MLMAC-UL and ECTS-MAC -
Two MAC Protocols for Wireless Sensor Networks with Unidirectional Links

Stephan Mank, Reinhardt Karnapke, Jörg Nolte
Distributed Systems/Operating Systems Group
Brandenburg University of Technology
Cottbus, Germany
{smank, karnapke, jon}@informatik.tu-cottbus.de

Abstract

Unidirectional links are a common property of all wireless networks. In many cases, the range of these links exceeds that of bidirectional links by far. Still, most protocols ignore their existence or simply discard them. In this paper we introduce two new medium access control (MAC) protocols called MLMAC-UL and ECTS-MAC that are able to transmit data over unidirectional links and receive acknowledgment messages for them using a neighborhood discovery protocol. Both protocols, as well as two reference protocols, were evaluated in simulations and using a real sensornetwork consisting of TMote Sky sensor nodes.

1. Introduction

Wireless sensor networks rise in importance every day, as more and more possible applications are discovered and implemented. These applications vary, but they all have one thing in common: The nodes need to communicate with each other and/or with a central sink. To enable this communication, often a multi hop path from the nodes to the sink has to be found, which is the duty of a routing protocol. But before longer routes can be established, the communication between two neighboring nodes needs to be possible, which is the job of the medium access control (MAC). Unidirectional links present a problem for both layers, but experiments like e.g., [8] have shown that unidirectional links are common, and can often exceed the range of bidirectional links. This increased range is an advantage for any routing protocol that may use unidirectional links, because it reduces the number of hops, thereby reducing the number of transmissions and consequently time and energy consumption.

Different routing protocols can work in the presence of unidirectional links with different kinds of success. But they all have one thing in common: They need a MAC protocol that can work with unidirectional links. If the MAC-layer can only work on bidirectional links, it will only supply those to the routing protocol, effectively eliminating all advantages the routing protocol might have had.

For this reason we decided to develop a MAC protocol that enables the usage of unidirectional links. But we soon discovered, that there is no such thing as the optimal MAC protocol, and decided to develop two different ones for two different scenarios: MLMAC-UL is designed for dense networks with a high network load while ECTS-MAC should be used in sparse networks with low network load.

This paper is structured as follows: Section 2 describes MLMAC-UL, a TDMA-based MAC. The contention based ECTS-MAC protocol is shown in Section 3 . The evaluation of both protocols can be found in Section 4 while Section 5 shows related work. We finish with conclusion and future work in Section 6.

2. MLMAC-UL

In previous work we introduced MLMAC [4, 5], a TDMA based MAC protocol for mobile wireless sensor networks. MLMAC divides time into frames, which are in turn divided into slots. Each node may use its own slot to transmit data to its neighbors, a slot reappears each frame. Nodes which have a common neighbor must have different slots to prevent collisions. For static networks it is fairly easy to find a schedule for all nodes that fulfills this property, for mobile nodes it is much harder. MLMAC uses an adaptive approach to enable each node in the sensor network to allocate a slot. In this approach there is no predefined starter node as in LMAC [9], rather the synchronization of nodes is started by the node that wants to transmit something first.

In MLMAC a node may have one of 7 different states, and transitions from one to the other under certain conditions. The complete state-machine can be found in [4].

In this section we introduce the changes we made to MLMAC, to stop only detecting unidirectional links and ignore collision that occurred because of them. Rather, MLMAC-UL uses a neighborhood discovery protocol to determine neighbors that can be used to inform the originator of a unidirectional link (the node that can be heard by the other one) about the link and make it usable to forward messages.

The first addition is an independent neighborhood discovery protocol, which is similar to the ones used in AMAC [11] and PANAMA [1] (see Section 5). It transmits the neighborhood table of a node periodically but seldom. In the case of changes, only small update messages are sent. The periodic sending of tables is used to remove any errors resulting from loss of update packets.

Another change in MLMAC-UL is the fact that nodes can give up their slots. If a node has transmitted only status messages for a certain time (e.g., 6 frames) it will inform its neighbors that it is giving up the slot and that it may be used by another node. Moreover, a node may not only hold one slot in MLMAC-UL. Rather, each node can use as many slots as it needs by claiming any unused ones, when it has to transmit lots of data. Once the sendqueue is emptied, it can give the additional slot up one after the other. For this to be effective it is useful to define a larger framesize from the beginning, so that there are always enough free slots available. This ability to hold more slots was introduced to reduce the delay and make MLMAC a better competitor against contention based protocols.

Each node maintains a list of all its neighbors. Three entries define this list: The link quality, the unidirectionality status and the compressed neighborhood information from that neighbor. The link quality can be good (more than 90%reception rate), medium (between 30% and 90 % reception rate) or bad (less than 30% reception rate). The unidirectionality status can be either bidirectional, unidirectional_sender or unidirectional_receiver. The compressed neighborhood list is maintained by the neighborhood discovery protocol and used to identify the 2-hop-neighborhood of the current node.

The statemachine of MLMAC-UL can be seen in figure 1. The arrows in the figure represent the transitions between states and are described in the following.

(1) When a node needs to acquire its first slot it switches into the state `UNSYNC`. (2) The node was in state `UNSYNC` for one frame. It chooses a slot and transitions into the `SYNC`-state. If no slot was empty, the node stays in its current state for another frame. (3) When its chosen slot arrives, the node changes to state `SLOTVERIFY`. (4) The node sends in its slots. After one frame, it reaches the `READY`- state. (5) If a negative acknowledgment for the last slot was received, the slot is deleted and the node changes to state `SLEEP`. (6) The node returns to the `WAIT`-state after a random amount of time. (7) Same as 5. (8) There is data

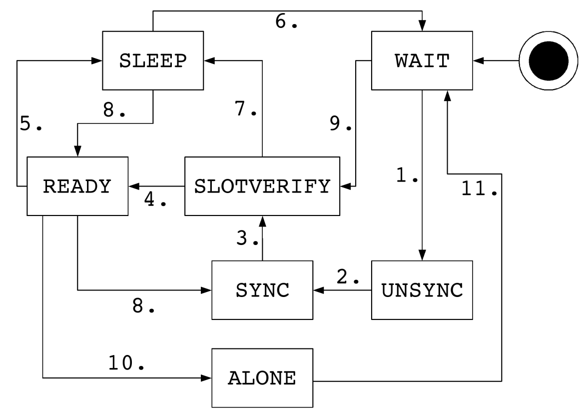


Figure 1. The statemachine of MLMAC-UL

to be transmitted and no neighboring node is transmitting. The node chooses a slot and an identification for the synchronization. After waiting for a random time it transmits the data and switches to `READY`. (9) If this node did not communicate before or it had previously given up one slot a new slot is acquired and the node changes into the state `SLOTVERIFY`. (10) No messages from neighbors were received for 5 frames even though this node is transmitting. This means that this node is either completely isolated, or has only unidirectional links to others, but no incoming link from any of them. This node switches to the `ALONE`-state and does not try to transmit anymore, even when data is available. (11) A message from a neighbor was received, which means that this node is no longer alone or a certain number of frames (e.g., 200) have passed. The node switches to `WAIT` and starts again.

3. ECTS-MAC

ECTS-MAC (Extended Clear To Send MAC) is a contention based protocol for sparse networks with rare communication. It is similar to BW_RES [6] (see Section 5), because it also tries to forward the CTS message to reduce the probability of collision. Unlike BW_RES, it does not calculate distances and power levels. Also, all ECTS messages are sent at the same time, whereas all BW_RES messages are sent one after another. This leads to more collisions of ECTS messages, but saves a lot of time. When a node receives a CTS message it forwards it with a certain probability (see figure 2). Experiments have shown that 50% seems to be the optimal value for sparse networks. If the probability is less, the ECTS message is not received by enough neighbors. If it is higher, the ECTS packets collide more often. These collisions are also the reason why the ECTS-MAC should only be used in sparse networks, as the ECTS packets would increase the network load in a dense network too much. To a certain extend, this effect is alleviated by reducing the probability of sending, but this also leads to more nodes that do not receive the ECTS message. The ECTS-MAC uses the neighborhood discovery protocol described in the previous section to detect unidirectional links. This is necessary to enable transmitting via a unidirectional link, because acknowledgments need to be forwarded to the sender using a second node.

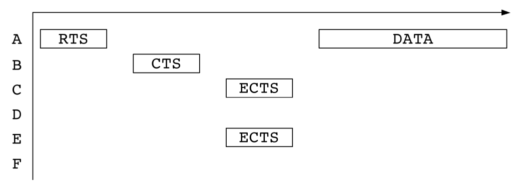


Figure 2. Propagation of ECTS Messages

4. Evaluation

To measure the performance of MLMAC-UL and ECTS-MAC, we evaluated them against two other protocols: The original MLMAC and a modified version of MMP (Multicast Mac Protocol) [2] (see Section 5). As its name suggests, MMP was designed for multicast, not for broadcast. We changed its behavior to enable broadcast transmissions, and to enable it to use unidirectional links. For this, we once again used the neighborhood discovery protocol described above. We call the resulting protocol NMAC (Neighborhood MAC). Because of the neighborhood discovery protocol, the node knows how many neighbors it has and addresses them all in the RTS packet. When a node receives a RTS message it waits for a time corresponding to its position in the RTS before transmitting a CTS. If it has received at least one CTS, the sender of the RTS transmits the data package after the time for all CTS messages has passed. For our evaluations we used the discrete event simulator OMNeT++ [10] as well as real sensornet hardware. In all simulations the nodes transmitted with 19.2 KBit per second and a transmission strength of 10 milliwatt. For our real experiments we used TMote Sky sensor nodes from MoteIV corporation. These feature a MSP430 microcontroller with a frequency of 8Mhz, 10 kB of Ram and 48 kB of flash. The radio module is IEEE 802.15.4 compatible and transmits 250kB/s, we configured the transmission strength to -25dBm to enable a multi hop scenario.

4.1 Single Hop Scenario Simulation

In this scenario the application behavior for a direct one-hop-neighborhood was simulated. The application tried to send as fast as possible. It generated a packet with 110 bytes data every 20 milliseconds, up to a total of 500 packets. In this simulation, all 4 protocols achieved a packet reception rate of nearly 100%. Figure 3 shows the amount of application data transmitted by each protocol. The figure shows that for 2-4 nodes the contention based protocols are able to transmit more data than the TDMA based ones. When more nodes are used, MLMAC-UL can once again gain an advantage because of the usage of multiple slots per node. As the number of slots was not changed for the MLMAC, it always delivers the same amount of data.

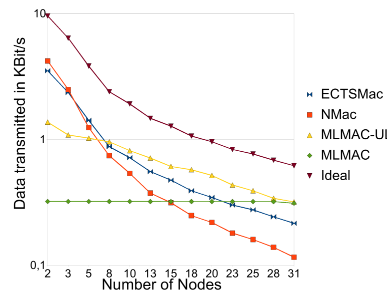
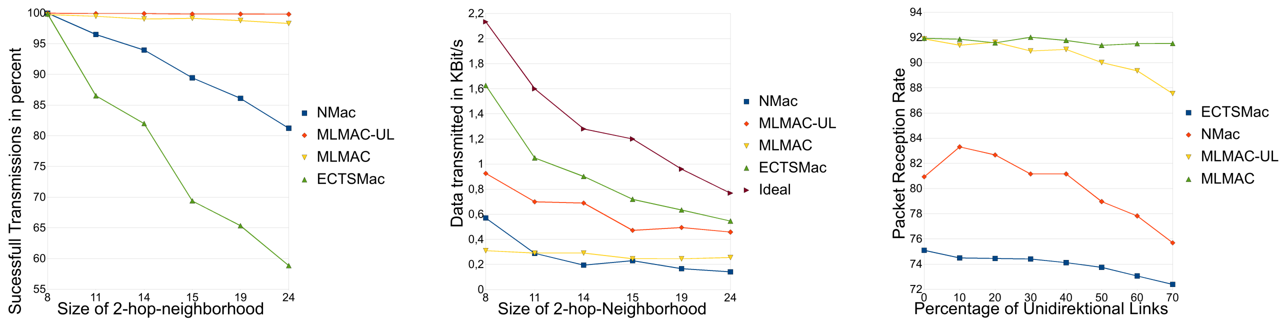


Figure 3. Data Transmitted 1-Hop Scenario

4.2 High Load Scenario Simulation

In this scenario a rectangle of 6 time 8 nodes was simulated. The application tried to send as fast as possible. It generated a packet with 110 bytes data every 20 milliseconds, up to a total of 500 packets. The node in the upper left corner started transmitting, each other node began transmitting its 500 packets after it had received the first packet. From a certain time on, all nodes want to transmit at nearly the same time, thus leading to a high network load. We evaluated the number of packets that were transmitted flawlessly against the number of nodes in the 2-hop-neighborhood. Figure 4(a) shows the percentage of successfully delivered packets for all 4 evaluated MAC protocols. As expected, the two TDMA based protocols were much better suited for this scenario then the contention based ones, which produced too many collisions.

Figure 4(b) shows the average amount of data each node was able to transmit for all 4 MAC protocols and the theoretical maximum. As can be seen, the ECTS-MAC is able to transmit most application data, followed by MLMAC-UL, MLMAC and NMAC. It is important to keep in mind here that this in the amount of application data transmitted, not received. If you correlate the bytes transmitted



(a) Delivery Ratio

(b) Application Data Transmitted

(c) Packet Reception Rate

Figure 4. Rectangle Scenario

to the delivery ratio of the protocols, the performance of the ECTS-MAC drops considerably. The original MLMAC suffers from the fact that nodes may transmit only each frame, whereas MLMAC-UL allows each node to use multiple slots.

Another evaluation using the 6 times 8 nodes rectangle was used to determine the protocols' ability to deal with unidirectional links. To do this, we varied the rate of these from 0 to 70% in steps of 10. Figure 4(c) shows that ML-MAC and MLMAC-UL can cope with the unidirectional links much better than ECTS-MAC and NMAC. Please note that these results were achieved using the neighborhood discovery protocol described above. If it is disabled, the performance of MLMAC-UL drops considerable, because it can no longer detect the unidirectional links and slots are given up too often. For the other protocols the impact is neglectable.

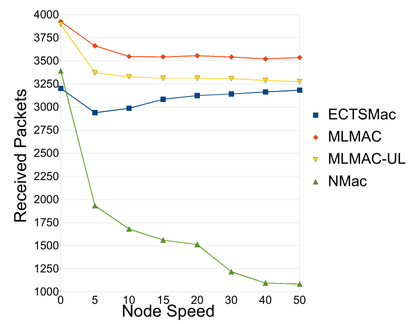


Figure 5. Received Packets

4.3 Mobility Simulation

In this scenario the 4 protocols were evaluated using mobility with different speeds, random starting points and random destinations. The application sent packets of 110 Byte every second. This lead once again to a high network load. Figure 5 shows the number of received packets for each protocol for the different speeds. Once again, MLMAC and MLMAC-UL provide the best results, with ECTS-MAC performing only a little worse. The strong problems of NMAC are the result of a high rate of collisions. This is due to the fact that nodes which are leaving each others vicinity and thus produce a high number of transmission errors are seen as unidirectional links by both nodes and thus not addressed in the RTS message. They don't forward the CTS, which leads to another rise in collisions. The problem gets worse when nodes re-enter each others vicinity shortly after leaving it, because their links remain marked as unidirectional too long.

4.4 Simulated Flooding over 50 hops

In this set of simulations, the performance under low network load is evaluated. For this, a line of 6 to 51 nodes was used, where each node was only able to communicate with its direct neighbors. Table 1 shows the time needed by each protocol to deliver a message over 50 hops. The times for the TDMA protocols are divided once using 5 slots and once using 31. Even though there were enough unused slots, the MLMAC-UL did not acquire new ones, because there was not much data to be sent and the sendqueue only ever held one packet. This leads to nearly the same time (one frame) needed as when using the original MLMAC, as the time for one hop only depended on the frame length. For all protocols, the time needed to reach the last node increased linearly with the number of nodes in use.

Table 1. Time Needed for 50 Hops (ms)

NMac	3,77
MLMAC-UL	70,29
MLMAC	69,25
ECTSMac	3,63
MLMAC-UL 5 Slots	12,32
MLMAC 5 Slots	10,79

4.5 Packet Overhead

This last evaluation in the simulator was based on the same topology as the high load scenario, but the size of the data generated by the application was varied between 20 and 110 Byte. It transmitted at random intervals between 0 and 5000 milliseconds. Figure 6 shows the relative overhead each protocol produced (Protocol Bytes/Total Bytes) for increasing size of the 2-hop-neighborhood. The calculation includes the periodic messages from the TDMA based protocols and the RTS, CTS and ECTS messages from the contention based protocols. It can be seen that NMAC produces by far the highest overhead, followed by the ECTS-MAC. Thus, contrary to common belief, sending periodic status messages does not produce a high overhead.

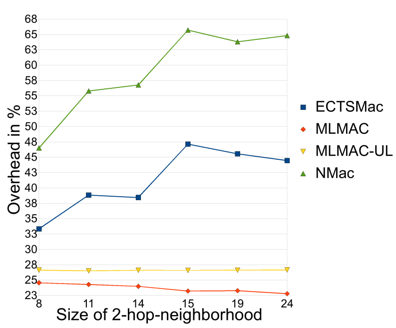


Figure 6. Overhead

4.6 Direct Neighborhood Experiments

In these experiments the application sent 500 packets of size 110 Byte every 10 Milliseconds. They were performed using 3, 7, 11 and 16 nodes. Figure 7 shows that for a small number of nodes all protocols perform relatively well. With an increasing number of nodes the performance of first NMAC and and then MLMAC drop considerably.

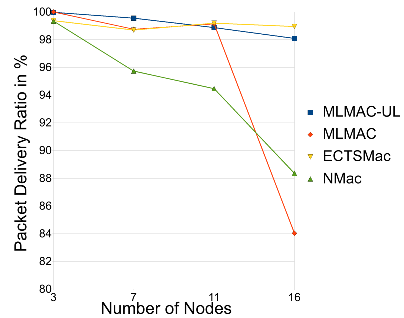


Figure 7. Packet Delivery Ratio

4.7 High Load Scenario Experiments

For these experiments we placed 14 TMote Sky sensor nodes on the floor in a building. As there are no ways to define link quality in a real experiment we could only measure it. Figure 8 shows the resulting communication graph. It can be seen that the radio neighborhood of the nodes and the link quality differ a lot.

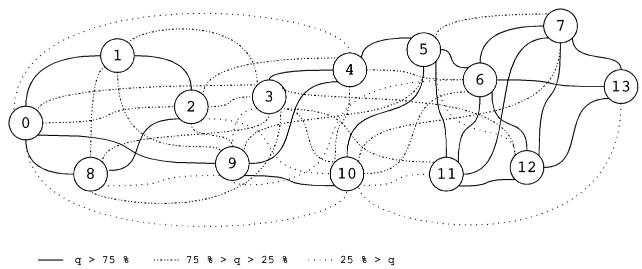


Figure 8. Linkgraph where q denotes the link quality

The application was once again the one producing the high network load. The left side of Figure 9 shows the average time needed to transmit one packet. MLMAC-UL and ECTS-MAC were the fastest ones, with MLMAC following and NMAC bringing up the rear. On the right side of the figure you can see the total number of received packets for each protocol. All protocols received nearly the same amount of messages, with only NMAC being considerably better. But this fact has to be put in perspective: all 4 protocols were evaluated one after another, using the same nodes and, most important, the same batteries. NMAC was the first protocol to be evaluated, which means that is had the advantage of fresh batteries which have been shown to have a positive effect on the range of the transceivers and thus link quality.

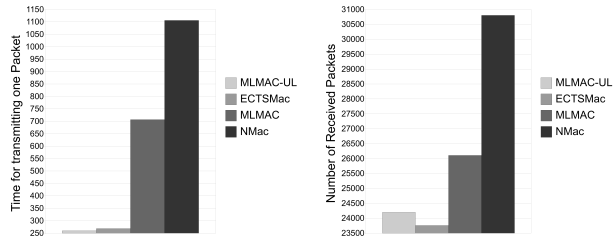


Figure 9. (l)Time to Send and (r)number of received packets

4.8 Memory Consumption

On Figure 10 the memory consumption of the protocols is shown, with 3 slots for the TDMA protocols on the left and 16 slots on the right. It is also differentiated whether the neighborhood discovery protocol was used or not, only the original MLMAC is shown only once, because it never uses that protocol. Please note that the numbers shown are for RAM consumption, the usage of flash memory follows the same distribution. On the figure it can be seen that MLMAC-UL needs most memory and ECTS has the lowest memory consumption. Combining this fact with the other results leads to the observation that for networks with low memory allowance and few nodes the ECTS-MAC should be chosen while the MLMAC-UL is best suited for denser networks with high load.

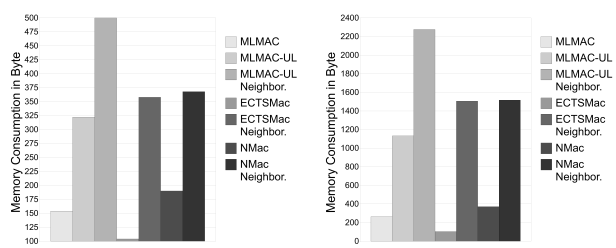


Figure 10. Memory Consumption for 3(l) and 16 slots(r)

5. Related Work

The problem of unidirectional links has been recognized before, and protocols have been developed which can use them.

The Multicast MAC protocol (MMP) [2] does not directly address the problem of unidirectional links, but it offers an easy way to realize a multicast communication, which can easily be increased to broadcast. BMMM [12] and Maclayer Multicast [3] follow a similar approach. MMP is an extension of the IEEE 802.11 MAC in DCF mode. The Request To Send (RTS) message of MMP contains the addresses of all nodes that should receive the multicast message. When a node receives this RTS, it waits a certain time, correlating to its position in the RTS, and sends a CTS. When the slots for all CTS messages have passed and the sender of the RTS has received at least one CTS, it begins transmission of the data packet. After the transmissions, the acknowledgment messages are send by all of the receivers in the same order as the CTS messages. While MMP needs to wait a time corresponding to the number of nodes addressed in the RTS message before sending data packets, ECTS-MAC waits only the time needed for a single ECTS message, thus providing much better scalability. Also, the size of the RTS is reduced drastically in ECTS-MAC, because the list of receivers is omitted.

AMAC [11] is built on top of the Sub Routing Layer (SRL) project [7], which is used to detect unidirectional links. When SRL is used with a routing protocol, it provides the abstraction of a network with only bidirectional links. To do this, it must identify unidirectional links, and find a suitable reverse route leading through multiple nodes. This is done using a reverse distributed Bellman-Ford algorithm. SRL also monitors the network for link changes. AMAC uses the information from SRL to make unidirectional links usable on the MAC layer. Four new types of messages are introduced to make communication over unidirectional links possible by forwarding protocol messages through neighboring nodes. AMAC uses a complex formula to identify the right nodes to forward all four types of messages, while the transmission of ECTS-messages in ECTS-MAC is done probabilistic.

Another extension to IEEE 802.11 is BW_RES [6]. It is based on the principal of forwarding CTS packets to all nodes that may disturb the planned communication. To determine how far a BW_RES message must be forwarded, the transmission strengths of all nodes must be known. The lowest one equals one unit, the highest one N units. The authors show that a CTS message needs to be retransmitted 2N-1 times to ensure that it is heard at least N units distant. A node that receives a CTS message waits between 0 and 6 SIFS before transmitting the BW_RES packet to prevent collisions. While this approach ensures that data communication in the presence of unidirectional links is possible, it delays the transmission and increases the network load proportional to the maximum difference in transmission strengths of nodes. In comparison, the network load produced by ECTS-MAC is rather low, depending on the chosen probability.

PANAMA (Pair wise Link Activation and Node Activation Multiple Access) [1] consists of two different algorithms. PAMA-UN (Pair wise link Activation Multiple

Access Unidirectional Networks) is intended for unicast communication, while NAMA-UN (Node Activation Multiple Access for Unidirectional Networks) supplies broadcast communication. PANAMA is based on CDMA (Code Division Multiple Access) and uses DSSS (Direct Sequence Spread Spectrum). Also, Time is divided into slots. In each slot, nodes with orthogonal spread codes can transmit simultaneously. Codes are reassigned every slot, nodes compete for the codes by comparing their priority. The node with the highest priority has won the medium and all its neighbors configure their radio modules to use its spread code. The link characteristic (bidirectional or unidirectional) is a part of the bandwidth value which is featured in the computation of the priority. The main difference between NAMA-UN and PAMA-UN is the way priorities are computed. In NAMA-UN, the priority depends on the sending node, whereas in PAMA-UN it is calculated using all incoming links of both nodes participating in the communication. The most complex part of PANAMA is the calculation of priorities. Each needs to know the exact priorities of all its neighbors at any time. MLMAC-UL is completely based on local decisions.

6. Conclusion and Future Work

We have introduced two new MAC protocols for wireless sensor networks. MLMAC-UL is an extension to the TDMA based MAC protocol MLMAC, which enables the usage of unidirectional links and multiple slots per node, which decreases latency in high load situations considerably. ECTS-MAC is a contention based protocol with a small memory footprint which informs nodes farther distant that may still disturb a communication by sending an Extended CTS message from a certain percentage of the nodes that received a CTS-message. Both protocols were evaluated in simulations using the OMNeT++ simulator and a real sensor network consisting of TMote Sky sensor nodes. Both presented protocols have their advantages and disadvantages. The choice of protocol depends on a number of factors: The node density, the assumed network load and the memory available. When memory is not a problem, MLMAC-UL is the protocol of choice, because it can handle heavy load situations quite well. On the other hand, if the expected traffic is low and the network is only sparsely populated, ECTS-MAC should be used, because it leaves more memory for the other protocols and the application that should be used on the nodes.

Now that we have designed, implemented and evaluated our MAC protocols, we will begin the design of adequate routing protocols that use unidirectional links, and are based on the MAC protocols described here, maybe using the same or a variant of the neighborhood discovery protocol described in this work.

References

[1] L. Bao and J. J. Garcia-Luna-Aceves. Channel access scheduling in ad hoc networks with unidirectional links. In *DIALM '01: Proceedings of the 5th international workshop on Discrete algorithms and methods for mobile computing and communications*, pages 9–18, New York, NY, USA, 2001. ACM.

[2] H. Gossain, N. Nandiraju, K. Anand, and D. P. Agrawal. Supporting mac layer multicast in ieee 802.11 based manets: Issues and solutions. In *LCN '04: Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks*, pages 172–179, Washington, DC, USA, 2004. IEEE Computer Society.

[3] S. Jain and S. R. Das. Mac layer multicast in wireless multihop networks. In *In COMSWARE '06: Proceedings of the 1st International Conference on Communication System Software and Middleware*, 2006.

[4] S. Mank, R. Karnapke, and J. Nolte. An adaptive tdma based mac protocol for mobile wireless sensor networks, best paper award. In *International Conference on Sensor Technologies and Applications*, 2007.

[5] S. Mank, R. Karnapke, and J. Nolte. Mlmac - an adaptive tdma mac protocol for mobile wireless sensor networks. In *Ad-Hoc & Sensor Wireless Networks: An International Journal, Special Issue on 1st International Conference on Sensor Technologies and Applications*, 2008.

[6] N. Poojary, S. V. Krishnamurthy, and S. Dao. Medium access control in a network of ad hoc mobile nodes with heterogeneous power capabilities. In *in: IEEE International Conference on Communications (ICC), 2001*, pages 872–877, 2001.

[7] V. Ramasubramanian, R. Chandra, and D. Mosse. Providing a bidirectional abstraction for unidirectional ad hoc networks, 2002.

[8] Turau, Renner, and Venzke. The heathland experiment: Results and experiences. In *Proceedings of the REALWSN'05 Workshop on Real-World Wireless Sensor Networks.*, Jun 2005.

[9] L. van Hoesel and P. Havinga. A lightweight medium access protocol (lmac) for wireless sensor networks: Reducing preamble transmissions and transceiver state switches. In *INSS, Japan*, Jun 2004.

[10] A. Varga. The omnet++ discrete event simulation system. In *Proceedings of the European Simulation Multiconference (ESM'2001)*, Prague, Czech Republic, June 2001.

[11] G. Wang, D. Turgut, L. Bölöni, Y. Ji, and D. C. Marinescu. A simulation study of a mac layer protocol for wireless networks with asymmetric links. In *IWCMC '06: Proceedings of the 2006 international conference on Wireless communications and mobile computing*, pages 929–936, New York, NY, USA, 2006. ACM.

[12] R. M. Yadumurthy, A. C. H., M. Sadashivaiah, and R. Makanaboyina. Reliable mac broadcast protocol in directional and omni-directional transmissions for vehicular ad hoc networks. In *VANET '05: Proceedings of the 2nd ACM international workshop on Vehicular ad hoc networks*, pages 10–19, New York, NY, USA, 2005. ACM.