainbow tries to find the best alternative parent node. If a transmission fails, the sending node will retransmit the packet in the next super frame at another frequency.

The main advantage of the Rainbow is its attention to RF interference to handle inter-network interference. However, the major disadvantage is that nodes have to switch its frequency at each super frame, whether interference exists or not, in other words, Rainbow does not use frequencies and time slots optimally. In addition, still an open question exists, how Rainbow handles collision in pilot channel where all control messages transmit on it.

5. Comparison of Multi-channel MAC Protocols

Table 1 shows the challenges that the reviewed multi-channel MACs have addressed. These challenges are covered in Section 3. *Hidden Terminal* column indicates that if the protocol suffer from the hidden terminal problem (shown by +) or not (shown by -). *Deafness* feature shows that whether missing receiver is probable in the protocol. *Synchronization* column points out whether the protocol assumes that the time synchronization is required externally. *Channel switching* column shows the number of frequency switching the protocol needs to do in each step. *Joining Network* as another challenge explains that when the protocol allows a node to join the network. *Intra-network* is a comparison parameter to show the method that is used by each protocol to avoid from internal interference. Finally, *Inter-network* column shows whether the protocol has a method to cope with inter-network interference.

Table 2 compares the different multi-channel MAC protocols discussed in the preceding section. This comparison is from the perspective of different parameters apart from to the challenges they respond. These parameters are *Supported Topology,* *Mobility Support, Implementation* type that can be distributed or centralized, *Channel assignment Method* that explained earlier in subsection 2.3.1, *Data Transfer Channel* refers to channel where sender and receiver have to switch to it for data transmitting, *Number of Transceivers, Evaluation Method, Medium Access* and the main *Objective* of the protocol.

Table 1 . The challenges that the reviewed Multi-channel MAC protocols are addressed.

Protocol Name	Year	Hidden Terminal	Deafness	Synchronization	Partitions	Channel Switching	Broadcast Support	Joining Network	The method used to Cope with Interference:	
									Intra-network	Inter-network
MC-LMAC [18]	2010	-	no info.	Required	-	Once per TS[1]	+	Anytime	Hop count-based (2-hops)	-
MMSN [20]	2010	-	-	Required	-	Multiple per TS	+	At channel assignment	Hop count-based (2-hops)	-
[24]	2010	-	+	Not required	+	Twice per TS	no info.	At channel assignment	Interference range (3MHz)	-
Rainbow [29]	2010	-	-	Required	-	Once per super frame	-	Anytime	Hop count-based (2-hops)	+ (FHSS)
CMAC [17]	2009	-	-	Not required	-	Twice per Packet	-	At channel assignment	Hop count-based (2-hops)	-
TMCP [22]	2008	+	+	Not required	+	None	Inside branches	At channel assignment	Interference disk	+
Y-MAC [21]	2008	-	+	Required	-	Once per TS	+	Anytime	Channel polling	-
[25]	2008	+	+	Not required	-	Between clusters	Inside clusters	Anytime	Channel shrinking + Channel expansion	-
COM-MAC [26]	2008	-	-	Required	+	Twice per frame	-	At clustering	Inter-cluster negotiation	-
HyMAC [23]	2007	-	+	Required	-	Once per TS	no info.	At channel assignment	Hop count-based (2-hops)	-
MCMAC [28]	2006	-	-	Required	-	Twice per frame	Inside clusters	At channel assignment	Inter-cluster negotiation	-

6. Conclusions and Future Research Directions

Several MAC protocols for the WSNs have been proposed by the researchers, although, none of them is accepted as a standard. This is because the MAC protocol in general is application specific. Most of the works on the single-channel MAC in WSNs focuses primarily on the energy efficiency. Nevertheless, multi-channel MAC protocols proposed for WSNs mostly focus on efficient channel assignment and less works have

1. Time Slot

been done on energy efficiency. The effectiveness of multi-channel MAC protocols mainly depends on the channel assignment strategy, which has to guarantee both fairness and low signaling overhead.

Many of recent works on multi-channel MAC in WSNs just concentrate on the theoretical points of the protocol and there is a little effort to design practical one. For instance, it is usually assumed that nodes' radio range is the same as nodes' interference range, but it is not true in the real world. Interference range is about 2 to 4 times more than a node radio range. Therefore, more attention to the real world paradigm makes MAC protocols more accurate and applicable.

Existing multi-channel MAC protocols mainly attempt to avoid from intra-network interference by introducing new channel assignment methods. Inter-network interference is a serious problem in applications that need to work in coexistence with other networks. To propose a realistic multi-channel MAC protocol, it is important to take both of them into account. Still, an open research issue exists, to develop a channel assignment method that results in a minimum interference.

Most of the channel allocation algorithms consider a period for nodes to join or rejoin the network. In addition, some of them even require an independent channel-allocating phase when the WSN cannot function. Hence, it is an essential requirement for designing a MAC protocol so that it can join or rejoin a new node without any interrupt in network mission.

Recently researchers have started to develop sensor networks for different environments such as underwater and underground and so Underwater Acoustic Sensor Networks (UW-ASNs) [31] and Wireless Underground Sensor Networks (WUSNs) [32] have been emerged. These networks are different from traditional terrestrial WSNs since they operate in different environments and communicate through completely diverse media. The diversity in the operating environment and the communication medium have significant effects on the network itself and hence, cause some extra challenges (except the ones inherited from traditional WSNs) for designing a good multi-channel MAC protocol.

Table 2. Comparing Multi-channel MAC Protocols

Protocol Name	Supported Topology	Mobility Support	Implementation	Channel Assignment Method	Data Transfer Channel	Number of Transceivers	Evaluation Method	Medium Access	Objective
MC-LMAC [18]	Tree	no info.	Distributed	Semi-dynamic	Sender	Single	Simulation (GloMoSim)	Schedule-based	Improving throughput
MMSN [20]	Flat	No	Distributed	Semi-dynamic	Receiver	Single	Simulation (GloMoSim)	Hybrid	Increasing parallel transmission
[24]	Tree	No	Centralized	Semi-dynamic	Receiver	Single	Simulation & Implementation (Telosb)	Schedule-based	Improving network throughput and fairness
Rainbow [29]	Tree	No	Distributed	Dynamic	Receiver	Single	Implementation (Tmote sky)	Schedule-based	Reliability in data collecting, evading RF interference
CMAC [17]	Clustered	No	Distributed	Semi-dynamic	Sender	2-Radios	Simulation (based on SimJava)	Contention-based (on LR), Dedicated channel (on MR)	Reduction in energy consumption, Improving throughput and end-to-end delay
TMCP [22]	Tree	No	Centralized	Fixed	Sub-tree	Multiple at sink	Simulation (GloMoSim)	Contention-based	Efficient data collection
Y-MAC [21]	Non-clustered	no info.	Distributed	Dynamic	Receiver	Single	In RETOS OS on TmoteSky	Hybrid	Handling burst traffic
[25]	Clustered	No	Distributed	Dynamic	Receiver	Single	Simulation & Implementation (MicaZ)	No Information	Maximizing throughput
COM-MAC [26]	Clustered	No	Centralized	Dynamic	Cluster Head	Multiple at Cluster Head	Simulation	Schedule-based	Maximizing throughput
HyMAC [23]	Tree	No	Centralized	Semi-dynamic	Sender	Single	Simulation	Schedule-based	Improving throughput and end-to-end delay
MCMAC [28]	Clustered	No	Centralized	Dynamic	Appointed channel	Single	Simulation (OMNET++)	Schedule-based	Energy Efficiency

It seems that design of an efficient enough multi-channel MAC protocol has still a long way to go and many works have to be done in other areas of the MAC layer in WSNs to support network security, reliability, node mobility, and real-time systems. Cross-layer techniques also can be utilized to achieve better communication performance.

References

1. I.F. Akyildiz, W. Su, Y. Sankarasubramaniam, E. Cayirci, Wireless sensor networks: a survey, Computer Networks, 38 (2002) 393-422.
2. I. Demirkol, C. Ersoy, F. Alagoz, MAC protocols for wireless sensor networks: a survey, Communications Magazine, IEEE, 44 (2006) 115-121.
3. single-chip 2.4 Ghz IEEE 802.15.4 compliant and zigbee(tm) ready RF transceiver, http://www.ti.com/lit/gpn/cc2420, in, 2010.
4. J.R. Agre, L.P. Clare, G.J. Pottie, N.P. Romanov, Development platform for self-organizing wireless sensor networks, Proceedings of SPIE - The International Society for Optical Engineering, 3713 (1999) 257-267.
5. E. Callaway, P. Gorday, L. Hester, J.A. Gutierrez, M. Naeve, B. Heile, V. Bahl, Home networking with IEEE 802.15.4: A developing standard for low-rate wireless personal area networks, IEEE Communications Magazine, 40 (2002) 70-77.
6. M.A. Yigitel, O.D. Incel, C. Ersoy, QoS-aware MAC protocols for wireless sensor networks: A survey, Computer Networks, In Press, Uncorrected Proof.
7. J. Chen, M. Díaz, L. Llopis, B. Rubio, J.M. Troya, A survey on quality of service support in wireless sensor and actor networks: Requirements and challenges in the context of critical infrastructure protection, Journal of Network and Computer Applications, In Press, Uncorrected Proof.
8. P. Baronti, P. Pillai, V.W.C. Chook, S. Chessa, A. Gotta, Y.F. Hu, Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards, Computer Communications, 30 (2007) 1655-1695.
9. D. Chen, P.K. Varshney, QoS support in wireless sensor networks: A survey, in: H.R. Arabnia, L.T. Yang, C.H. Yeh (Eds.) Proceedings of the International Conference on Wireless Networks, ICWN'04, Las Vegas, NV, 2004, pp. 227-233.
10. G.P. Halkes, T. Van Dam, K.G. Langendoen, Comparing energy-saving MAC protocols for wireless sensor networks, Mobile Networks and Applications, 10 (2005) 783-791.
11. R. Jurdak, C.V. Lopes, P. Baldi, A survey, classification and comparative analysis of medium access control protocols for ad hoc networks, Communications Surveys & Tutorials, IEEE, 6 (2004) 2-16.
12. N. Piyush, M.S. Krishna, A survey of MAC protocols for sensor networks, in: Wireless sensor networks, Kluwer Academic Publishers, 2004, pp. 93-107.
13. S. Tilak, N.B. Abu-Ghazaleh, W. Heinzelman, A taxonomy of wireless micro-sensor network models, SIGMOBILE Mob. Comput. Commun. Rev., 6 (2002) 28-36.
14. N. Burri, P. Von Rickenbach, R. Wattenhofer, Dozer: Ultra-low power data gathering in sensor networks, in: IPSN 2007: Proceedings of the Sixth International Symposium on Information Processing in Sensor Networks, 2007, pp. 450-459.
15. J. Crichigno, M.Y. Wu, W. Shu, Protocols and architectures for channel assignment in wireless mesh networks, Ad Hoc Networks, 6 (2008) 1051-1077.
16. IEEE 802.15 wpan task group, http://www.ieee802.org/15/1735pub/TG4.html, in, 2010.
17. K.R. Chowdhury, N. Nandiraju, P. Chanda, D.P. Agrawal, Q.A. Zeng, Channel allocation and medium access control for wireless sensor networks, Ad Hoc Networks, 7 (2009) 307-321.
18. O.D. Incel, L. van Hoesel, P. Jansen, P. Havinga, MC-LMAC: A multi-channel MAC protocol for wireless sensor networks, Ad Hoc Networks, 9 (2011) 73-94.
19. L.F.W.v. Hoesel, Sensors on speaking terms: Schedule-based medium access control protocols for wireless sensor networks, in, University of Twente, Enschede, 2007.
20. G. Zhou, Y. Wu, T. Yan, T. He, C. Huang, J.A. Stankovic, T.F. Abdelzaher, A multifrequency MAC specially designed for wireless sensor network applications, Transactions on Embedded Computing Systems, 9 (2010).
21. Y. Kim, H. Shin, H. Cha, Y-MAC: An energy-efficient multi-channel MAC protocol for dense wireless sensor networks, in: Proceedings - 2008 International Conference on Information Processing in Sensor Networks, IPSN 2008, St. Louis, MO, 2008, pp. 53-63.
22. Y. Wu, J.A. Stankovic, T. He, S. Lin, Realistic and efficient multi-channel communications in wireless sensor networks, in: Proceedings - IEEE INFOCOM, Phoenix, AZ, 2008, pp. 1867-1875.

23. M. Salajegheh, H. Soroush, A. Kalis, HYMAC: Hybrid TDMA/FDMA medium access control protocol for wireless sensor networks, in: IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, PIMRC, Athens, 2007, pp. 1-5.
24. Y. Yang, Y. Liu, L.M. Ni, Level the buffer wall: Fair channel assignment in wireless sensor networks, Computer Communications, 33 (2010) 1370-1379.
25. H.K. Le, D. Henriksson, T. Abdelzaher, A practical multi-channel media access control protocol for wireless sensor networks, in: Proceedings - 2008 International Conference on Information Processing in Sensor Networks, IPSN 2008, St. Louis, MO, 2008, pp. 70-81.
26. C. Li, P. Wang, H.H. Chen, M. Guizani, A cluster based on-demand multi-channel MAC protocol for wireless multimedia sensor networks, in: IEEE International Conference on Communications, Beijing, 2008, pp. 2371-2376.
27. I.F. Akyildiz, T. Melodia, K.R. Chowdury, Wireless multimedia sensor networks: A survey, IEEE Wireless Communications, 14 (2007) 32-39.
28. X. Chen, P. Han, Q.S. He, S.L. Tu, Z.L. Chen, A multi-channel MAC protocol for wireless sensor networks, in: Proceedings - Sixth IEEE International Conference on Computer and Information Technology, CIT 2006, Seoul, 2006, pp. 224-230.
29. Y. Yang, W. Yi, Rainbow: Reliable data collecting MAC protocol for wireless sensor networks, in: IEEE Wireless Communications and Networking Conference, WCNC2010, Sydney, NSW, 2010, pp. 1-6.
30. D. Divsalar, F. Pollara, Turbo codes for PCS applications, in: Anon (Ed.) IEEE International Conference on Communications, IEEE, Seattle, WA, USA, 1995, pp. 54-59.
31. I.F. Akyildiz, D. Pompili, T. Melodia, Underwater acoustic sensor networks: research challenges, Ad Hoc Networks, 3 (2005) 257–279.
32. I.F. Akyildiz, E.P. Stuntebeck, Wireless underground sensor networks: research challenges, Ad Hoc Networks, 4 (2006) 669–686.