Ad Hoc Networks 9 (2011) 73–94

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

MC-LMAC: A multi-channel MAC protocol for wireless sensor networks

Ozlem Durmaz Incel [a,*], Lodewijk van Hoesel [b], Pierre Jansen [c], Paul Havinga [c]

[a] NETLAB Research Group, Department of Computer Engineering, Bogazici University, Istanbul 34342, Turkey
[b] Ambient Systems B.V., Colosseum 15d, 7521 PV, Enschede, The Netherlands
[c] Pervasive Systems Research Group, Department of Computer Science, University of Twente, 7500 AE Enschede, The Netherlands

ARTICLE INFO

Article history:
Received 13 October 2009
Received in revised form 10 May 2010
Accepted 14 May 2010
Available online 4 June 2010

Keywords:
Wireless sensor networks
MAC protocols
Multi-channel communication
Schedule based communication

ABSTRACT

In traditional wireless sensor network (WSN) applications, energy efficiency may be considered to be the most important concern whereas utilizing bandwidth and maximizing throughput are of secondary importance. However, recent applications, such as structural health monitoring, require high amounts of data to be collected at a faster rate. We present a multi-channel MAC protocol, MC-LMAC, designed with the objective of maximizing the throughput of WSNs by coordinating transmissions over multiple frequency channels. MC-LMAC takes advantage of interference and contention-free parallel transmissions on different channels. It is based on scheduled access which eases the coordination of nodes, dynamically switching their interfaces between channels and makes the protocol operate effectively with no collisions during peak traffic. Time is slotted and each node is assigned the control over a time slot to transmit on a particular channel. We analyze the performance of MC-LMAC with extensive simulations in Glomosim. MC-LMAC exhibits significant bandwidth utilization and high throughput while ensuring an energy-efficient operation. Moreover, MC-LMAC outperforms the contention-based multi-channel MMSN protocol, a cluster-based channel assignment method, and the single-channel CSMA in terms of data delivery ratio and throughput for high data rate, moderate-size networks of 100 nodes at different densities.

1. Introduction

In typical wireless sensor network (WSN) applications, it is of interest to extend network lifetime due to battery limitations of the sensor devices. As an important source of energy consumption, wireless communication in WSNs has received a lot of attention. Especially several MAC protocols [1] have been extensively studied with the objective of energy efficiency whereas throughput, bandwidth utilization, fairness and latency were considered as secondary objectives [2].

Typically, bandwidth is not a primary concern in traditional low duty cycle, low data rate applications. However, it becomes crucial when sampling at high rate is required, or during certain periods of time when a large burst of packets is generated, for instance, due to a change in monitored conditions. For example, it has been noted that in networked structural health monitoring, more than 500 samples per second are required to efficiently detect damages [3]. Multimedia WSNs [4], which are composed of embedded cameras and microphones besides scalar sensors, also require high throughput and high delivery rate. Moreover, it becomes more common to use sensor nodes that run multiple concurrent applications requiring higher data rates.

The fundamental limitations on the achievable throughput are the limited reuse and/or wastage of bandwidth due to interference and half-duplex operation of the radios. In general, in wireless networks multiple channels[1] have been

* Corresponding author. Tel.: +90 2163597125.
E-mail addresses: ozlem.durmaz@tam.boun.edu.tr (O.D. Incel), lodewijk.vanhoesel@ambient-systems.net (L. van Hoesel), jansen@cs.utwente.nl (P. Jansen), havinga@cs.utwente.nl (P. Havinga).

[1] A channel is defined to be a frequency range over which two nodes communicate. We use the terms "channel" and "frequency" interchangeably in the text.

1570-8705/$ - see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.adhoc.2010.05.003

provisioned to mitigate the effects of interference by scheduling interfering transmissions on different frequencies.

In this paper, we investigate the use of multi-channel MAC protocols to improve the achievable throughput of WSNs. Although the typical WSN radios operate on a limited bandwidth, the operating frequency of the radios can be adjusted over different channels. Once different channels are assigned to previously interfering or contending links, more concurrent transmissions can take place and more data can be delivered to the sink node in shorter intervals.

We first present the challenges and requirements of multi-channel communication from the perspective of WSNs. Next, we introduce Multi-Channel Lightweight Medium Access Control (MC-LMAC), which is a schedule-based multi-channel MAC protocol that takes advantage of contention and collision-free parallel transmissions on different channels. MC-LMAC is designed to provide higher throughput over multiple channels besides meeting the traditional requirements of WSNs, such as energy efficiency and scalability.

The main design is based on the single-channel Lightweight Medium Access Control (LMAC), which has been proven to be an efficient and energy-aware MAC protocol for WSNs [1,5]. A node selects a time slot and a channel on which it is allowed to transmit. time slot and channel selection are fully distributed and guarantee that the same slot/channel pair are not used for conflicting transmissions. A timeslot consists of a control period and a data transmission period. During the control period, all the nodes switch their interfaces to a common channel. The control period is used for notifying the destination about the incoming packet and the channel on which the data transmission will take place such that the receiver switches its interface.

Our contribution is to present a new multi-channel MAC protocol with a fully distributed scheduling mechanism that does not require a centralized scheduler. In contrast to the rich literature on scheduling, especially those using offline graph-coloring approaches, nodes discover and take control of their slots and channels in a localized way by only exchanging information within their local neighborhood in MC-LMAC. The MC-LMAC protocol addresses all the challenges and meet the requirements of multi-channel communication in WSNs. Moreover, we do not assume any unrealistic communication or interference models, such as the graph based models where nodes are assumed to have a communication link if they are within a certain distance from each other, but we do implement the protocol in the GlomoSim simulator using realistic physical layer models and also on real sensor motes. The following are some of the other key highlights of this work:

- We present a review of existing multi-channel MAC protocols for WSNs and discuss the requirements and challenges of multi-channel communication.
- MC-LMAC not only supports many-to-one communication toward the sink node but also broadcasts and local-gossip operations. This can be quite challenging in a multi-channel communication environment with a single transceiver available on each node [6].
- We evaluate the performance of MC-LMAC with extensive simulations in Glomosim, and present a large study of comparisons with MMSN [7], which is a recently proposed multi-channel MAC protocol for WSNs. Different from the scheduled communication in MC-LMAC, MMSN provides contention-based channel access. The protocols with completely different designs allow us to study a large set of trade-offs between different performance metrics. Moreover, we compare the performance of MC-LMAC with single-channel CSMA, and with a clustering mechanism where the branches of the convergecast routing tree are assigned different channels to prevent inter-branch collisions and interference.
- To show the advantages of multi-channel protocols, we compare MC-LMAC and the above-mentioned techniques with an alternative where the communication takes place on a single-channel but over a larger bandwidth.
- We implement MC-LMAC on the Ambient μNode sensor node platform as a proof of concept of time synchronization. In [8], it is argued that frequent channel switching may cause potential packet losses. With this implementation, we aim to show that nodes can change their operating frequency without losing synchronization and packets while running MC-LMAC.

The remainder of the paper is organized as follows: Section 2 presents related works. Section 3 motivates the use of multiple channels in WSNs. Section 4 introduces the MC-LMAC protocol. Section 5 presents the performance of our proposed protocol for typical WSN traffic patterns. Finally, Section 6 draws the conclusions.

2. Related work

2.1. Use of multiple channels in general wireless networks

The problem of channel assignment and multi-channel MAC protocols are well-studied topics for both cellular and wireless ad hoc networks. In cellular networks [9], base stations use different frequency domains within a cell, while clients share the time domain to access the wireless medium. However, this approach is either infrastructure-based or works within a single-hop neighborhood, and so it may not suitable for WSNs where multi-hop topologies are used to cover large areas with short-range radios.

Multi-channel communication has been extensively used in multi-hop ad hoc networks to increase system throughput [10–13]. Most of these approaches are based on IEEE 802.11; for instance, IEEE 802.11b allows 11 channels that are spaced 5 MHz apart. However, the IEEE 802.11 protocols are expensive in terms of energy consumption and may not meet the requirements of WSNs as addressed in [7]. Protocols in [14] either assume multiple radios on the nodes or consider radios that can listen to multiple frequencies simultaneously. Protocols in [15,16] can operate with frequency-hopping spread spectrum wireless cards. However, in WSNs, usually nodes are equipped with much simpler radios and there is only one available on each node.

Considering these differences, multi-channel protocols that are developed for wireless ad hoc networks may not be directly applied to WSNs since the traditional requirements of WSNs, such as energy efficiency and scalability, remain important concerns. On the other hand, the fundamentals of the presented channel assignment strategies can guide protocol designing since WSNs share the challenges of single-radio wireless ad hoc networks, such as broadcast support and avoiding network partitioning.

Single-radio, multi-channel protocols for wireless ad hoc networks can be classified according to the following channel assignment methods: (i) *fixed assignment*, (ii) *semi-dynamic assignment*, and (iii) *dynamic assignment*. In fixed assignment, radios are assigned channels for permanent use. Although the assignment of channels can be renewed, for instance due to changing interference conditions, radios do not change the operating frequency during communication. In the semi-dynamic approach, radios are assigned constant channels, either for receiving or transmitting, but it is possible to change the channel for communicating with radios that are assigned different channels. In dynamic channel assignment method, nodes are not assigned static channels and 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 [13]: (i) *Dedicated Control Channel*, (ii) *Split Phase*, and (iii) *Frequency Hopping*. With the dedicated control channel approach [10,17], nodes synchronize by exchanging control packets on the dedicated control channel and negotiate for the channel to be used for data exchanges. In split phase protocols, nodes access the medium in 2 phases: a control phase and a data exchange 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 exchange phase. Usually, during the control phase, access to the medium is contention based. Protocols differ according to the channel access mechanisms they support during data exchange. An example of contention-based protocols is *Multi-Channel MAC* (MMAC) [12], whereas *Multi-Channel Access Protocol* (MAP) [18] and TMMAC [19] are examples of protocols that are based on scheduled access. In frequency-hopping approaches [16,20], nodes switch, or in other words hop, between different channels.

Following this classification, our contribution is to present a combination of different approaches: semi-dynamic channel assignment that benefits from the "*split phase*". Different than other split phase protocols, access to the medium during the control phase is not contention-based but by following a schedule. This makes the protocol robust against possible collisions during increased contention, and in turn reduces the retransmissions and energy consumption in the network.

2.2. Use of multi-channel communication in WSNs

In WSNs, there are many MAC protocol proposals that consider single channel communication [21–27]. These protocols exhibit good performance in terms of energy efficiency [1], scalability, and adaptability to changes [2].

There are other single-channel MAC protocols that aim to provide high throughput, especially with scheduled communication, such as Z-MAC [25] and Burst-MAC [28]. While these protocols function well in single-channel scenarios, parallel transmissions over multiple channels can further improve the throughput by eliminating contention and interference.

2.2.1. Challenges and requirements

In this section, we discuss the challenges and requirements for single-radio multi-channel communication and address how we use them in our protocol:

- *Synchronization*: 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. Scheduled access, on which our protocol is based, overcomes this complexity.
- *Partitions*: If transceivers of two nearby nodes are fixed on different frequencies, they cannot communicate with each other.[2] MC-LMAC uses a common channel during the control period of each timeslot to let the receivers be informed about the requests and channels on which data will be sent.
- *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. In MC-LMAC, communication on a common channel at the beginning of each timeslot lets the new node collect full information about its neighborhood before starting transmission.
- *Broadcast support*: If the nodes are switching between channels dynamically, it might be problematic to support local broadcasts. However, local broadcasts are important for WSN traffic, for instance, sensor nodes may require in-network processing before they transmit the data toward the sink node or use broadcasts for route discovery. In MC-LMAC, all the receivers of a broadcast are informed on the common channel at the beginning of each slot.
- *Channel switching*: The radio can not switch between the channels immediately but takes some time, for instance, around 200 μs on CC2420 [29] radios. The size of a timeslot in MC-LMAC is large enough to accommodate the switching time; its associated overhead can be considered negligible.

Besides these challenges, the traditional challenges of WSNs, such as energy efficiency and scalability, remain important concerns in designing multi-channel protocols.

[2] This is actually a design decision, because the nodes that do not require to communicate with each other may be placed in different clusters. Here, we refer to the nodes on different channels that require to communicate.

2.2.2. Existing work

There exist recent proposals for multi-channel usage in WSNs. In this section, we discuss the differences between prior work and our work. The performance of existing protocols have been compared with single-channel protocols; here, we compare our protocol with the multi-channel protocols via simulations.

The IEEE 802.15.4 protocol [30,31], which is originally designed for low-rate personal area networks (PAN), 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 and beaconless modes. In the beacon-enabled mode, a coordinator node, which is a full function device, is responsible for adjusting the channel, on which its end-devices, i.e., members, 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 than its parent. Compared to the slotted mode of operation in IEEE 802.15.4, MC-LMAC does not require a central scheduler to allocate timeslots. Due to the hierarchy in the IEEE 802.15.4 networks, the PAN coordinator is responsible for binding of new nodes in the network, scheduling and routing. Additionally, in IEEE 802.15.4, since all the nodes in a PAN, communicate on the same channel, contention within the network is not resolved. In MC-LMAC, nodes that contend with each other are assigned different channels.

In [7], Zhou et al., introduced the MMSN multi-frequency MAC protocol especially designed for WSNs. It is a slotted CSMA protocol where at the beginning of each timeslot nodes need to contend for the medium before they can transmit. On the other hand, in the MC-LMAC protocol we assume scheduled access, where each node is granted a timeslot and performs its transmissions within this timeslot without contention. Contention-based protocols are known to have a lower delay and promising throughput potential at lower traffic loads, which generally happens to be the case in WSNs [2]. However, when the network load is high, there is a higher waste of bandwidth from collisions and back-offs. On the other hand, schedule-based communication has the inherent advantage of collision-free medium access but is less efficient when the network load is low.

MMSN assigns channels to the receivers. When a node intends to transmit a packet it has to listen for the incoming packets both on its own frequency and the destination's frequency. A snooping mechanism is used to detect the packets on different frequencies which causes the nodes to switch between channels frequently. MMSN uses a special broadcast channel for broadcast traffic, and the beginning of each timeslot is reserved for broadcasts. Different from MMSN, MC-LMAC does not require a dedicated broadcast channel. On the other hand, at the start of each timeslot, all nodes are required to listen on a common channel (different from the dedicated broadcast channel, this can be used for data exchanges as well), in order to exchange control information, which simply adds to the protocol overhead. But doing so provides many advantages [5], such as collision-free addressing, maintaining synchronization, allowing distributed operation of the medium access. Moreover, the control period is much smaller compared to the data period, and during the data period the nodes can transmit multiple packets to minimize the overhead.

TMCP [8] is a tree-based multi-channel protocol for data collection applications. The goal is to partition the network into multiple subtrees while minimizing the intra-tree interference. The protocol partitions the network into subtrees and assigns different channels to the nodes residing on different trees. TMCP is designed to support convergecast traffic, and it is difficult to have successful broadcasts due to the partitions. Contention inside the branches is not resolved since the nodes communicate on the same channel.

Similar to TMCP, the protocol in [32] uses a control-theory approach to assign channels to the clusters of nodes. Initially all the nodes communicate on the same channel and when a channel becomes overloaded, nodes migrate to new channels based on the feedback information from their neighbors.

Y-MAC [33] is another recently proposed multi-channel MAC protocol designed for WSNs that is based on scheduled access. However, timeslots are not assigned to the senders but to the receivers. At the beginning of each timeslot, potential senders for the same receiver contend for the medium. Each timeslot is long enough to transmit one data message. If multiple packets need to be transmitted, then the sender and the receiver hop to a new channel according to a predetermined sequence. Other potential senders also follow the hopping sequence of the receiver. As we mentioned, increased contention especially around the sink node with high data rate scenarios is hard to resolve with contention-based protocols.

Another multi-channel MAC protocol proposed for WSNs is HyMAC [34]. Similar to our protocol, HyMAC is also a combination of TDMA and FDMA. However, timeslots and frequencies are assigned according to *Breadth First Search (BFS)* order [35] on a tree topology, and there remain open questions relating to maintaining time-synchronized communication, resolving collisions, new nodes joining the network, and implementing the protocol in a distributed way.

Table 1 shows a classification of the existing MAC protocols and how MC-LMAC differs from these existing works.

3. Motivation

Theoretically speaking, the throughput capacity of a WSN with n nodes under a many-to-one communication pattern can not exceed W/n per node, where W is the transmission capacity of the radio [36]. Practically, this bound is usually not achieved due to the half-duplex nature of the radios and due to the increased amount of contention and interference in dense deployments with multi-hop topologies. In this section, we study a simple benchmark scenario to show the efficiency of multiple channels.

In Fig. 1(a), we present a topology where all the source nodes can directly reach the sink node. Let W represent the capacity of the shared medium. In an idealized setting, aggregate throughput would be W, and each source node should transmit with a capacity of $W/4$. When we switch to a multi-hop scenario, which is shown in Fig. 1(b), if there is no interference then with a suitable scheduling mechanism, we can achieve the $W/4$ throughput per node. However, if all the transmissions interfere with each other, each node can get only $W/6$ capacity (nodes 1 and 2 forward node 3 and 4's packets besides their own packets). On the other hand, if nodes can use different non-interfering channels to transmit, then interference can be eliminated and the nodes can reach the $W/4$ capacity.

In Section 5.1, we present simulation results on the efficiency of multi-channel communication for capacity improvements in WSNs, using the presented topologies in Fig. 1.

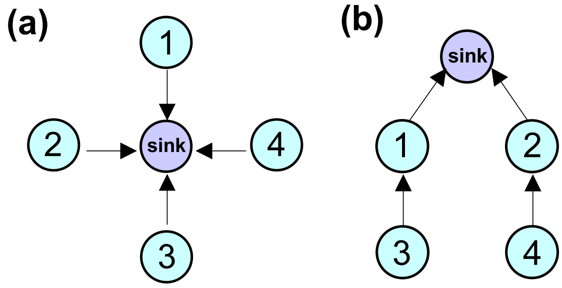


Fig. 1. (a) Single-hop topology used in the benchmark scenario. (b) Multi-hop topology used in the benchmark scenario.

4. MC-LMAC protocol

MC-LMAC is a schedule-based multi-channel MAC protocol. The main design is based on single-channel LMAC [5], which is an energy-efficient medium access protocol designed for WSNs. The LMAC protocol enables the communicating entities to access the wireless medium on a schedule basis in which each node periodically uses a timeslot for transmission. The main aspects of LMAC are:

- *Self-configuration*: LMAC can operate in a fully-distributed ad hoc manner and does not require a centralized scheduler.
- *Adaptability to changes*: LMAC can adapt the communication schedule according to network dynamics, for instance, due to the topology changes, and this is done by local decisions of the nodes without the need of a centralized scheduler.
- *Energy efficiency*: Timeslot scheduling has the natural advantage of collision-free medium access that avoids wasting energy.

Moreover, time-scheduled communication eases the coordination of multi-channel communication. Since nodes switch their interfaces between different channels, a detailed coordination of channel switching is required between the senders and receivers in order to be on the same channel at the same time. Scheduled access overcomes this complexity.

Another key aspect of time-slotted communication is the robustness during high peak loads [37]. Alternative carrier sense protocols may fail to successfully allocate the medium, and thus may result in collisions when the number of sources or data rates increase. Scheduled communication has the advantage of collision-free access. Since we focus on scenarios with a high demand on the medium, we consider LMAC to be a proper choice.

4.1. Protocol organization

As in any scheduled protocol, nodes should first synchronize in order to send and receive messages with correct timing. To access the medium and send messages, nodes select/control a timeslot together with a frequency on which the transmissions do not conflict with the other concurrent transmissions. Equal number of timeslots are grouped into frames.

Nodes transit between five different states while running the protocol: *initialization, synchronization, discovery, timeslot-channel selection*, and *medium access*, as shown in Fig. 2.

Table 1
Comparisons of multi-channel MAC protocols for WSNs.

	MC-LMAC	Y-MAC	MMSN	TMCP	HyMAC	[32]
Broadcast support	+	+	+	No information	No information	No information
Partitions	–	–	–	+	–	–
Medium access	Scheduled	Scheduled	Slotted contention	No information	Scheduled	No information
Channel assignment	Senders	Dynamic	Receivers	Clusters	Senders	Receivers (home channel)
Channel switching	Once per time slot	Once per time slot	Multiple times per time slot	No	Once per time slot	If needed
Joining network	Anytime	Anytime	At channel assignment	At channel assignment	At channel assignment	Anytime (with-scanning)

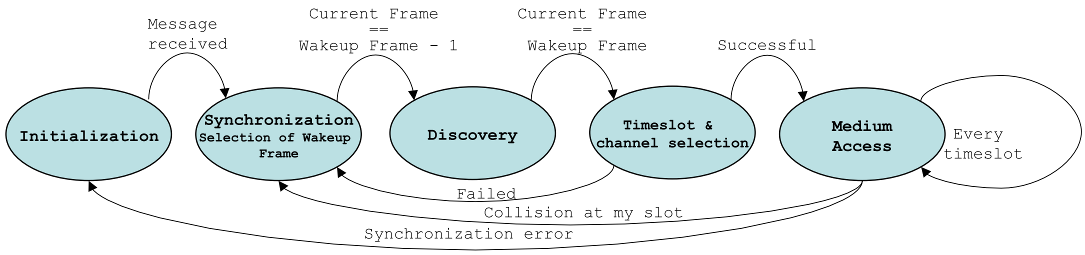


Fig. 2. State diagram of a node while executing the protocol.

Nodes are in the *initialization state* (Fig. 2) if the network is recently deployed or if the nodes rejoin the network, for instance after a reset. In the initialization state, nodes sample the medium for an incoming packet to synchronize with the network. If such a packet is received, a node enters the *synchronization state* and synchronizes with the network by the information about the current slot and frame number in the received packet. After the synchronization, a node follows the schedule to receive packets in the upcoming slots. If the node needs to actively join the network and send packets, it selects a random wake-up frame[3] to select a timeslot within a channel. Before the wake-up frame, a node enters the *discovery state* to get a list of conflict-free slots and channels in the neighborhood, as shown in Fig. 2. In the discovery state, a timeslot and channel pair is recorded as occupied if the received signal level of the transmissions during the timeslot on the channel is above a threshold, or if a neighbor node is already receiving from another node during the timeslot and channel. After a node collects the information for a frame duration, it enters the *timeslot-channel selection state* to select the timeslot and the frequency on which it will perform its transmissions. At the end of the discovery state, if the node succeeds in finding an empty slot and frequency, it transits to the *medium access state* and simply transmits packets in the selected slot on the selected frequency. During the other timeslots, a node samples the medium for potential incoming packets. If a conflict, such as a collision, is reported by an intended destination on a node's selected slot, then the node restarts the slot selection process. If a node experiences synchronization error (details as discussed in Section 4.3), then it transits back to the synchronization state.

When the network is initialized for the first time, the sink node selects a timeslot and a frequency, and starts transmitting. Accordingly, the nodes receiving from the sink synchronize and follow the state transitions. The basic states, shown in Fig. 2, are the same for the single-channel LMAC protocol and the MC-LMAC protocol. However, in MC-LMAC, nodes not only choose a timeslot but also a frequency. The details of the message exchange and medium access are also different.

In the following, we explain the details of the different five states of the protocol. First, we explain the rules for initialization and synchronization. Then we explain the rules for timeslot and frequency discovery and selection, and finally those for medium access control and message exchange together with packet formats used in the protocol.

4.2. Initialization

As shown in Fig. 2, nodes reside in the *initialization state* if the network is recently deployed or if the nodes rejoin the network, for instance after a reset or replacement of the batteries. In the initialization state, nodes sample the medium for an incoming packet to synchronize with the network and enter the synchronization state.

4.3. Synchronization

Synchronization is achieved by a hierarchical scheme such that every node synchronizes with its parent (every node selects a parent node from the set of the nodes that are closer to the sink node in terms of number of hops). Prior to data transmission, the nodes send control messages which include information about the current slot and frame numbers. Upon the reception of a message during initialization, a node records the current slot and frame numbers, which are sent in the control message (as detailed in Section 4.5.1). As mentioned earlier, the timing scheme is started by the sink node at network initialization. When the neighbors of the sink receive the transmission, they synchronize their clocks with the sink's clock. The synchronization continues hop-by-hop as each node synchronizes with its parent node.

Due to possible clock drifts, synchronization errors may occur from time to time. The nodes detect synchronization errors by comparing the received slot and frame numbers in the control message with their local slot and frame numbers. If a difference is detected, nodes transit back to the initialization state. Moreover, guard intervals are used to ensure that receivers are ready to listen before the senders start transmitting to allow small timing differences. In [5], performance of LMAC was evaluated in terms of synchronization. It was shown that, if the nodes synchronize to every frame then a maximum drift of (+, −)2 clock ticks are observed. Therefore a guard interval of 2 clock ticks before and after the expected time of a message reception is used[4] and the nodes synchronize to every frame.

[3] Selection of random wake-up frame aims to reduce the number of collisions, such that the nodes do not select the same timeslot and frequency with a neighbor node, especially at the network setup.

[4] If the crystal oscillators runs at 32.768 kHz, this translates to 61 μs.

4.4. Discovery & time slot and channel selection

In this section, we explain the methods of time slot discovery and selection for the LMAC protocol, and describe the extensions to the case of multiple channels for the MC-LMAC protocol.

4.4.1. Time slot discovery and selection in LMAC

The localized scheduling algorithm of single-channel LMAC is presented in [5]. The algorithm allows a node to periodically get a time interval, called a time slot, during which it is allowed to control the wireless medium and transmit data. The nodes choose a time slot autonomously, such that a node's transmissions in that slot does not conflict with the transmissions of other nodes in the same slot.

If there is no conflict (we explain the causes and resolution of conflicts that may cause packet losses in Section 4.5.3), a node uses the same time slot in the upcoming frames. Each frame has a fixed number of time slots depending on deployment density. Due to the multi-hop nature of WSNs, reuse of time slots or the medium, is possible. All the nodes use one time slot per frame but the algorithm can be extended to allocate more time slots, i.e., allocate more rate, if needed [38].

Time slot selection process takes place either during network initialization or whenever a conflict occurs and a node is required to select a new time slot to eliminate a conflict. If the time slots are selected during network initialization, the sink node starts the selection process by getting the control of a time slot. When a node joins a network, first it has to discover a "free" time slot to transmit its data. A free slot is defined as a slot:

- during which a node is not receiving a packet or detecting a carrier: if a node is receiving a packet during a slot, it would not be able to exchange messages with those nodes transmitting during this slot, and if a node is detecting a carrier, this means the medium is busy and a transmission of the node may potentially cause collisions,
- during which the potential receivers are already receiving from other senders.

To guarantee that the first constraint holds, a node that is searching for a free time slot should exclude all time slots during which a message is received (or a carrier is detected) from the list of potential slots. This means if a transmitting node is within the communication range of a node, it should defer its transmission during that slot. The other constraint should be fulfilled by potential receivers such that they should transmit a list of the time slots during which they are already receiving (or detecting a carrier). This scheme is similar to the RTS/CTS exchange used in CSMA with collision avoidance. The first constraint defers the potential transmissions to prevent possible collisions similar to the RTS transmission to reserve the medium. The second constraint is similar to the CTS transmission such that the receivers notify the potential senders about the message exchange.

The scheme lets the new node determine the list of free slots that can be used without possible collisions. With this information, nodes get a view of the time slot usage within their local neighborhood, and this lets them make a list of potential free slots. A node randomly selects its time slot from the set of free slots (for other methods of time slot selection the reader can refer to [5]). Since the number of time slots is selected according to the expected node density at the deployment phase, all the nodes are given an opportunity to select an empty slot. Together with the time slot selection scheme described above, this guarantees that every node can select a slot to carry out its transmissions without conflicts.

Time slot selection is implemented as follows: All the nodes keep a bit vector called "occupied slot vector" with a length equal to the number of time slots. It is used for storing the information about the slots occupied by neighbors, and is transmitted during the node's time slot to share this information with potential transmitters. Initially, it is filled with 0's, meaning all the slots are free. When a packet is received or a carrier is detected during a time slot, the node inserts a "1" in the vector at the respective position of the vector. This guarantees that a node collects information about the occupied slots that are used by its neighbors. To get a two-hop view of the network, the node should also be aware of the occupied slots in which a neighbor node receives from its neighbors. This guarantees that the node does not transmit on a time slot where a neighbor node is already receiving, thus preventing potential collisions. For that matter, nodes that already selected a time slot share/transmit their list of occupied slots and provide the necessary local information for the nodes searching for a time slot. After a complete frame has passed, the node searching for a time slot can make a list of the free slots by executing an 'OR' operation on all the received occupied slot vectors and the local occupied slot vector.

Fig. 3 shows the timeslot selection. Here, the number of timeslots per frame is 7 and the node "X" is searching for a timeslot. All the other nodes control the timeslot represented by the number inside the circles. Node X receives the occupied slots information (the position of a bit in the vector is the timeslot number: 1 means the timeslot is occupied and 0 means free) from the neighbors, next it executes the OR operation and finds timeslot 7 as free and grabs it.

In order to keep the list of the occupied slots up-to-date, nodes clear their occupied slot vectors after transmissions and collect new information on the usage of time slots until their next transmissions in the upcoming frame. This makes the protocol robust against variations in the wireless links, which may cause the topology and the connectivity of the network change and also helps the nodes keep an up-to-date neighbor list for routing purposes. Moreover, a new node joining the network does not disrupt the already established time slot organization.

4.4.2. Time slot and channel selection in MC-LMAC

The number of required time slots per frame depends on the connectivity of the network topology. If the number of time slots is larger than what is required, the bandwidth may get wasted during empty slots and nodes have to wait longer before they can access the medium. On the other

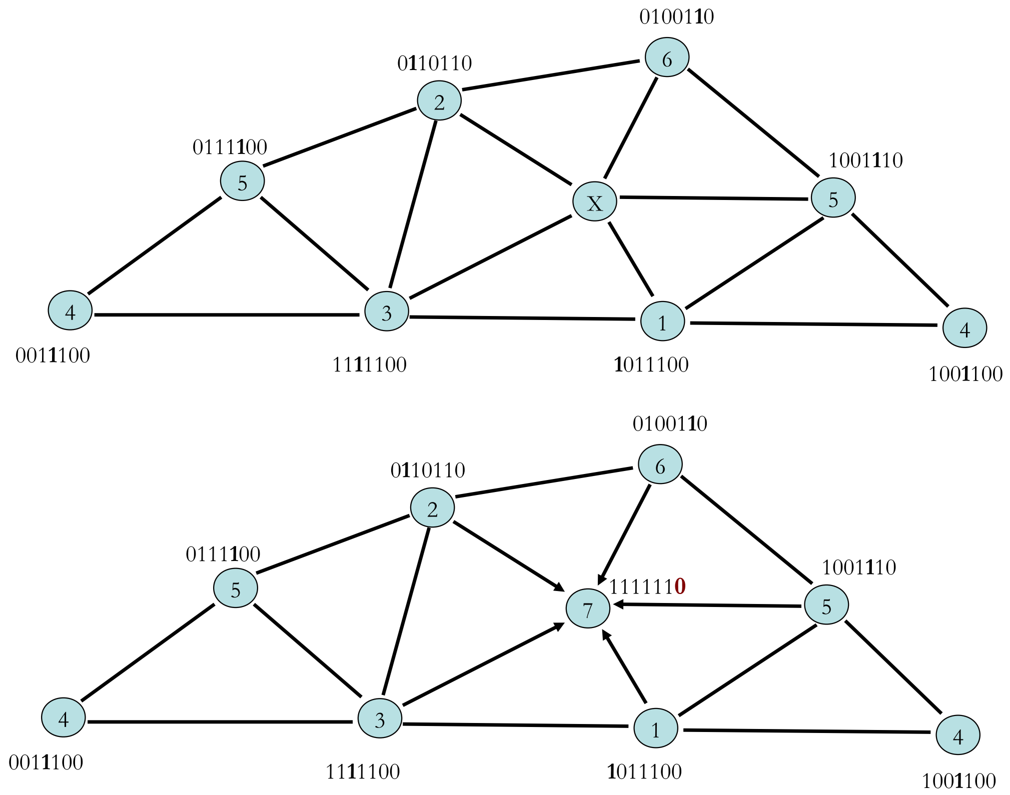


Fig. 3. LMAC timeslot selection.

hand, when there are not enough slots (i.e., the local connectivity is higher than expected), the node remains in the discovery state, periodically monitoring frames for an empty time slot. In single-channel LMAC, the number of transmissions is limited by the number of time slots in a frame. However, in MC-LMAC time slots are selected with channels.[5] A node can use the same time slot that is used by a 2-hop neighbor on a different frequency so that parallel transmissions are not disturbed at common neighbors. Consequently, more transmissions can take place with the same number of time slots.

In MC-LMAC, a node occupies slot vectors per channel and selects a time slot to be used on a particular channel. A node which is trying to get a control of a time slot, executes an "OR" operation over each occupied slot vector per channel and discovers the free slots on different channels. Similar to single-channel LMAC, this method guarantees that the same "time slot/channel" pair is not used within the local neighborhood. Note that, the nodes do not select the time slots used by their neighbors on any frequency due to the limitation of the half-duplex radio.[6]

Fig. 4 shows an example for time slot and channel selection. The node "X" is searching for a time slot while others are marked by time slot/frequency pair that they are using. The number of time slots per frame is 5 and the number of frequencies is 2. The node without a time slot receives the occupied slot information (the position of a bit in the vector is the time slot number: 1 means the time slot is occupied, and 0 means free) from the neighbors, executes the OR operation and finds that all the slots are occupied on F1 (frequency 1), however there is one free slot on F2. The node selects time slot 5. Here note that although slots 1, 2 and 3 are not occupied by any of the neighbors on F2, the node lists those slots as occupied on all frequencies since these are used by the neighbors on frequency F1. According to rules of MC-LMAC, a node does not select a time slot on any of the frequencies which is used by the neighbors.

4.4.3. Required number of timeslots per frame

The required number of time slots in the MAC frame is an important parameter; it affects the possibility of collisions and latency of message delivery. Each node in the network should be given an opportunity to use a time slot without causing or experiencing collisions. The required number of time slots depends on the expected node density of the network. In [5], a detailed analysis is presented using graph-coloring approaches on distance constrained paragraphs. In single-channel LMAC, a node should not use the same time slot which is used within the 2-hop neighborhood, i.e., slots that are sensed to be occupied and slots reported as occupied by the nodes with which a node communicates. On the other hand, in MC-LMAC a node can use the same slot used by 2nd-hop neighbors

[5] We assume the use of only orthogonal channels. If overlapping channels are also used, transmissions on closer channels on the spectrum may interfere with each other which depends on the distance between the transmissions, filtering characteristics and blocking values of a given transceiver.

[6] In our design, the nodes keep a list of the free channels in the neighborhood. For other methods of channel selection mechanisms, such as power sensing or measurement based, the reader can refer to [13].

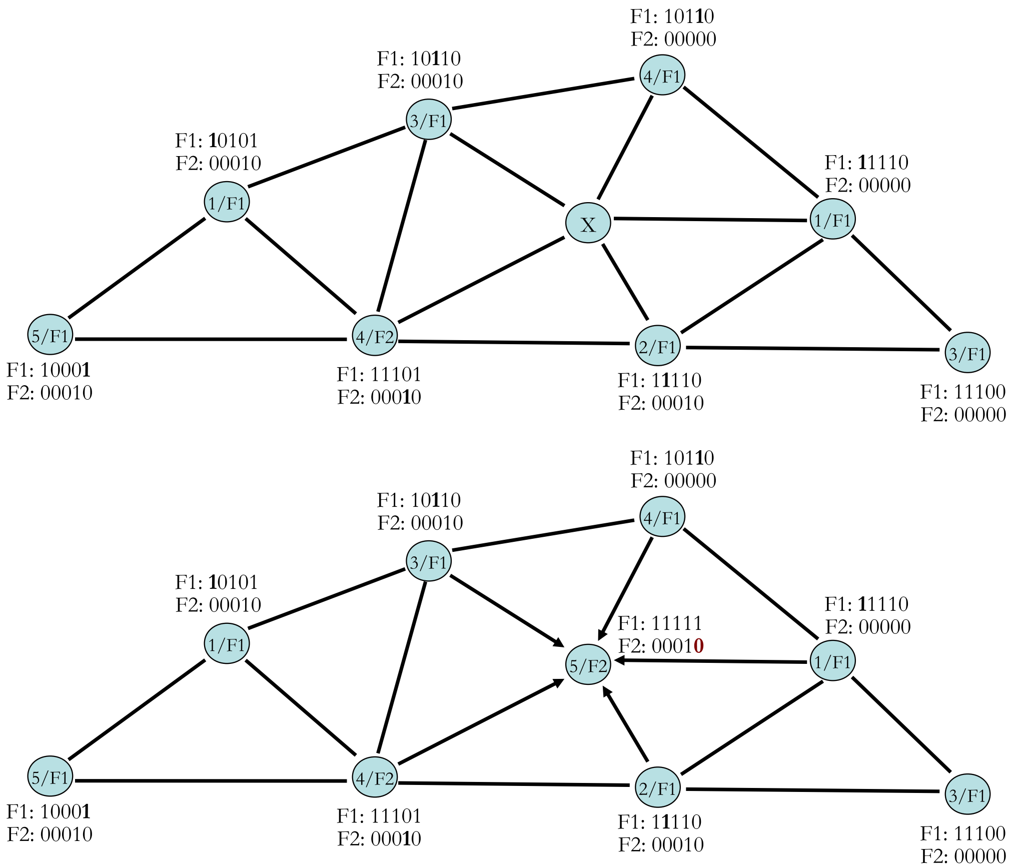


Fig. 4. MC-LMAC timeslot and channel selection.

on a different channel. The number of required time slots is calculated during network initialization according to the density of deployment, and MC-LMAC uses the information to tune its parameter. If new nodes are later deployed in the network, the sink can decide to restart the scheduling and nodes follow the transition between states given in Fig. 2.

4.5. Medium access

After a node synchronizes with the network and selects a time slot and a channel, it stays in the medium access state. In the medium access state, it follows the schedule according to the state diagram shown in Fig. 5 unless a conflict occurs during its transmission (we explain the causes and resolutions of conflicts in Section 4.5.3), or unless a synchronization error occurs. At every time slot, when the timer expires, a node first controls whether the current slot is equal to its own used slot. If so, the node transmits during the current slot. If not, the node listens for incoming potential packets where it is addressed as a destination. If it is addressed by any of the current transmissions, then it receives the incoming packet. If no packet is destined to this node, then the node increases the slot number (if maximum slot number is reached, the slot number is reset and frame number is increased), sets its timer to the beginning of the next slot, and enters into passive state (sleep) to conserve energy. After the reception and transmission of messages, other nodes also update

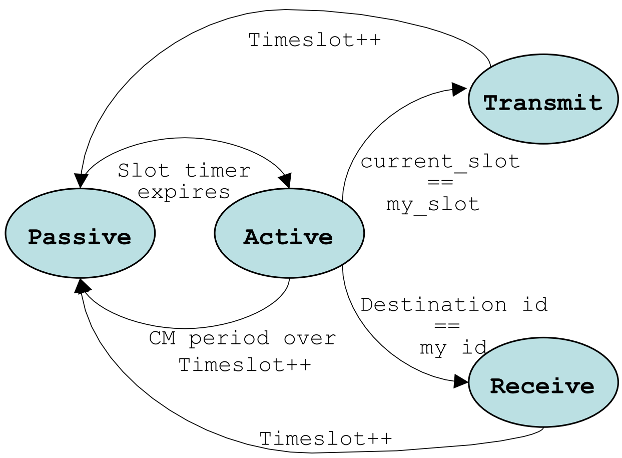


Fig. 5. State diagram of a node while following the schedule.

the slot information and set the timer to the beginning of the next slot.

In the following, we first explain the time slot structure, and then present the packet structures and message exchange procedure.

4.5.1. Time slot structure

A time slot consists of a common frequency (CF) phase and a split phase. 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 cur-

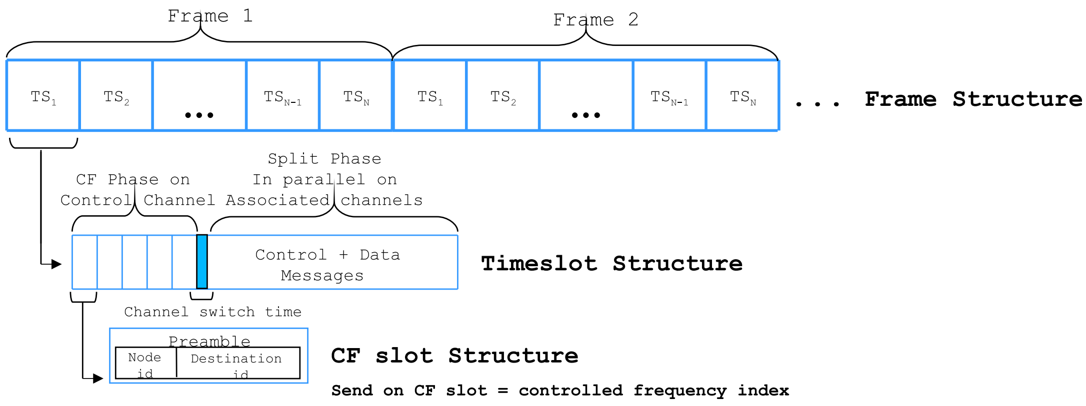


Fig. 6. MC-LMAC timeslot structure.

rent slot. In the split phase, senders and intended receivers or the nodes that are in the discovery state switch to the channel on which the control message and data transmission will take place.

Communication during the CF phase is also based on scheduled access and takes place in small slots called CF slots. The number of CF slots is equal to the number of channels, and each slot is indexed by a channel number. Since the number of channels, on which the radio can be adjusted, is usually fixed, the number of CF slots is determined at the initialization of the network and does not change over time. This means that each CF slot is reserved for the senders of the channel whose number is equal to the CF index (first CF slot is reserved for the senders that selected channel 1, second is for channel 2 and so on). A sender controlling the current time slot addresses the destination during the CF slot which is reserved for the channel number it controls. During the CF phase, the intended destination id (or broadcast address) and node id (sender's address) are transmitted.[7] This enables the sender to notify the destination node and invite it to switch its radio to the sender's channel. The sender id is needed for the destination node to be able to match the sender of the CF message and the sender of the upcoming data messages in the split phase. As mentioned earlier, the sender's channel number is equal to the index, or CF slot number, where it notifies the destination node. Therefore, no extra information is needed to be transmitted. Fig. 6 shows the time slot structure of the protocol.

In the split phase, the sender first sends a control message, which can be considered as a preamble packet, and then continues with the transmission of the data message. The split phase has a fixed maximum length. Depending on the amount of data to be transmitted, the node can send only a single packet or multiple packets. Even if a node does not have data to transmit, it sends a control message during its time slot which provides neighbor discovery and free time slot and channel discovery for the nodes that are in the discovery state. The split phase of MC-LMAC is similar to the time slot structure used in the single-channel LMAC protocol.

The contents of the control message transmitted during the split phase are as follows:

- *ID* represents the node id of the sender.
- *Occupied Slots* represents the bit vector for the occupied slots per channel in the neighborhood.
- *Collision Slot* represents the slot number during which a collision has been detected.
- *Collision Frequency* field is used to distinguish the channel on which a collision has been detected.
- *Current Slot* and *Current Frame* represent the slot and frame numbers, and are used for synchronization by the new joining nodes.
- *Hop Count* field lets the nodes announce their hop distance to the sink node and it is used for synchronization.
- *Acknowledgement Bit Vector* per channel has a length equal to the number of timeslots per frame. Nodes keep track of the slots and channels on which they receive data. Initially the vector is filled with 0's. If a message is received, a logical 1 is inserted at the position of the respective timeslot in the acknowledgement field.

Fig. 7 shows the contents, together with their size, of the control message.

The split phase at each time slot can accommodate the transmission of multiple data packets. Each time slot has a fixed size. Therefore, the size of the split section changes according to the size of the CF section, which in turn also changes by the number of channels. In order to give a number about the sizes, with a frame size of 1.6 s and 32 slots, the number of data messages that can be transmitted per time slot is 18 with 1 channel, and 16 with eight channels, where size of the each data message is around 32 bytes.

4.5.2. Data transmission

The receivers listen during the whole CF phase in order to be informed about the intended destinations. If a receiver is addressed during a CF slot, it switches its transceiver on the sender's associated frequency. If not, the node switches its transceiver to standby mode by entering into passive state for the remainder of the time slot to conserve

[7] It may be argued that, sending only the node id and destination id may not be long enough, and may not be received by the receiver on time if there is a clock drift. In the implementation of the protocol in Glomosim, we checked the duration of sending these fields and found that it is longer than the expected clock drift of 2 clock ticks.

ID	Occupied Slots	Collision slot	Collision Frequency	Current Slot	Current Frame	Hop Count	Acknowledgement
16	32*8	5	3	5	16	4	32*8

Fig. 7. Contents of control message according to 32 slots per frame and eight channels, sizes are in bits.

energy. Note that, the common frequency can also be used by the nodes for data transmission, and it has the same characteristics as the other channels. After the CF phase, the receiver switches to the sender's channel and the time slot owner transmits a control message, followed by a DATA message.

If the transmission is not successful, for instance, due to a packet error or external interference on the medium, the sender should be notified for a retransmission. The receivers acknowledge the correctly received packets by using the acknowledgement field in the control message. Therefore, a sender should listen to the control message that will be sent during the receiver's time slot.

Besides unicast messages, nodes can also send broadcast messages by transmitting a broadcast address during the CF slot. In this case, all the nodes receiving the broadcast request, switch to the sender's frequency.

An example of the overall medium access coordination is shown in Fig. 8. The initial part shows the topology: the numbers inside the circles represent node ids. It is assumed that there are three channels available (represented as *F*1, *F*2, and *F*3), and accordingly there are three CF slots. In time slot 1, sender *S*1 addresses receiver *R*1 in the first CF slot to communicate on channel *C*1, sender *S*2 addresses receiver *R*2 in the second CF slot to communicate on *C*2 and sender *S*3 addresses receiver *R*3 in the third CF slot to communicate on *C*3. The CF phase takes place on the control channel, which is Channel 1 (*C*1). During the split phase, the nodes tune their transceivers on the associated channels: pair 1 on *C*1, pair 2 on *C*2, pair 3 on *C*3. In time slot 2, we observe a similar schedule. Note that, due to interference the three parallel transmissions at each time slot would not be possible if there was only a single channel available.

4.5.3. Conflict resolution

Nodes always use the same time slot in each frame unless a conflict occurs. The causes of conflicts, i.e., packet losses at the receiver, may be due to various factors. Collisions are the major causes of packet losses, when two senders address the same receiver at the same time on the same channel. This can happen during network setup or when network topology changes, for instance due to changes in the routing paths. When a collision has been detected at a time slot, the destination node records the number of the timeslot and the channel and reports these during its own timeslot by marking the collision fields in the control message. If the reported numbers by a node's destination matches with the node's own selected timeslot and channel of the node, the node releases its timeslot and restarts the timeslot selection procedure. To reduce the number of collisions, especially at the start up, nodes wait random times, i.e. random number of frames, to start with the timeslot selection.

Another cause of packet losses may be due to variations in the link quality. In the literature, there are three distinct regions of link quality defined [39]: a nearby connected region where packet reception rates are consistent and high; beyond this lies a transitional region (or a gray area), where reception rates vary much and the disconnected region. The quality of the links in the connected region usually exhibits a packet reception rate above 99% and does

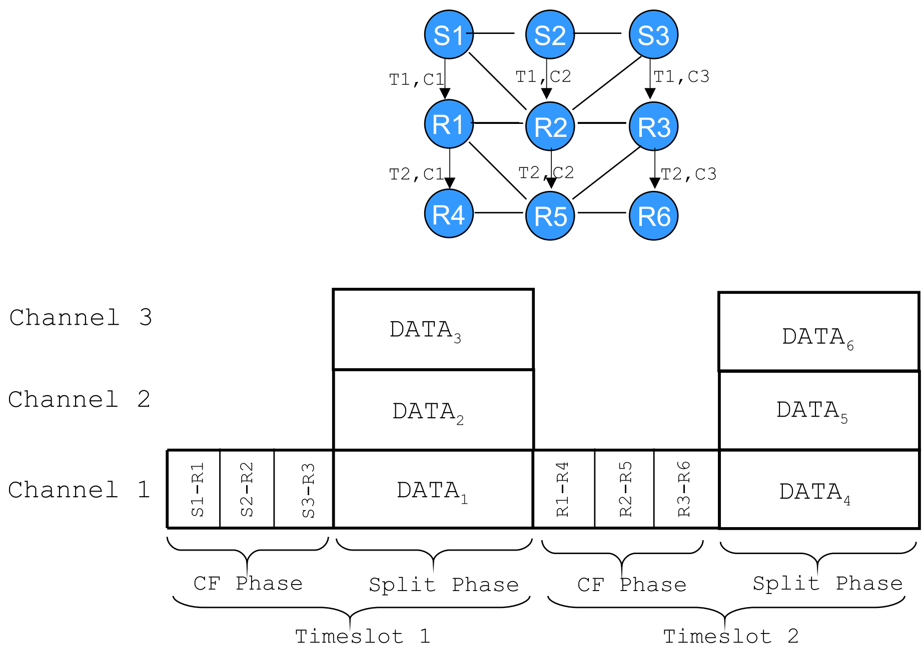


Fig. 8. MC-LMAC coordination scheme.

not exhibit variations. In the transitional region, the links have varying qualities and can be asymmetrical.

In [39], blacklisting of links in the transitional region is given as a solution to deal with the unreliability of the links. A blacklisting mechanism can easily be incorporated in MC-LMAC by using the control messages which carry updated information about the neighborhood. This would help the nodes to select strong links in the connected region for communication. It can be argued that still packet losses may occur on strong links from time to time with a little percentage. In this case, if a packet loss occurs the receiver does not need to report a collision and make the sender to change its slot and channel selection. The receiver can make the distinction by comparing the RSSI (Received Signal Strength value Indicator) value of the received packet with the SINR threshold [5]. If it is just below the threshold the loss is probably due to a temporal reduction in the received signal strength. In the other case, the loss is most probably due to a collision.

Another cause of packet losses may be due to interference from external networks that share the same parts of the spectrum. During the CF phase the nodes update the list of their local occupied slots and channels when they detect a carrier or receive a packet on the control channel. In certain cases, although a node does not detect a carrier from external networks on the control frequency, it is possible that after the channel switching, excessive external interference on the sender's channel may cause a packet loss. In this case, the receiver would report a collision. However, this does not prohibit the sender to select the same channel again, since the receiver will not report the selected slot and channel as occupied. If this case persists, then the receiver concludes that the interference on a particular channel prevents successful communication and blacklists the channel, i.e. report all the slots as occupied on that channel.

In MC-LMAC, if a node is addressed by multiple senders in the same timeslot but on different channels, it receives the transmission requests during the CF phase. However, it can receive from only one of them during the data transmission. We define this situation as a "clash". We discuss the details and solutions for these occurrences in Section 4.6.

4.6. Resolving clashes and discussion of overheads

4.6.1. Clashes

An issue to be solved is a receiver's response if it is addressed by multiple senders in the same timeslot but on different channels, in case of a clash. An option would be to select a sender randomly or select a sender according to a priority mechanism. In our protocol, we use a priority mechanism during timeslot selection by prioritizing the selection of the timeslots that are not used by the other children and the parent of the parent node on the convergecast tree. For instance, in Fig. 9, Nodes *C* and *D* do not select the same slot with node *A* on any channel as long as there are free slots to select from. They can select the same timeslot with Node *E* since *E* is not a parent or a child of node *B*. This efficiently reduces the probability of clashes. For instance, according to normal timeslot selection procedure, the parent, node *A* in Fig. 9, and a child, node *C*, of a node *B* may select the same timeslot which may cause clashes and a problem for node *B* while deciding on which sender it should receive from. In this case, both of the transmissions may be important for the node. By using the new timeslot selection mechanism the occurrence of clashes is reduced. In case a clash still occurs, the acknowledgement field in the control message is used and a retransmission is requested to inform the unselected senders.

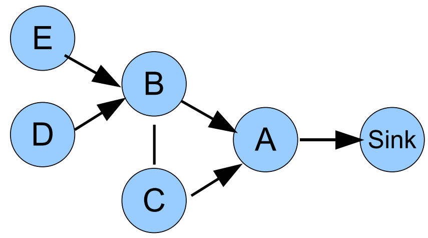


Fig. 9. Prioritization of the selection of timeslots not used by the other children and parent of a parent.

Two problems may arise with clashes and we solve them as explained in the following. If two nodes that control the same slot have a common neighbor, this would not be a problem unless the common neighbor is the addressed receiver, i.e, the parent for a convergecast, of both of the nodes. Normally, if the nodes select their timeslots at different times they would not select the same timeslot since the parent reports the occupied slots in the neighborhood. They would notice that the parent is receiving from another node. In case two senders of a parent node select the same timeslot at the same frame (which is possible although the nodes wait random number of frames before selecting a slot), the parent can only receive from one of them. In this case the parent node uses the collision field to inform the unselected child to change its timeslot selection.

Another problem would be as follows: if a node is in the discovery state and it has two neighbors that cause a clash, such that they control the same timeslot on different frequencies, the node would then be able to receive the occupied slots vector from one of them. As shown in Fig. 10(a), node *S*1 is in the discovery state and receives a message from node *N*1 at timeslot 1 (*T*1) on channel 1 (*C*1). Since it cannot receive a list of occupied slots from node *N*2, it is not aware that *N*2 receives from *S*2 at timeslot *T*2 on channel *C*1. If *N*1 selects *T*2 on *C*1 then a collision may occur at node *N*2 as in Fig. 10(b). In Fig. 10(c), when *N*2 reports this collision during its timeslot then *S*2 gets notified about the collision and starts the new slot selection process. Since *N*2 will report (*T*2, *C*1) as occupied in its local neighborhood,[8] *S*2 will not select the same slot and channel as shown in Fig. 10(d).

In case of clashes where one of the transmissions use a broadcast address, nodes prioritize to receive the broadcast message. Besides convergecasts, we also evaluate the

[8] Node *N*2 may have another neighbor transmitting at *T*2 on another channel but it gets notified about node *S*1 during CF slots and updates its local occupied slot vector. Note that CF slots are clash-free.

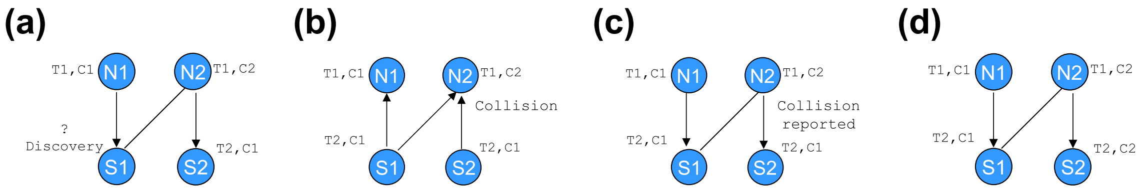


Fig. 10. (a) Clash at a node in discovery state. (b) Node selects the same slot and channel, collision occurs. (c) Collision reported. (d) Selection of a new channel.

impact of clashes on the performance when the nodes use broadcasts, i.e. for local-gossip operations, in Section 5.5.

4.6.2. Protocol overhead

As mentioned earlier, the main overhead of MC-LMAC is introduced by the control messages exchanged before data transmissions. CF phase should be kept as short as possible to minimize the overhead of energy consumption since the nodes have to keep their transceivers on. In single-channel LMAC, nodes have to keep their transceivers on during the control message section. However, in MC-LMAC keeping the interface on during CF phase is sufficient. Moreover, in the implementation of the protocol, the idle-listening overhead is minimized by taking one sample of the carrier in an unused slot to sense any activity (i.e., preamble sampling). If there was activity, the slot is included in the occupied slot vector and listened to completely in the next frame. Moreover, the control period is much smaller compared to the data period and during the data period the nodes can transmit multiple packets to minimize the overhead (similar to message passing in S-MAC [27]: one RTS/CTS sequence is used to reserve the medium to transmit multiple packets). In Section 5 we show that when the network load is high, the CF phase does not add an overhead in terms of energy consumption compared with other protocols, but it enables higher throughput at the sink node by coordinating transmission over different channels.

If the nodes switch between channels very frequently, the channel switching may bring an overhead. However, as we mentioned the timeslot duration in MC-LMAC is large enough to accommodate the switching time. For instance, if there are 32 slots in a 1 s frame and a channel switching is around 200 μs, less than 1% of a timeslot duration is spent on channel switching. Moreover, the nodes do not transmit only one packet per timeslot but are allowed to transmit multiple packets, which in turn compensates the amount of time spent in channel switching. We should also note that, in some of the other multi-channel protocols designed for WSNs, nodes switch channels more frequently. For instance in Y-MAC [33] nodes hop between channels at every message whereas MMSN requires channel switching between the sender's own channel and receiver's channel a couple of times for each message.

5. Performance analysis

In this section, we analyze the performance of the MC-LMAC protocol by extensive simulations with Glomosim [40]. Different simulation scenarios are studied according to four different performance metrics: *aggregate throughput*, *delivery ratio*, *latency* and *energy efficiency*. Aggregate throughput is calculated as the total amount of data delivered to the sink node per unit time by the MAC protocol. We study the performance according to different system loads, different source rates, different numbers of frequencies, different node densities and different traffic patterns.

Simulation parameters are presented in Table 2. Instead of using a simple graph-based interference model, where it is assumed that when the nodes are in the transmission range of each other they cause interference, we use the RADIO_ACCNOISE model, which simulates the behavior of the physical interference model [41] such that interference from multiple senders is captured by comparing the received signal strength to the SINR threshold. According to the radio parameters, the transmission range of the nodes is around 40 m. The sink node is positioned in the center of a square area. In the simulations, we assume that the packet losses occur due to collisions or interference from concurrent packet transmissions in the same timeslot on the same channel. Testing the performance of the protocols with other causes of packet loss is planned as a future work on a real testbed. In most of the simulations, the Geographic Forwarding (GF) [42] routing protocol, which constructs routes according to shortest path or minimum hop count criteria, is used but we also study the performance of a gossip traffic pattern without GF.

Performance of MC-LMAC is compared with the MMSN protocol [7]. Other than the protocols explained in Section 2.2.2, MMSN fulfills most of the requirements/challenges about the multi-channel usage. Its performance has been deeply studied from different aspects [7] and it is a representative of the slotted, contention-based, multi-channel MAC protocols designed for WSNs.

Moreover, we simulate a previously introduced channel assignment algorithm [43] based on LMAC where each branch of the convergecast tree is assigned a different channel; in other words, each branch is clustered into

Table 2
Simulation parameters.

Terrain size	150×150 m^2
Number of nodes	100
Node placement	Random
Number of frequencies	1–10
Bandwidth	250 kbps
Transmit power	1 dBm
Radio model	RADIO_ACCNOISE
Radio range	40 m
MAC protocol	MC-LMAC, MMSN, CSMA
Routing protocol	GF

different channels. Inside the clusters, nodes communicate according to the single-channel LMAC protocol and we refer to this as clustered LMAC. The operation of clustered LMAC is similar to TMCP [8], which was mentioned in Section 2.2.2. In TMCP the level of interference that a node creates on the nodes of a branch is considered. However, in clustered LMAC, nodes join the branches according to the minimum hop count to the sink node or randomly in case of a tie. We also compare the performance of MC-LMAC with clustered LMAC and CSMA (no RTS/CTS mechanism is used). All the results are averaged over 1000 simulation runs. We present the results with 95% confidence intervals.

5.1. Benchmark results

Before presenting the results with 100-node deployments, for which the simulation parameters are shown in Table 2, in this section, we discuss the results of the benchmark scenario. The benchmark was discussed in Section 3 with the illustrated topologies in Fig. 1.

Fig. 11 shows the simulation results. The vertical axis shows the aggregate throughput in total bits per second received at the sink node, and the abbreviations are as follows: SHSF: Single-hop, single-frequency, MHSF:Multi-hop, single-frequency, MHMF:Multi-hop, multi-frequency. In the multi-hop scenario, we assume all the nodes are in the carrier sensing range of each other and nodes in the 2nd level cannot directly communicate with the sink node. Therefore, parallel transmissions on different links, unless they are assigned different channels, may interfere with each other. Nodes transmit 32-byte packets continuously (every 2 ms) to the sink node (effective data rate is 250 kbps). The maximum aggregate throughput, i.e. total amount of data that the sink can receive per unit time from all sources, is approximately calculated as 104 kbps. When the topology is single-hop and there is a single-channel (SHSF), slotted MC-LMAC performs close to the maximum. The only overhead is that of the control messages sent at the beginning of timeslots. Contention-based protocols CSMA and MMSN perform worse. MMSN performs worse than CSMA since some part of the timeslot is spent to listen on the broadcast frequency. In the single-hop scenario,

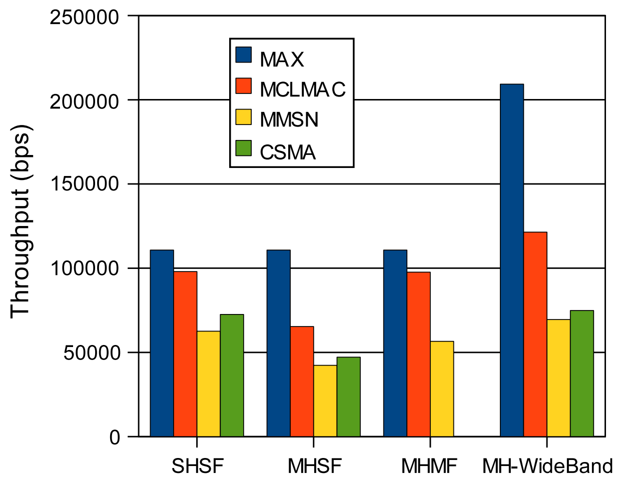


Fig. 11. Benchmark results.

having multiple channels does not improve the results since senders transmit to the same sink node and have to wait for each other's transmission.

When the topology is multi-hop and there is a single-frequency (MHSF), transmissions of all the nodes interfere with each other. All the protocols perform quite poorly. However, MC-LMAC still performs better than the others since collisions are eliminated but it takes six timeslots to deliver all the data compared to the four timeslots in the single-hop scenario due to relaying of the messages. This causes the sink to stay idle (not receiving) during two timeslots which are used by its 2nd-hop neighbors (grandchildren). When there are multiple frequencies available (MHMF), MC-LMAC performs similar to the SHSF scenario achieving a performance very close to the maximum. On the other hand, MMSN performs better than its performance in the MHSF scenario and better than CSMA but cannot achieve the throughput of the SHSF scenario.

If the objective is to maximize the throughput, instead of using multiple channels, using a more powerful radio with a higher data rate could work better than the multi-channel scenario. In the last column of Fig. 11, we present the results where the nodes can transmit over a double size band, i.e. with an effective data rate of 500 kbps. Compared with the results of MHSF and MHMF, all protocols achieve higher throughput. However, most of the band is still not utilized due to the interference experienced on the same channel. Moreover, contention-based protocols, CSMA and MMSN, over a wider band perform worse than the MC-LMAC protocol with two channels presented in the MHMF scenario. Additionally, using a radio that can transmit over a wider band may consume more energy which is not desired by WSNs due to the energy constraint on the sensor nodes.

Observations from the benchmark results are two-fold: Due to the common destination problem with the many-to-one traffic pattern, aggregate throughput is limited by the reception capacity of the sink node. However, this throughput is usually not achieved in multi-hop scenarios due to contention, interference and collisions that increase with relaying of data. Multi-channel communication can cope with interference and collisions and improve the throughput and delivery performance. Next, we conclude that schedule-based medium access can indeed better cope with high peak loads [37] since the contention and collisions are eliminated.

5.2. Impact of the number of channels

In this section we analyze the impact of the number of channels on the network performance. All the nodes initiate CBR (continuous bit rate) streams towards the sink node and each node generates a packet every 2 s (if nodes transmit more frequently, buffer overflows start to occur). This scenario can be considered as a periodic data collection, which is commonly used in WSN applications. Although methods such as data aggregation can be used to reduce the amounts of data to be delivered to the sink node, in these simulations all the raw data from sources is relayed towards the sink node (we previously discussed increasing the data collection rate of aggregated converge-

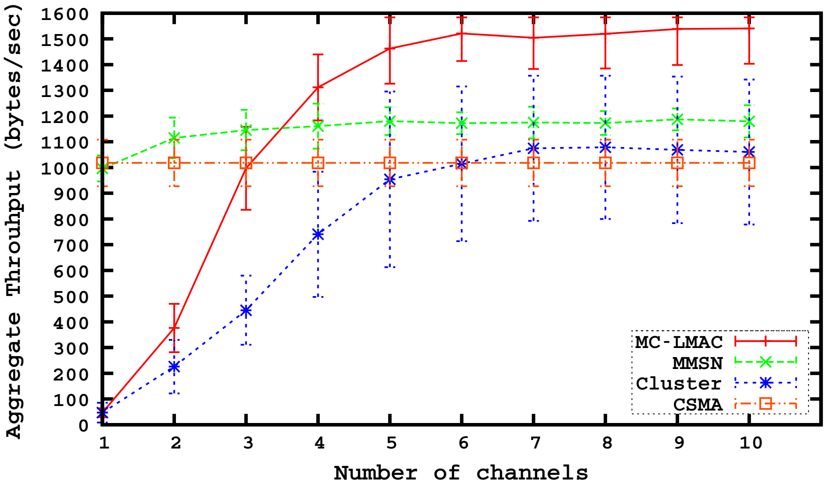


Fig. 12. Aggregate throughput with different number of channels.

cast operations in [44]). The number of channels varies between 1 and 10. The terrain size is 150 × 150 m^2.

5.2.1. Aggregate throughput

Fig. 12 presents the results in terms of aggregate throughput. The x-axis shows the number of available channels; the y-axis shows the aggregate throughput in terms of the number of bytes per second received by the sink node. Maximum aggregate throughput at the sink node is 1584 bytes/s (99 sources generate 32 byte packets every 2 s). Different lines present the results collected with different protocols: MC-LMAC, MMSN, Clustered LMAC and CSMA. In MC-LMAC, the number of timeslots per frame is 32.[9] The maximum hop count for different random deployments is usually two or three hops according to the density.

Aggregate throughput increases when the number of channels increases from 1 to 10 with all the protocols except CSMA where the number of channels is fixed to 1. Example radios used on sensor nodes may actually provide more channels. For instance CC2420 [29], can tune its operating frequency over 16 different channels in the 2.4 GHz band and Nordic NRf905 [45] radio can operate over 512 channels in 868/915 MHz band. However, radio signals are usually not limited to their allocated frequency band. Therefore, channel overlaps, that may cause interference, may be examined between adjacent bands depending on the filtering characteristics and the blocking values of the transceiver. Accordingly, the number of orthogonal channels can be limited. For instance, in [46], we experimented the impact of adjacent channel interference using Nordic Nrf905 radio and concluded that 10 channels can be used as orthogonal from the 512 channels, considering the worst case. The same holds for the CC2420 radio [8]. Usually all of the 16 channels cannot be used but 6-10 channels are reasonable numbers that can be used in parallel.

MC-LMAC achieves lower throughput than MMSN with 1–3 frequencies since some of the nodes cannot get a free timeslot, i.e, 32 slots per frame is not sufficient for all the nodes to get a timeslot according to the deployment density, and cannot start transmissions. Especially, when there is a single-channel nodes cannot select a timeslot that is used in the 2-hop neighborhood and the size of the 2-hop neighborhood is larger than 32. With the given density the number of timeslots per frame should be higher than 32 slots when the number of channels is small. A solution could be to increase the number of timeslots. However, increasing the number of timeslots would increase the throughput but cannot achieve the maximum throughput since the nodes cannot be active during the slots used by 2-hop neighbors, as we discussed in Section 5.1.

As the number of channels increases, more nodes can get the control of a timeslot. After six channels, MC-LMAC achieves the maximum throughput in most of the simulations. On the average, the achievable throughput is 99% of the maximum throughput. Loss is due to the clashes that may occur. As we described earlier, in the implementation of the protocol we reduced the probability of the clashes by prioritizing the selection of the timeslots that are not used by the other children of the parent node on the convergecast tree. Compared to CSMA the MC-LMAC achieves approximately 1.4 times better performance and compared to MMSN the throughput performance is 1.3 times better but most importantly the maximum throughput can be achieved using MC-LMAC.

Aggregate throughput with MMSN is observed to be limited and does not increase much after three channels. This is due to the failure of the nodes around the sink to successfully sense the channel and prevent the collisions. The nodes should switch between the destination's frequency and their own frequency to sense incoming packets. However, while switching, nodes may miss some of the incoming packets. Additionally, sensing the destined packets to the receiver may not always be successful since no RTS/CTS mechanism is used.

[9] The number of timeslots is adapted according to the expected node density. According to the example deployment, average number of first hop neighbors per node is around 24 nodes. Moreover, in order to have efficient forwarding during convergecasts, the nodes should not select the same timeslot as the other children and the parent of the parent node to prevent clashes. Accordingly, 32 slots per frame is an experimented, suitable value for the given density in the example deployment.

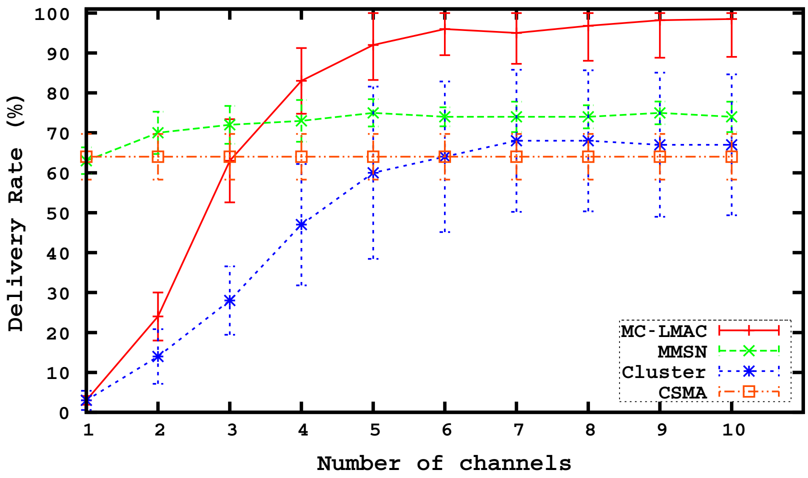


Fig. 13. Packet delivery rate with different number of channels.

Achievable throughput with clustered LMAC is rather limited due to the single-channel communication inside the clusters. Nodes need to select timeslots that are not used in the 2-hop neighborhood to prevent collisions and contention and this reduces the number of concurrent transmissions. The performance of the clustered LMAC protocol can be improved by balancing the number of nodes on the branches. However, due to the random deployment of the nodes and connectivity constraints it may not be always possible and the interference inside the clusters may still limit the performance.

On the average, single-channel CSMA achieves an aggregate throughput of 64% of the maximum throughput. Due to the high contention, the protocol fails to successfully allocate the medium to the nodes. Compared to CSMA, MMSN achieves slightly lower throughput on a single channel which is due to the time spent on sampling the broadcast channel at the beginning of each slot.

5.2.2. Delivery ratio

Fig. 13 presents the results in terms of delivery ratio. The *x*-axis shows the number of available channels; the *y*-axis shows the delivery ratios. The figure has a very similar shape with the aggregate throughput graph presented in Fig. 12. The reason for this is that, we considered only best effort delivery, such that the lost packets, i.e., the unacknowledged packets, are not retransmitted. Although MC-LMAC provides reliability by using acknowledgement fields in the control message, MMSN and CSMA does not have the acknowledgement mechanism which is left to the upper layers. For fair comparisons, we did not activate the retransmission mechanism in MC-LMAC.

With sufficient channels, MC-LMAC achieves to deliver 99% of the packets on average. As we mentioned, the small percentage of losses is due to clashes. However, with a smaller number of channels, the delivery ratio is more limited since most of the nodes cannot get a free timeslot.

On the other hand, contention-based MMSN protocol saturates around 70% delivery ratio with an increasing number of channels and CSMA delivers only 64% of the packets. As we mentioned, during peak traffic, contention-based protocols cannot handle the high contention on the medium whereas the slotted MC-LMAC guarantees the medium access by assigning different timeslots to the possible contending nodes.

5.2.3. Latency

Fig. 14 shows the results in terms of end-to-end latency between the transmission of a packet at the source node and reception at the sink node. When we compute the latency values, we do take into account the time spent for exchanging control messages in MC-LMAC. Similarly, we also include the time for listening on different channels in MMSN prior to data transmission.

Although MC-LMAC achieves lower latency than clustered LMAC and CSMA, MMSN has much lower delay compared to the MC-LMAC protocol. Higher latency is a typical characteristic of the schedule-based protocols. If a node has a packet to transmit it has to wait till its assigned slot. The average delay from source to the sink is equal to a frame size which is approximately 1.6 s.[10]

A simple solution to decrease the latency would then be to decrease the frame size. However, in that case the number of packets that can be delivered per timeslot will also decrease and the packets will be buffered to be transmitted later. The best option then is to assign the relaying nodes consecutive timeslots according to their hop distance to the sink node. Another cause of latency in MC-LMAC is the fixed size of section in a timeslot that is reserved for the data messages. If a node has less packets than what can be transmitted during a timeslot, the idle time simply adds to the latency. Variable data sections per timeslot could be another solution to improve the latency in MC-LMAC. The question may arise why MC-LMAC performs worse in terms of latency although it performs very good in terms of aggregate throughput. Aggregate throughput is the amount of data delivered to the sink node divided by the duration the sink has completed receiving packets from the network. However, latency is computed per

[10] The selection of the timeslots that are before the parent node's slot are prioritized. Delay per hop count is on the average half of the frame size. Considering an average number of 2 hops between a source and the sink node, this makes around 1 frame duration.

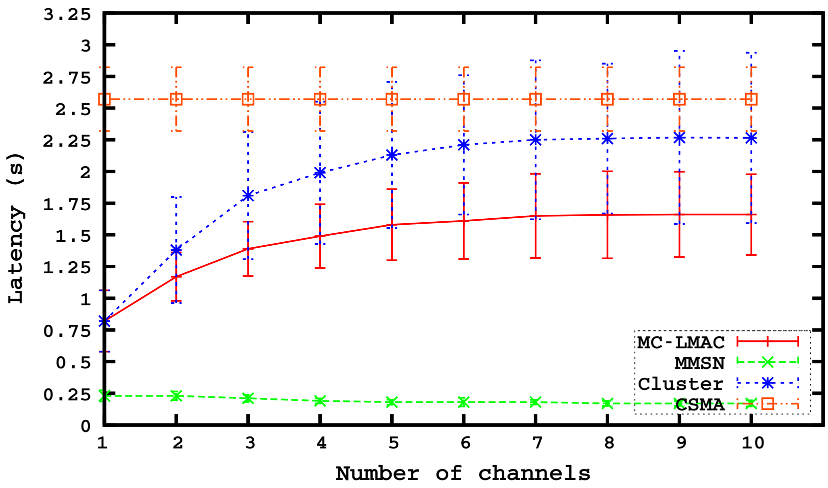


Fig. 14. Latency with different number of channels.

packet, which is the duration between the transmission at the source node and reception at the sink node.

CSMA also experiences higher delay than MMSN which is due to the exponential and higher number of back-offs during high contention.

As we mentioned, we did not consider the retransmission of the lost packets, due to collisions. MMSN achieves to deliver only 70% of the the transmitted packets. This means much less packets are scheduled compared to the MC-LMAC protocol.

5.2.4. Energy efficiency

Fig. 15 shows the results in terms of energy efficiency, i.e. energy consumed per successfully delivered packet. This translates to total energy consumed that is divided by the total number of successfully delivered packets. We consider both the energy spent to receive and transmit as well as the energy spent for relaying the packet towards the sink node. Additionally, energy spent on sending control messages are also included. Energy spent per delivered packet is quite high with MC-LMAC when there is only a single-channel. This is due to the very low delivery rate. As the number of channels increases, both MC-LMAC and MMSN spend much less energy than CSMA. It can be argued that in MC-LMAC, the duration of the CF period increases with more channels and this causes the nodes to spend more energy on listening for the potential incoming packets. The same holds for CSMA where the nodes should always be listening for incoming packets and with MMSN nodes always listen for a period on the broadcast channel and then listen on the destination's frequency and its own frequency in alternating cycles for the incoming packets. On the other hand, clustered LMAC with a single channel spends less time on idle-listening but the number of attempts per transmission is much lower since some of the nodes cannot control a timeslot due to the interference and contention within the branch. That is why we preferred to show energy efficiency per successfully delivered packets instead of the total energy consumption for fair comparisons.

5.3. Impact of load

In this section, we analyze the impact of the load on the network performance. In particular, we vary the number of sources, which in turn changes the level of contention in

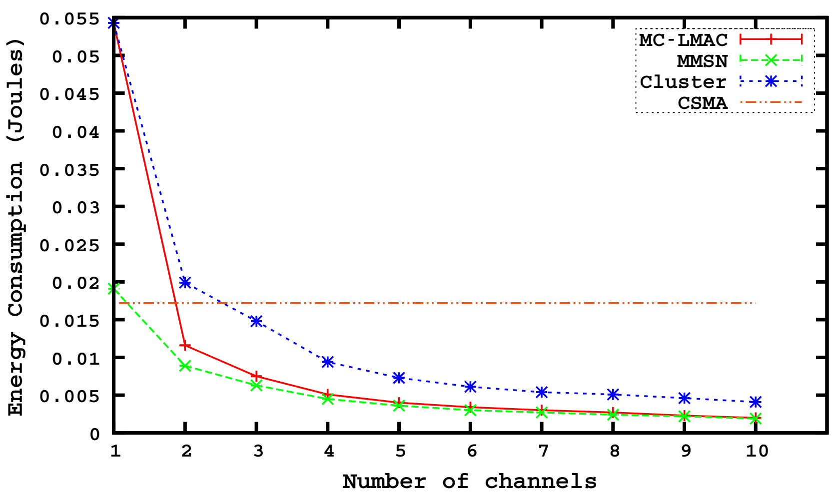


Fig. 15. Energy efficiency, energy consumption per successfully delivered packet, with different number of channels.

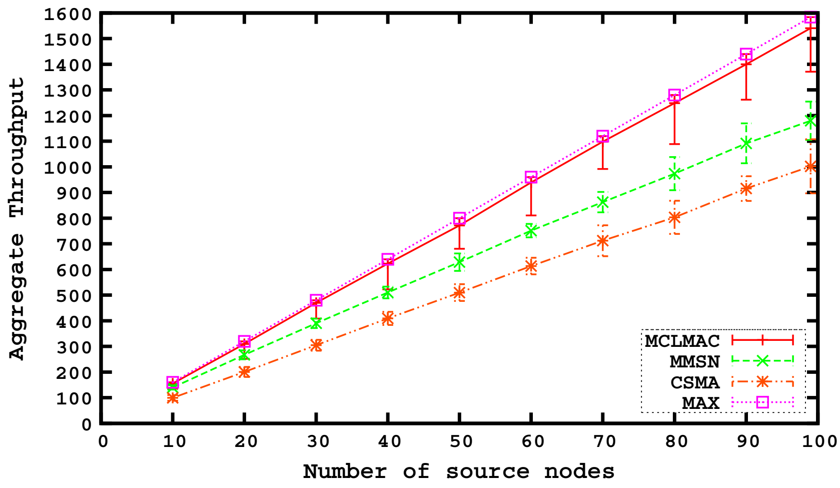


Fig. 16. Aggregate throughput with different number of sources.

the network. Fig. 16 shows the results in terms of the active sources. We vary the number of sources from 10 to 99. The number of channels for both MC-LMAC and MMSN is 8.

Since MC-LMAC assigns slots to all of the nodes, whether they are the sources or not, the performance of MC-LMAC is close to the maximum aggregate throughput in all cases. However, MMSN and CSMA suffer from contention. When more sources are active, i.e. there is more load to be forwarded to the sink, the medium access mechanism of MMSN cannot sense the incoming packets at the destination's frequency, particularly around the sink node. Therefore packet losses increase. This again verifies the conclusion that schedule-based medium access can better cope with increasing contention/load in the network. Additionally, schedule-based medium access combined with multi-channel communication enables more concurrent transmissions and achieves a throughput close to the maximum.

In this set of simulations, the nodes generate packets every 2 s. We also investigated scenarios where nodes generate packets more frequently. In that case, all the protocols experience buffer overflows with higher data rates and the achievable throughput gets much lower than the maximum.

5.4. Impact of density

In this section, in order to test the scalability of the protocols we vary the density of the node deployment. We change the terrain size between $50 \times 50\,m^2$ and $225 \times 225\,m^2$ (beyond 225 m, unconnected nodes appear with random deployment). Fig. 17 presents the results. The *x*-axis shows *L*, the side length of the deployment area whereas *y*-axis shows the aggregate throughput. The number of channels is 8 for both MMSN and MC-LMAC.

Aggregate throughput with MC-LMAC is lower when $L \neq 150$ since 32 slots per frame is lower in denser scenarios and higher in sparser scenarios than required. In sparser scenarios the sink stays idle during unused timeslots. We repeat the experiments with different numbers of timeslots that are adjusted according to the expected den-

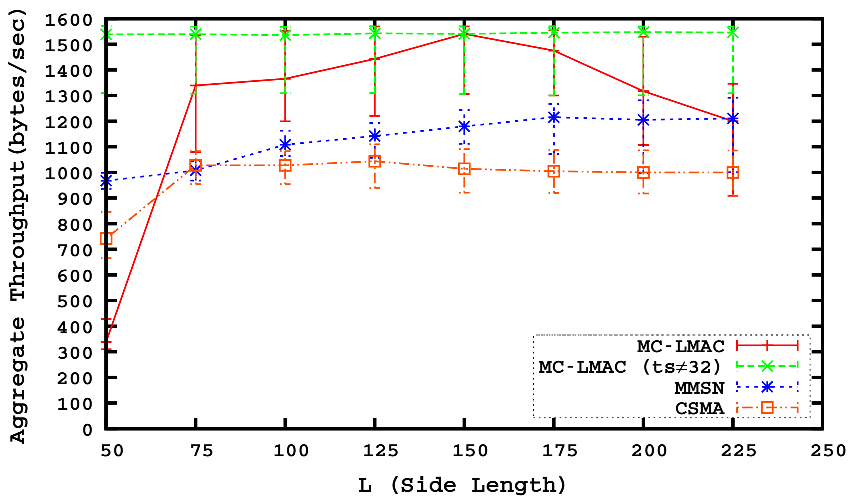


Fig. 17. Aggregate throughput with different densities.

sity and the results are presented with the second line where the maximum throughput can be achieved. In order to achieve maximum performance with MC-LMAC the number of slots per frame should be adjusted according to the density of the deployment.

Aggregate throughput with MMSN continues to increase when the network gets sparser since the contention is lower and the nodes can successfully sense the incoming packets. However, the performance of MMSN is still lower than the throughput of MC-LMAC. The same holds for CSMA. As contention decreases by the decreasing density, CSMA can deliver more packets. However, the performance does not change much when $L \geqslant 100$ and it even decreases a bit due to the failure of the nodes sensing packets at the destination since SINR values tend to be lower as the distances between the nodes increase.

5.5. Impact of traffic patterns

In this section we evaluate the network performance with a different traffic pattern: local gossip. We can think of this scenario as in-network processing such that the source nodes exchange packets before they decide to transmit the data towards the sink node. The nodes in the center of the terrain are assumed to be the sources and they exchange broadcast packets.

All the source nodes are are located in a $30 \times 30\,m^2$ (such that all source nodes are within the carrier sensing range of each other and can receive packets from each other) area in the center of the deployment terrain. We vary the density by changing the terrain size and the number of channels is 8 for MC-LMAC and MMSN. Fig. 18 shows the results in terms of delivery ratios. When the network is dense, the rate of successful deliveries is low. MC-LMAC suffers from the clashes whereas CSMA and MMSN suffer from collisions. In the case of a clash, the receiver randomly selects the sender to listen in MC-LMAC. Additionally, the number of timeslots with MC-LMAC is 32. This is lower than the required number of timeslots for the given density and causes some of the source nodes not to be able to get a slot. In order to achieve higher delivery ratios in denser deployments, the number of timeslots should be increased. In MC-LMAC nodes give priority to receive the broadcast packet during a clash. However, if the clashing transmissions are both broadcast packets, then nodes randomly select a transmitter. However, in a real network observing a scenario where all the sources generate broadcast packets is low. When $L \geqslant 125$, MC-LMAC can deliver more than 98% of the broadcast packets. In contrast, MMSN and CSMA protocols need more sparseness to mitigate the effects of contention. The reason for CSMA to have first a decrease and then an increase in the delivery ratios is that, when the network is too dense the nodes can sense each others' transmissions and backoff. When the network gets sparser, the distances between the nodes increase and sensing of the parallel packets fails more. However, when the network gets even more sparser then the number of nodes in the $30 \times 30\,m^2$ area decreases and contention decreases accordingly.

5.6. Multiple sinks versus multiple channels

As we discussed in Section 3, the limiting factor is the reception capacity of the sink node. Contention-based protocols fail to successfully allocate the medium during high contention around a single sink node. In this section, as an alternative to single-sink multi-channel scenario, we discuss the impact of deploying more sink nodes using a single-channel on the achievable throughput.

Multiple sink nodes are randomly deployed in a $150 \times 150\,m^2$ area. Source nodes transmit packets every 2 s to the closest sink node. Fig. 19 shows the results. In this set, the nodes communicate on a single-channel (except the line titled "MMSN (F10)" in the figure). Our aim is to compare the results of n sink nodes with 1 channel with the results of 1 sink node with n channels, which were presented in Fig. 12. Compared to the results in Fig. 12, both CSMA and MMSN achieve higher throughput since the contention around the sink nodes has lower impact compared to the contention around a single sink. Beyond four sink nodes, MMSN starts to perform better than MMSN with four channels and a single sink node. How-

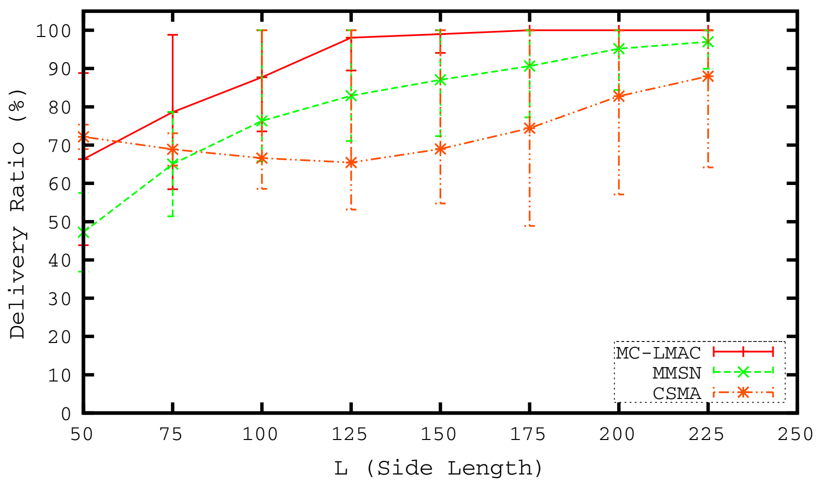


Fig. 18. Delivery ratio with different densities.

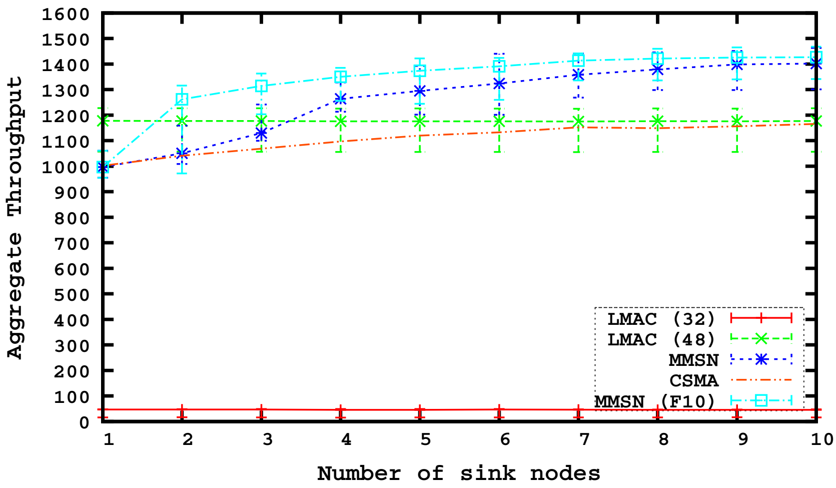


Fig. 19. Aggregate throughput with different number of sink nodes.

ever, around nine sink nodes the aggregate throughput starts to saturate.

In contrast, the single-channel LMAC has a constant lower performance with a single channel and 32 timeslots since most of the nodes cannot get a free slot on a single channel. However, if the number of timeslots is increased to 48, a higher performance is achieved. Although the packet delivery ratio with 48 timeslots is 100%, the aggregate throughput is on the average 75% of the maximum aggregate throughput since the nodes cannot choose the timeslots that are used by their 2nd-hop neighbors on the same channel and this reduces the number of parallel transmissions. The question may arise: why does not the throughput increase with higher number of sink nodes? As we mentioned, the maximum hop count with the given deployment density is around three hops. Although the nodes may transmit to different sinks, according to the timeslot selection rules they cannot get the same timeslot used in their 2-hop neighborhood. Compared with the results in Fig. 12, MMSN and CSMA perform better with multiple sinks but still they cannot achieve the performance of MC-LMAC with multiple channels which has the advantage of collision-free medium access over multiple channels.

The line named "MMSN (F10)" in the figure shows the results when MMSN operates with multiple sink nodes and there are 10 channels available. The performance is better with 10 channels for a smaller number of sink nodes than single-channel communication results given on the line titled "MMSN". However, beyond 7 sinks there is little difference in the performance and aggregate throughput is still less than the achievable throughput with MC-LMAC where there is a single sink to collect data over multiple channels (Fig. 12). This is due to the failure of resolution of contentions and collisions around the sink nodes and due to the random access behavior of MMSN.

As a conclusion, multiple sink nodes can be used as a complementary solution with MC-LMAC that can further improve the achievable throughput for higher data rate scenarios.

5.7. Implementation on sensor nodes

The single-channel LMAC protocol has been implemented and previously tested [5] on Ambient μNode sensor platform. We added the MC-LMAC extension and performed a simple test as a proof of concept using a simple topology where two pairs of nodes are communicating in parallel. The aim of the experiments is to investigate the impact of channel switching on the synchronization of the nodes.

The sensor platform is equipped with Nordic Nrf905 radio that can operate on the 868/915 MHz ISM band. Nodes continuously transmit 32-byte packets every 1/8 s. The conclusions of the experiments are that nodes can change their operating frequency without loosing the synchronization and the protocol runs successfully on the nodes. The aggregate throughput with parallel communication over different channels is doubled, as expected. As a future work, we are interested in comparing the performance of different protocols on a large testbed by using a larger set of parameters.

6. Conclusions

We have presented MC-LMAC, designed for wireless sensor networks with high throughput requirements. MC-LMAC takes advantage of both scheduled and multichannel communication. Scheduled communication has the advantage of minimizing collisions whereas the multi-channel communication overcomes the increased contention and interference on the limited bandwidth and improves the throughput and bandwidth utilization. Nodes can transmit in parallel on different channels without disturbing each other. Simulation results show that, MC-LMAC achieves a throughput very close to the maximum with the increased number of channels and outperforms the MMSN protocol and the channel clustering method for moderate-size, 100-node networks. While MC-LMAC supports higher throughput, it also meets the typical characteristics of WSNs such as energy efficiency and scalability. Besides convergecast traffic MC-LMAC supports broadcasts and local-gossip operations are performed efficiently. As a proof of concept, a simple test case of MC-LMAC demonstrates that nodes do not loose synchronization while switching between frequencies and the protocol runs successfully on the nodes.

As a future work, that would be interesting to incorporate different channel selection mechanisms in MC-LMAC that can allocate not only orthogonal channels but also overlapping channels based on the interference conditions due to transmissions in closer channels or due to external networks or external devices that share the same parts of the spectrum. Another future interest lies in testing the reliability of the protocol in case of channel errors on a testbed.

Acknowledgments

We gratefully acknowledge Dr. Gang Zhou for sharing the source code of the MMSN protocol on Glomosim.

References

[1] K. Langendoen, G. Halkes, Energy-efficient medium access control, in: R. Zurawski (Ed.), Embedded Systems Handbook, CRC Press, 2005 (Chapter 34).

[2] I. Demirkol, C. Ersoy, F. Alagoz, Mac protocols for wireless sensor networks: a survey, IEEE Communications Magazine 44 (4) (2006) 115–121.

[3] K. Chintalapudi, T. Fu, J. Paek, N. Kothari, S. Rangwala, J. Caffrey, R. Govindan, E. Johnson, S. Masri, Monitoring civil structures with a wireless sensor network, IEEE Internet Computing 10 (2) (2006) 26–34.

[4] I.F. Akyildiz, T. Melodia, K.R. Chowdhury, A survey on wireless multimedia sensor networks, Computer Networks 51 (4) (2007) 921–960.

[5] L.F.W. van Hoesel, Sensors on speaking terms: schedule-based medium access control protocols for wireless sensor networks, Ph.D. thesis, University of Twente, Enschede, June 2007.

[6] P. Kyasanur, J. So, C. Chereddi, N. Vaidya, Multichannel mesh networks: challenges and protocols, IEEE Wireless Communications 13 (2) (2006) 30–36.

[7] G. Zhou, C. Huang, T. Yan, T. He, J.A. Stankovic, T.F. Abdelzaher, Mmsn: multi-frequency media access control for wireless sensor networks, in: INFOCOM '06: Proceedings of the 25th IEEE Conference on Computer Communications, 2006, pp. 1–13.

[8] Y. Wu, J. Stankovic, T. He, S. Lin, Realistic and efficient multi-channel communications in wireless sensor networks, in: INFOCOM '08: Proceedings of the 27th IEEE Conference on Computer Communications, 2008, pp. 1193–1201.

[9] I. Katzela, M. Naghshineh, Channel assignment schemes for cellular mobile telecommunications: a comprehensive survey, IEEE Personal Communications (1996) 10–31.

[10] N. Jain, S. Das, A. Nasipuri, A multichannel csma mac protocol with receiver-based channel selection for multihop wireless networks, in: IC3N '01: Proceedings of the 10th IEEE International Conference on Computer Communications and Networks, 2001, pp. 432–439.

[11] S. Roy, A. Das, R. Vijayakumar, H. Alazemi, H. Ma, E. Alotaibi, Capacity scaling with multiple radios and multiple channels in wireless mesh networks, in: WiMesh '05: Proceedings of the First IEEE Workshop on Wireless Mesh Networks, 2005.

[12] J. So, N.H. Vaidya, Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver, in: MobiHoc '04: Proceedings of the 5th ACM International Symposium on Mobile Ad Hoc Networking and Computing, 2004, pp. 222–233.

[13] J. Crichigno, M. Wu, W. Shu, Protocols and architectures for channel assignment in wireless mesh networks, Ad Hoc Networks 6 (7) (2008) 1051–1077.

[14] A. Raniwala, T. Chiueh, Architecture and algorithms for an ieee 802.11-based multi-channel wireless mesh network, in: INFOCOM '05: Proceedings of the 24th IEEE Conference on Computer Communications, vol. 3, 2005, pp. 2223–2234.

[15] Z. Tang, J.J. Garcia-Luna-Aceves, Hop reservation multiple access (hrma) for ad-hoc networks, in: INFOCOM '99: Proceedings of the 18th IEEE Conference on Computer Communications, 1999, pp. 194–201.

[16] A. Tzamaloukas, J. Garcia-Luna-Aceves, Channel-hopping multiple access, in: ICC '00: Proceedings of the IEEE International Conference on Communications, 2000, pp. 415–419.

[17] J. Li, Z. Haas, M. Sheng, Y. Chen, Performance evaluation of modified ieee 802.11 mac for multi-channel multi-hop ad hoc network, in: AINA 2003: Proceedings of the 17th International Conference on Advanced Information Networking and Applications, 2003, pp. 312–317.

[18] J. Chen, S. Sheu, C. Yang, A new multichannel access protocol for ieee 802.11 ad hoc wireless lans, in: PIMRC' 03: Proceedings of the 14th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, vol. 3, 2003, pp. 2291–2296.

[19] J. Zhang, G. Zhou, C. Huang, S. Son, J. Stankovic, Tmmac: an energy-efficient multi-channel mac protocol for ad hoc networks, in: ICC '07: Proceedings of the IEEE International Conference on Communications, 2007, pp. 3554–3561.

[20] P. Bahl, R. Chandra, J. Dunagan, Ssch: slotted seeded channel hopping for capacity improvement in ieee 802.11 ad-hoc wireless networks, in: MobiCom '04: Proceedings of the 10th Annual International Conference on Mobile Computing and Networking, 2004, pp. 216–230.

[21] G.P. Halkes, K. Langendoen, Crankshaft: an energy-efficient mac-protocol for dense wireless sensor networks, in: EWSN '07: Proceedings of the European Conference on Wireless Sensor Networks, 2007, pp. 228–244.

[22] G. Lu, B. Krishnamachari, C. Raghavendra, An adaptive energy-efficient and low-latency mac for tree-based data gathering in sensor networks, Wireless Communications and Mobile Computing 7 (7) (2007) 863–875.

[23] J. Polastre, J. Hill, D. Culler, Versatile low power media access for wireless sensor networks, in: SenSys '04: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, 2004, pp. 95–107.

[24] V. Rajendran, K. Obraczka, J. Garcia-Luna-Aceves, Energy-efficient, collision-free medium access control for wireless sensor networks, in: SenSys '03: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, 2003, pp. 181–192.

[25] I. Rhee, A. Warrier, M. Aia, J. Min, Z-mac: a hybrid mac for wireless sensor networks, in: SenSys '05: Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems, 2005, pp. 90–101.

[26] T. van Dam, K. Langendoen, An adaptive energy-efficient mac protocol for wireless sensor networks, in: SenSys '03: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, ACM Press, 2003, pp. 171–180.

[27] W. Ye, J. Heidemann, D. Estrin, An energy-efficient mac protocol for wireless sensor networks, in: INFOCOM '02: Proceedings of the 21st IEEE Conference on Computer Communications, 2002, pp. 1567–1576.

[28] M. Ringwald, K. Römer, Burstmac – a mac protocol with low idle overhead and high throughput (work in progress), in: Adjunct Proceedings of DCOSS '08: International Conference on Distributed Computing in Sensor Systems, 2008.

[29] Cc2420 single-chip 2.4 ghz ieee 802.15.4 compliant and zigbee(tm) ready rf transceiver, 2008. <http://www.ti.com/lit/gpn/cc2420>.

[30] Ieee 802.15 wpan task group, 2006. <http://www.ieee802.org/15/pub/TG4.html>.

[31] S.C. Ergen, Zigbee/ieee 802.15.4 summary, Technical Report, September 2004.

[32] H. Le, D. Henriksson, T. Abdelzaher, A practical multi-channel media access control protocol for wireless sensor networks, in: IPSN '08: Proceedings of ACM/IEEE International Conference on Information Processing in Sensor Networks, 2008, pp. 70–81.

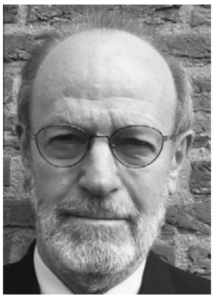
[33] Y. Kim, H. Shin, H. Cha, Y-mac: an energy-efficient multi-channel mac protocol for dense wireless sensor networks, in: IPSN '08: Proceedings of ACM/IEEE International Conference on Information Processing in Sensor Networks, 2008, pp. 53–63.

[34] S. Mastooreh, S. Hamed, K. Antonis, Hymac: hybrid tdma/fdma medium access control protocol for wireless sensor networks, in:

PIMRC' 07: Proceedings of the 18th Annual IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2007, pp. 1–5.
[35] T. Cormen, C. Stein, R. Rivest, C. Leiserson, Introduction to Algorithms, McGraw-Hill Higher Education, 2001.
[36] E. Duarte-Melo, M. Liu, Data-gathering wireless sensor networks: organization and capacity, Computer Networks 43 (4) (2003) 519–537.
[37] Y. Sagduyu, A. Ephremides, The problem of medium access control in wireless sensor networks, in: IEEE Wireless Communications, vol. 11, 2004, pp. 44–53.
[38] S. Chatterjea, L. van Hoesel, P. Havinga, Ai-lmac: an adaptive, information-centric and lightweight mac protocol for wireless sensor networks, in: Issnip'04: Proceedings of the International Conference Intelligent Sensors, Sensor Networks and Information Processing, 2004, pp. 381–388.
[39] B. Krishnamachari, Networking Wireless Sensors, Cambridge University Press, New York, NY, USA, 2005.
[40] X. Zeng, R. Bagrodia, M. Gerla, Glomosim: a library for parallel simulation of large-scale wireless networks, SIGSIM Simulation Digest 28 (1) (1998) 154–161.
[41] P. Gupta, P.R. Kumar, The capacity of wireless networks, IEEE Transactions on Information Theory IT-46 (2) (2000) 388–404.
[42] B. Karp, Geographic routing for wireless networks, Ph.D. thesis, University of Cambridge, 2000.
[43] O.D. Incel, L. van Hoesel, P. Jansen, P. Havinga, Impact of network density on bandwidth resource management in wsn, Technical Report TR-CTIT-05-43, University of Twente, Enschede, September 2005.
[44] O.D. Incel, B. Krishnamachari, Enhancing the data collection rate of tree-based aggregation in wireless sensor networks, in: SECON '08: Proceedings of the Fifth Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, 2008, pp. 569–577.
[45] Nordic semi-conductors, 2005. <http://www.nvlsi.no/>.
[46] O.D. Incel, S. Dulman, P. Jansen, S. Mullender, Multi-channel interference measurements forwireless sensor networks, in: LCN 2006: Proceedings of the 31st IEEE Conference on Local Computer Networks, 2006, pp. 694–701.

Özlem Durmaz İncel is currently a post-doctoral researcher in the Networking Laboratory (NETLAB) of the Bogazici University, Turkey. She received her PhD in computer science from the University of Twente, Netherlands, in March 2009. Her dissertation focused on efficient data collection in wireless sensor networks and was entitled as "Multi-Channel Wireless Sensor Networks: Protocols, Design and Evaluation". She was a visiting student in the Autonomous Networks Research Group of the University of Southern California as part of her Phd studies in 2007–2008. She received both her MSc and BSc degrees in computer engineering from the Yeditepe University, Turkey, in 2005 and 2002 respectively. Her research interests are in the design and analysis of algorithms/protocols for wireless networks, particularly for ad hoc and sensor networks, and in the performance evaluation of computer networks.

Lodewijk van Hoesel received his PhD in the area of wireless sensor networks, in particular energy-efficient communication protocols and he has a MSc in electrical engineering. He has worked for the University of Twente, The Netherlands as a post-doctoral researcher. Van Hoesel is one of the founders of Ambient Systems and is as Senior Systems Architect responsible for many of technology advances of Ambient. He is responsible for wireless networking stack and system architecture of commercial wireless sensor networks and 3rd generation active RFID applications. He is a member of the OptiWSN standardisation committee. He has participated in many research projects related to wireless sensor networks and wireless cooperating objects e.g. EYES, CoBis, e-Sense, AWARE, Sensei and FREE.



Pierre Jansen has performed research on computer architecture and operating systems within industrial and academic contexts. He was group leader for the design of an experimental high performance MIMD computer and holds a list of patents in this field. Since 1982 he has a research position at the University of Twente. His interests gradually shifted from computer architecture to system support for embedded systems. In the recent past he co-operated in projects focusing on "systems on a chip" and "wireless sensor networks". Currently he is retired but he is still active in the mentioned areas.

Paul Havinga is a professor in the Computer Science department at the University of Twente in the Netherlands, chair of the Pervasive Systems research group and founder and CTO of Ambient Systems, in Enschede, the Netherlands. His research themes have focussed on: wireless sensor networks, ambient intelligence, distributed systems, energy-efficient wireless communication, and mobile computing. The common theme in these areas is on the development of large-scale, heterogeneous, wireless, distributed systems. Research questions cover architectures, protocols, programming paradigms, algorithms, and applications. He is the project manager of several large international projects, he is involved as program committee chair, member, and reviewer for many conferences and workshops, and he regularly serves as independent expert for reviewing and evaluation of international research projects for the EU, the US, and international government.