Ideas on Node Mobility Support in Schedule-based Medium Access

Lodewijk van Hoesel, Ayşegül Tüysüz-Erman, Paul Havinga

Department of Electrical Engineering, Computer Science and Mathematics, University of Twente

Postbus 217, NL-7500 AE Enschede, THE NETHERLANDS

E-mail: lodewijk.vanhoesel, aysegul.tuysuz, paul.havinga@utwente.nl

Abstract

Typically, medium access control (MAC) protocols for wireless sensor networks implement synchronised periodic sleeping to conserve energy. We argue that (local) synchronisation between nodes is the main cause why MAC protocols do not efficiently support node mobility e.g. nodes waste valuable energy to resynchronise. In this paper, we present ideas on mobility support in schedule-based medium access control protocols for wireless sensor networks.

The resulting MAC protocol is a hybrid protocol, which combines schedule-based access with contention-based access. The rationale is that the static part of the wireless sensor network can benefit from the high delivery ratio and support for high peak loads of schedule-based access, while mobile nodes can benefit from the natural self-organization of contention-based access. We try our protocol ideas by simulation and real-live experiments.

1. INTRODUCTION

The AWARE project (EU IST-2006-33579) considers self-deploying of wireless sensor networks (WSNs) with autonomous, unmanned aerial vehicles (UAVs) [6]. The AWARE platform targets to enable operation in sites which are difficult or impossible to access and which are without a pre-existent communication structure. One of the focus application scenarios of the AWARE project is disaster management and civil security, in which wireless sensors collaboratively detect critical events (such as fire), or continuously monitor physical conditions of fire brigade personnel, e.g. to prevent them from overheating. In this safety critical application, wireless sensors are the ears and eyes of the AWARE platform, they are added to the network on-the-fly and might be attached to mobile objects. Communication needs to be reliable: self-starting and self-organizing properties are key in the effectiveness of the AWARE platform.

Schedule-based medium access control protocols are generally well suited for (energy-constrained) wireless sensor networks, because this class of protocols minimizes energy-wasting effects like idle-listening, hidden terminal problems and collision of packets. In addition, schedule-based access provides good data throughput characteristics [9]. Therefore, this class of medium access is considered for the AWARE wireless sensor network. In [4], [2], we present LMAC, a schedule-based medium access control protocol for WSNs. However, a drawback of schedule-based medium access protocols is that they often lack the ability to deal with node mobility, which is a key issue for the AWARE scenarios and many other wireless sensor network applications.

The LMAC protocol, which heavily relies on periodic sleep to save energy, may not react fast enough to establish connection with travelling nodes. Consequently, the network performance degrades so badly that it may become unworkable. To be effective in both stationary parts of the network and mobile nodes, we require a protocol that can work efficiently for sensors when they are static, and at the same time this protocol needs to provide an acceptable performance level when sensors are mobile.

In this paper, we discuss adding mobility support to the schedule-based medium access control protocols e.g. LMAC.

2. RELATED WORK

Pham et al. [7] present a mobility-aware sensor MAC protocol for mobile sensor applications called MS-MAC. MS-MAC provides a contention-based access for mobile wireless sensor networks. The protocol takes S-MAC [10] as a starting point and extends the protocol to support mobility.

The SMAC protocol recognises two phases in transceiver usage of network nodes: an receive/transmit period (also called the listen period) and a sleep period. In the sleep period, the nodes turn off their power-consuming transceiver and application packets are backlogged. In this period, the energy consumption of the node's transceiver is minimal and hence the node lifetime is extended beyond the lifetime of a node that is always listening. After the sleep period, the nodes wake-up and listen for communication that is addressed to them, or they initiate communication themselves. This implies that the sleep and listen periods should be (locally) synchronised between nodes. Since the protocol is CSMA(/CA)-based in the listen period, synchronisation does not have to be very strict and nodes can also use their sleep period for communication, if needed.

When a mobile node travels through the network, it might enter regions with different synchronisation and it potentially has to wait (i.e. receive) until it discovers the local duty-cycle (which is announced with so called SYNC messages), before it can communicate with other nodes. In other words, node mobility causes links to be lost and mobile nodes spend (scarce) energy to maintain synchronisation.

MS-MAC [7] uses an approach which establishes "make before break" connections. It proposes a mechanism to prevent that nodes have to wait for synchronisation information. As consequence, mobile nodes do no longer lose connectivity with static nodes in the same network. First, the MS-MAC protocol determines which nodes are mobile by monitoring fluctuations in received signal strengths. Next, it creates an "active zone" around mobile nodes. In active zones, nodes establish synchronisation more often. In other words, the effects of mobility remain the same as in S-MAC, but the severity is reduced by increasing frequency of synchronisation (at the cost of higher energy expenditure at nodes in the active zone).

The Zebra MAC (Z-MAC) protocol [8] has not been designed specifically for mobility, however, it has some interesting properties, because it combines the strengths of schedule-based and contention-based medium access. The main feature of Z-MAC is its adaptability to the level of contention in the network so that under low contention, it behaves CSMA, and under high contention, like TDMA. Contention-based medium access is used as baseline, but the protocol can switch to schedule-based access to enhance contention resolution. A node can be in either of high contention level (HCL) or low contention level (LCL) mode. When a node receives an explicit contention notification (ECN) message from a direct or second order neighbour, it changes its mode into HCL and uses schedule-based access. Otherwise, the node is in LCL mode and uses contention-based access.

When a node has data to transmit, it checks whether it is the owner of the current time slot. If it is the owner of the slot, it takes a random back-off time within a fixed time period . When the back-off time expires, it does a clear channel assessment (CCA) and if the channel is clear, it transmits the data. If the channel is not clear, it waits until the channel is not busy and repeats the above process. If the node is not the owner of the current time slot, then it waits for T_o, and then performs a random back-off within a contention window [T_o ... T_{no}]. When the back-off time expires, it does a CCA. If the channel is clear, it starts to transmit. If the channel is not clear, it waits until the channel is clear, and repeats the above process. To support node mobility, e.g. mobile nodes do not own a time slot and operate always in LCL mode. However, the performance of the protocol has to be investigated in this case.

3. Network structure assumptions

The AWARE wireless sensor network consists of a mix of mobile and static sensor nodes. When we consider a fire-fighter application, the goals of the application are to protect the fire-fighters against dangerous situations and to increase the efficiency of fighting the fire.

In line with the application goals, the health conditions of the personnel are monitored by attaching sensors (hearth rate, body temperature, oxygen saturation, motion sensors and so on) to the firemen. These sensors collaborate in a so called body area network (BAN) to establish locally a context-aware description of the health conditions of the firemen. However, in our work, we assume that these sensors act as one mobile sensor node per fire-fighter.

Other dangers can arise from the environment. For example, forest fire can entrap firemen or other people. To prevent this, UAVs or human operators such as firemen typically deploy static ground sensor nodes, which virtually create e.g. a temperature map of the area of interest, showing hot spots and unsafe areas. We assume that the area of interest is fully covered (radio-wise) by static ground sensor nodes i.e. at every location in the area of interest a reliable radio link can be established with at least one static ground sensor node. Once deployed, these sensor nodes remain at fixed locations.

The MAC protocols, as we discussed in Section 2, assume that every sensor node in the network can potentially be mobile. Each node takes provisions to be mobile (MAC protocol-wise) and all devices implement identical protocols, mobile or not. We take a different approach. Static nodes act as backbone of the wireless sensor network and communicate with other static network efficiently via contention-free schedule-based MAC protocol. Mobile nodes implement a different MAC protocol, which uses contention, but does not require additional initialisation effort or re-synchronisation. Static nodes frequently reserve part of there schedules to communicate with mobile nodes. As further restriction, mobile nodes can only communicate with static nodes. The latter also take care of routing packets to their final destination. In the next section, the protocol is discussed.

4. Design of M(obile)-LMAC

The goal of M-LMAC is to let mobile nodes benefit from the natural self-organising capabilities of contention-based medium access, while static nodes benefit from the advantages of schedule-based access. The schedule-based part of the M-LMAC protocol is inspired upon LMAC [4], however, other schedule-based protocols can be used as well, if the following conditions are met: (1) the protocol periodically reserves the wireless channel for each node, (2) during its time slot, a node can transmit without causing collision at any other receiving node and (3) the time slot can be extended with an addition section without disturbing the protocol operation (e.g. initialisation of the protocol).

A. M-LMAC for static nodes

In this section, we describe how time slots of the schedule-based MAC protocols are extended to interact with nodes that have the predicate "mobile". Figure 2 shows the three packet types that are added to the schedule-based protocol e.g. LMAC. We assume that these packet types can be added without changing the behaviour of the schedule-based part and that nodes are able to determine if a packet belongs to the scheduled or contention part of the time slot.

Static nodes put their transceiver in stand-by mode during the contention part of most time slots. The schedule-based

Fig. 1: Experiment setup. (1) Helicopter carrying a mobile node, (2) static node, (3) fire and smoke generators and (4) fire-fighter carrying a mobile node

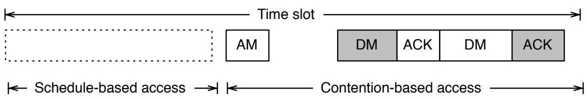

Fig. 2: Extension of the time slot with a part to interact with mobile devices. White blocks represent packets that are transmitted by the static node. Grey blocks represent received packets

MAC protocol reserves per node a certain set of time slots to use. Only during these time slots, static nodes do the following:

- **Transmit announce message (AM)** — This message is transmitted by the static node and marks the beginning of the contention-based access of the time slot. Immediately after the message, the static node switches to "receive mode" at time T_{cw} to interact with mobile nodes and it remains receiving until (1) a complete data message from a mobile node has been received or (2) a time out occurs. All communication with mobile nodes has to be finished at the end of the contention-based access part and the time out T is set to meet this criteria.
- **Receive data message (DM)** — The start time of this message is not scheduled, but mobile nodes content for the wireless medium within time interval T after the AM. In Section 4-B, we describe how the start time is selected. Note that when two or more mobile nodes want to transmit data, collision can occur. If the static node detects a corrupt packet, it switches to standby mode to conserve energy.
- **Transmit acknowledgement message (ACK)** — If the static node successfully received a DM from a mobile node, it checks if it needs to transmit a (backlogged) message to the mobile node. The primary function of the ACK is to acknowledge the DM of the mobile node, but it also carries a notification for the mobile node if the static node will follow the ACK with a data message.
- **Transmit data message** — The static node (optionally) transmits the backlogged message for the mobile node, else both mobile and static nodes enter idle mode to conserve energy.
- **Receive acknowledgement** — The mobile node acknowledges the DM of the static node, if correct received, and the static node removes the backlogged message.

Next, we describe how mobile nodes make use of the communication backbone facilities that are offered by the static nodes.

B. M-LMAC for mobile nodes

Mobile nodes use a simple communication cycle. They primarily keep their transceiver in standby mode to conserve energy and only "wake" if they have higher layer data to transmit or when a predefined time has elapsed since the last communication cycle. The latter is to ensure that the mobile devices can be reached periodically e.g. to change the subscription rate.

The communication cycle of mobile nodes looks as follows:

1) **Find contention-window** — Scan the wireless channel for an AM. This indicates that a static node is ready for receiving a data message. If no AM is detected, the

mobile node is outside the coverage area of the static nodes; it should periodically scan for AMs.

2) **Content for the medium** — In this step, a listen-before-talk strategy is used to minimise the probability of interfering an ongoing transmission. If the channel is clear just before the selected start time, then the mobile node continues with the transmission of a DM. Otherwise, the transmission should be deferred to prevent a collision. The described mechanisms are commonly used in contention-based MAC protocols.
   The contention-window [T_{cw} ... $T_{cw}+T$] is divided in n equal time intervals with duration T_u, which is shorter than the shortest possible packet. The mobile node selects a random integer i ($0 \leq i < n$) and computes the transmission start time $T_s = T_{cw} + iT_u$ such that T_s falls within the contention window. If $T_s = T_{cw}$, the mobile node immediately begins transmitting a DM, otherwise, the mobile node waits until $T_s - T_u$ with its transceiver in low-power state and performs a channel assessment. If the channel is not clear, the mobile node backs off, puts its transceiver in standby and retries to communicate at a next opportunity.
3) **Transmit DM and receive ACK** — The channel assessment indicated a clear channel and the mobile node transmits the prepared DM at T_s. The static node takes care of the routing of the message towards its destination. The DM should be confirmed with an ACK, if not, the mobile node can decide to retry the sending in a new communication cycle. Note that the DM can still collide.
4) **Receive subscriptions etc.** — The static node has potentially a backlogged packet for the mobile node. If so, the mobile node receives the DM and if correctly received, it transmits an ACK message. Note that sub
5) **Switch to idle mode** — At the end of the communication cycle, the mobile node can switch to idle mode to conserve energy. Since the static network can only passively send data to the mobile node, the mobile node should from time to time poll the static network for configuration data.

We assume that during one communication cycle, a mobile node stays within communication range of a static node which triggered it to content for the medium. In our M-LMAC implementation (Section 5), the whole contention part of a time slot takes $1/32$s, in which a mobile node travelling at 60km/h has a displacement of roughly $1/2$m. The displacement is comfortable small compared to the expected transmission range of the wireless sensor node (100m, [3]) and hence our assumption is justified.

In M-LMAC, the static nodes need to backlog messages for mobile nodes. Since the network is unaware of the location of mobile nodes and the location might change, we assume that messages for mobile nodes are broadcasted to all static nodes. Hence it does not matter with which static node a mobile node communicates; it will receive the backlogged message. But to prevent it from receiving multiple copies of the same message, the static node communicating with the mobile node must inform other static nodes that the backlogged message can be deleted. We leave the topic of backlogging messages to our future work.

Since the mobile nodes use a ”first fit” approach in communication and do not rely on any topology information, the proposed protocol is very well suited for mobile scenarios.

5. Experiments with M-LMAC

The main objectives of a field tests [6] were to obtain feedback if our ideas on mobility support in schedule-based MAC protocols are feasible to implement.

The M-LMAC protocol (in combination with LMAC [4]) is implemented for the wireless sensor platform used in the AWARE project (based upon a 868MHz transceiver, [5]). The original LMAC protocol uses 32 time slots per MAC frame, which has a duration of 1s. For the static nodes in our setup, we add the M-LMAC contention part with a total duration of $1/32$s, such that the total time slot duration is doubled (i.e. 16 time slots per second). Additionally, the static nodes implement shortest path routing of messages to one static data sink.

The mobile nodes implement the protocol as described in Section 4-B, however, the mobile nodes do not implement the listen-before-talk strategy, because the used transceiver does not provide means to perform a CCA [5]. In other words, mobile nodes always select to transmit in the first time interval in the contention-window i.e. always $i = 0$.

The experiments were carried out in a simulated fire scenario (Figure 1, http://www.aware-project.net). Static nodes are deployed in the area surrounding a simulated —by means of a scaffolding structure– building. Fire-fighters had access to the simulated three-floor building by means of a ladder. Smoke and fire machines were used to simulate fires in and around the building. The mobile nodes were equipped with temperature sensors, which actually are —as we learned during the experiments– not suited to detect a (simulated) fire. However, the mobile sensors report the measured temperature once every 10s to a static node, which route the information to the data sink. The data sink communicates the sensor readings to other entities in the AWARE platform and logged the data. Six mobile nodes and eight static nodes were active during our small scale experiment. Four mobile nodes were deployed within the building, one was carried by a fire-fighter and one was attached to a small helicopter, which circled around the building.

Figure 3 shows the data collected from a mobile sensor very close to the simulated fire. After the fire is lighted, the temperature value of the sensor rises suddenly from 18^oC to 54^oC and triggers a fire alarm. At the same time, a fireman started to walk inside the building carrying a mobile sensor. However, the sensed temperature increase of the mobile sensor is low in this time period. This is due to the small size of the fire and the windy weather conditions. Interestingly, we see a temperature increase of the mobile sensor node carried by the fireman when he is in the fire truck (at 48200s).

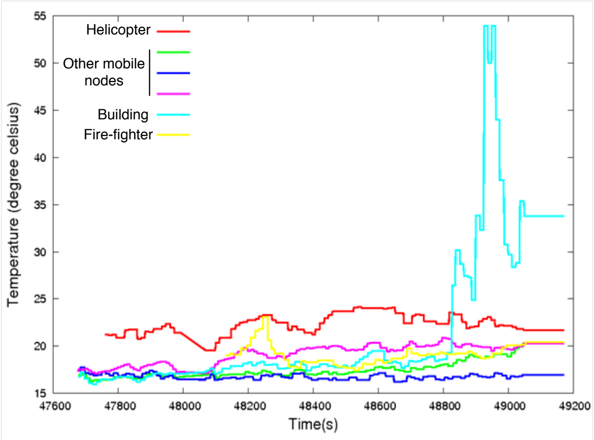

Fig. 3: Temperature measurement results of M-LMAC experiments

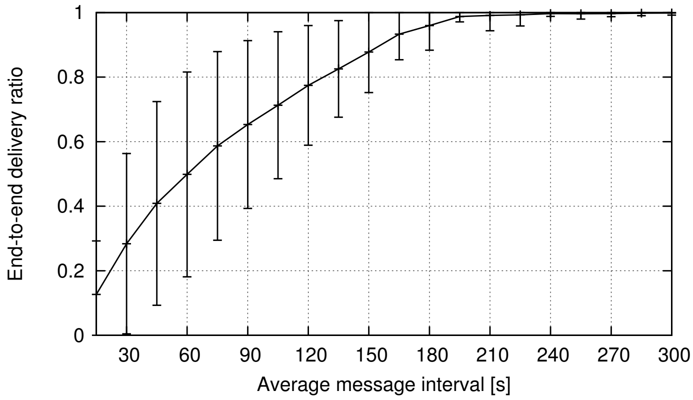

Fig. 4: Delivery ratio results (simulation)

The experiments showed that a real-live application can be built with M-LMAC. The results show that the networking functions well (i.e. 100% delivery ratio) under both low mobility (4-5 km/h for walking humans) and high mobility (40-50 km/h for UAVs). However, we leave it to our future work to determine performance of the M-LMAC protocol, especially in larger scale networks. The next section presents the performance of M-LMAC in a evaluation model.

6. Delivery ratio and energy consumption

We tested the protocol in our OmNet++ (http://www.omnetpp.org) evaluation model and evaluate the protocol from mobile node perspective: (1) delivery ratio of messages, and (2) expected lifetime. We used measurements of current consumption etc. on nodes to create a realistic physical layer model (Table 1).

Table 1: Transceiver parameters and settings used in simulation

Description	Value
Bit rate (after Manchester encoding)	250 kbps
Start up time from standby to transmit/receive	192 μs
Duration of automatically added header	192 μs
Length of automatically added CRC	16 bits
Maximum packet duration	4.3 ms
Current consumption Tx	30 mA
Current consumption Rx	30 mA
Current consumption Idle	10 μA
Current consumption during switch	30 mA

Our simulation setup uses 27 static nodes, 1 data sink and 300 mobile nodes. All are deployed in a 140 x 80m area and have a circular transmission range of 30m. The MAC protocol being used is LMAC combined with M-LMAC and the mobile nodes generate packets of 64 bytes addressed to the data sink with a random (uniformly) interval. RF channel effects and dynamics are ignored. If packets are not acknowledged, they are (at most) resend three times. In our simulation model, mobile nodes use CCA ($n = 4$).

Figure 4 presents our (averaged) end-to-end delivery ratio results for packets generated by mobile nodes. Note that packets in our simulation can get lost due to different causes: (1) collision in the contention part of the time slot, and (2) full buffers in the static nodes (due to memory constraints in real sensor nodes, we carried out the simulations with a buffer size of 8 data messages). When the average packet interval is short, the load of the network is high and due to the two above causes packets are lost. When mobile nodes generate packets once every 3min (or with larger interval), all generated packets arrive at the data sink.

In Tables 2 and 3, transceiver duty-cycle results are presented (obtained with a realistic physical layer model). Given the parameters of Table 1, the static nodes consume on average 0.86 mJ/s (with standard deviation 0.04 mJ/s) and the mobile nodes 0.12 mJ/s (0.04 mJ/s). These estimates exclude all components of the wireless sensor node platform, except the transceiver. On two AA batteries with capacity of 1Ah, the static nodes would survive ≈0.8 year and mobile nodes ≈5.9 years, excluding energy-consumption of other components of the hardware platforms.

Table 2: Transceiver duty-cycle of static nodes (mobile nodes reporting on average once every 90s)

State	Avg. time fraction	St.dev.
Idle	97.2 %	0.1 %
Rx	2.6 %	0.1 %
Tx	0.2 %	0.1 %

Table 3: Transceiver duty-cycle of mobile nodes (reporting on average once every 90s)

State	Avg. time fraction	St.dev.
Idle	99.7 %	0.1 %
Rx	0.4 %	0.1 %
Tx	0.0 %	0.0 %

The energy consumption of the static nodes is much higher than the energy consumption of the mobile nodes, mainly because the communication is much more intensive in the backbone of static nodes (these nodes also take care of forwarding data from other nodes, something the mobile nodes do not do).

7. Conclusion

In this paper, we presented our ideas on adding mobility support to schedule-based MAC protocols. Typically, MAC protocols for wireless sensor networks apply (local) synchronisation between nodes to be energy-efficient. However, synchronisation is in the way of efficiently supporting node mobility. Related work shows —e.g. in the SMAC case– that mobile nodes consume more energy than static nodes, because they need to rediscover local timing. We propose M-LMAC, a hybrid protocol, which combines schedule-based access with contention-based access. The rationale is that the static part of the wireless sensor network can benefit from the high delivery ratio and support for high peak loads of schedule-based access, while mobile nodes can benefit from the natural self-organization of contention-based access.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Elsevier Computer Networks*, 38(4):393-422, 2002.

[2] A. Fehnker, L.F.W. van Hoesel and A.H. Mader. Modelling and Verification of the LMAC Protocol for Wireless Sensor Networks. *Integrated Formal Methods conference (IFM 2007)*, Oxford, UK, June 2007.

[3] J. Geerlings, L. F. W. van Hoesel, F. W. Hoeksema, C. H. Slump and P. J. M. Havinga Spatial reuse of wireless medium in multi-hop wireless sensor networks. *Proceedings of IEEE SPS-DARTS 2007*, Antwerp, Belgium, March 2007.

[4] L.F.W. van Hoesel and P.J.M. Havinga. Design Aspects of An Energy-Efficient, Lightweight Medium Access Control Protocol for Wireless Sensor Networks. *CTIT Technical report series, TR-CTIT-06-18*, June 2006. ISSN 1381-3625.

[5] Nordic, Single chip 433/868/915 MHz Transceiver nRF905, Nordic Semiconductor, January 2005.

[6] A. Ollero, M. Bernard, M. La Civita, L.F.W. van Hoesel, P.J. Marron, J. Lepley and E. de Andres. AWARE: Platform for Autonomous self-deploying and operation of Wireless sensor-actuator networks cooperating with unmanned AeRial vehiclEs. *IEEE International Workshop on Safety, Security and Rescue Robotics (SSRR 2007)*, Rome, pages 1–6, ISBN 978-1-4244-1569-4, September 2007.

[7] H. Pham and S. Jha. Addressing Mobility in Wireless Sensor Media Access Protocol. *In Proceedings of Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2004)*, Melbourne, Australia, December 2004.

[8] I. Rhee, A. Warrier, M. Aia and J. Min. Z-MAC: a hybrid MAC for wireless sensor networks. *In Proceedings of the 3rd international conference on Embedded networked sensor systems*, San Diego, USA, November 2005.

[9] Y.E. Sagduyu and A. Ephremides. The Problem of Medium Access Control in Wireless Sensor Networks. *IEEE Wireless Communication Magazine*, 11(6)44–53, December 2004.

[10] W. Ye, J. Heidemann, and D. Estrin. Medium Access Control with Coordinated, Adaptive Sleeping for Wireless Sensor Networks. *IEEE/ACM Transactions on Networking*, 12(3):493–506, 2003.