K.G. Saur Verlag, München, 2009

P. Hurni

Unsynchronized Energy-Efficient Medium Access Control and Routing in Wireless Sensor Networks

Philipp Hurni received his B.S. and M.S. degree in 2006 and 2008 from University of Bern, Switzerland. He received the "KUVS GI/ITG" price for his master thesis "Unsynchronized Energy-Efficient Medium Access Control and Routing in Wireless Sensor Networks" in the year 2008. He has been working as a Researcher at Purdue University during Winter/Spring 2008. Since April 2008, he is working towards PhD in the field of energy-efficient MAC and routing protocols in wireless ad hoc and sensor networks at University of Bern.

SUMMARY

The challenge in the design of energy-efficient medium access protocols for Wireless Sensor Networks (WSNs) is to use the radio only in an "on demand" manner, as the radio transceiver is a major source of WSN energy consumption. Energy-efficient MAC protocols periodically switch the radio transceiver hardware between the costly operation modes receive and transmit, and an energy-saving sleep mode. Most existing power saving MAC approaches synchronize the nodes in the network to let the nodes synchronously wake up at designated instants to exchange pending messages. Synchronization however is expensive and uneasy to achieve. With low traffic, the overhead for maintaining synchronization and slot coordination across a large multi-hop WSN may exceed the energy spent for the actual data. Mechanisms renouncing on synchronization are likely to be more efficient in low-traffic scenarios. We investigate modifications and optimizations on recently proposed fully unsynchronized power saving MAC protocols for wireless sensor networks based on asynchronous wake-up patterns and intended for sensor networks with low traffic requirements.

1 INTRODUCTION

This paper is an excerpt from the investigations related to unsynchronized energy-efficient MAC protocols carried out within [1]. The main focus of [1] lies in the design and optimization of medium access control and routing mechanisms tailored for use in wireless sensor networks renouncing on costly synchronization schemes. This paper provides a brief outline over the ideas and concepts developed in the context of [1].

2 WISEMAC

Quite a few WSN medium access protocols based on asynchronous wake-ups have recently been proposed, such as WiseMAC [2, 3], B-MAC [4] or X-MAC [5]. WiseMAC is one of the most established protocols of this kind, and currently one of the most energy-efficient MAC protocols for scenarios with low or variable traffic requirements. WiseMAC senses the carrier for a preamble signal with short periodic duty cycles. All nodes in the network sample the medium with a common basic cycle duration T, but their wake-up patterns are independent and left unsynchronized. When transmitting a frame, a preamble is prepended for alerting the receiving node. When the receiver's wake-up pattern is still unknown, the duration of the preamble equals the full basic cycle duration T. The own schedule offset is then piggybacked to the frame and transmitted to the receiver. After successful frame reception, the receiver node piggybacks its own schedule to the respective frame acknowledgment. Received schedule offsets of all neighboring nodes are subsequently kept in a table and are periodically updated. Based on this table, a node can determine the wake-up patterns of all its neighbors, which in turn allows minimizing the preamble length for the upcoming transmissions

3 IMPACT OF THE CONSTANT SAMPLING INTERVAL

WiseMAC switches the transceiver between receive and sleep state in a simple periodic manner with a fixed constant interval T. Once a node has been turned on, it starts alternating between receive and the sleep state in its individual wake-up pattern. This leads to uniformly distributed asynchronous wake intervals of the network's nodes over time. We found that the WiseMAC fixed periodic wake-up pattern yields three main drawbacks:

a) Invisibility / Non-(Re)discovery

The WiseMAC wake-up pattern with its constant sampling period T makes it impossible for nodes with non-intersecting wake intervals to learn about the presence of their local neighbors by overhearing messages. In order to let nodes discover the neighborhood, one will need to introduce bootstrapping phases during which the neighboring nodes exchange their id and schedule information. This will have to be achieved using costly WiseMAC broadcasts with preambles stretched to the entire interval duration T. A fixed wake-up pattern further makes it impossible for nodes to rediscover each other by overhearing neighboring nodes' transmissions. If two node's wake-up patterns are too different, nodes will never overhear each other's transmissions. However some nodes in the WSN might get in

10.1515/piko.2009.008

and out of range of each other, e.g. due to slight movement, or some nodes being deployed at a later point of time. Therefore mechanisms for neighborhood rediscovery have to be designed.

b) Shadowing Problem with Near Wake Intervals

In situations with increased load, WiseMAC nodes with near wake-up intervals can systematically hinder each other from receiving messages. With fixed static wake-up pattern, this problem can have severe performance-restraining consequences. Consider nodes B and C in Fig. 2, which share almost the same wake-up pattern. Node C always slightly precedes the wake-up interval of node B. If two respective neighbors A and D want to reach B and C, the transmission A→B will always be shadowed by the transmission D→C, as node C always wakes up earlier. D will always find the medium idle and will transmit to C, whereas B will wake up, notice that there is a transmission going on and go back to sleep. Node A will have to wait until there is no message transfer to C such that it can finally transmit to B. This leads to a high latency for A's packets whenever there is traffic destined to C. The wake pattern of C therefore "shadows" the wake pattern of B and hinders it from receiving frames.

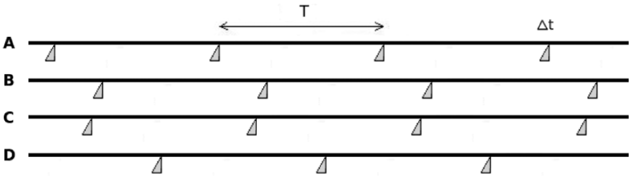

Fig. 1 WiseMAC nodes with independent wake-up patterns

c) Systematic Overhearing with Near Wake Intervals

Nodes with "near" wake-up intervals can suffer from systematic overhearing. The later node B in Fig. 2 will always overhear messages that are destined to node C. When nodes transmit frames to C, node B will always wake up, notice that there is a transmission going on, remain awake and overhear it. A static basic interval T leads to the property that nodes either *never* overhear each other's transmissions that they *constantly* overhear each other's transmissions. This is an undesirable property, as nodes hearing every transmission of a neighbor will be prone to much higher energy wastage and will drain out of energy first. Depending on their location and role in the WSN this might be disadvantageous. The problem can have severe impact on the service properties for a large part of the nodes, especially if C and D are central nodes in the WSN which have to forward data packets from large sub-trees. If D continuously generates or forwards packets, the traffic that needs to be forwarded by B is blocked. This can lead to high delays for packets that need to be routed along B or even packet losses due to buffer overflows. Node B is likely to drain out of energy first, as it will have to handle its own transmissions and probably overhear many transmissions to C.

4 MOVING INTERVALS WAKE-UP PATTERN

In this section we outline a mechanism and modification on the WiseMAC fixed periodic sampling pattern and prove that our proposed scheme delivers better results than original WiseMAC in situations with increasing traffic rate. We verify our observations with experiments carried out in simulation in section 4a) and on a real sensor hardware test bed in section 4b).

The WiseMAC fixed periodic wake-up can be improved in a quite simple manner by introducing a time movement function for the wake intervals. This function shall determine the instants for the wake intervals as depicted in Fig. 3. An initially chosen interval T shall be kept as base for the movement function, but the medium samplings shall not always begin at the start of the interval, but move from one slot to the next slot and jump back when reaching the end. The function that determines the moving wake interval position is deterministic and predictable.

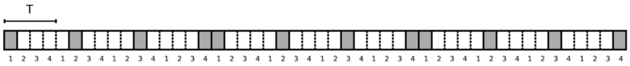

Fig. 2 Moving Wake Interval Scheme

The concept of moving wake intervals softens the degrading impact of the three aforementioned problems. With moving wake intervals, nodes sooner or later overhear neighboring nodes' transmissions. The moving wake interval ensures that – given some periodic traffic – this happens within limited time. Thanks to the mobility of the wake interval, the probability that two nodes share exactly the same behavior, thereby constantly hindering each other from receiving messages is much lower. It might happen that some nodes tact starts are similar, as in WiseMAC. But as nodes choose the initial position (offset) of the wake-interval at boot time in a random manner, the dangerous situation that one node's wake interval always slightly precedes another node's wake interval is far less probable.

a) Moving Intervals Simulation Experiment

In order to evaluate and compare the properties of the moving intervals concept in a sensor network scenario, we deployed and simulated 90 nodes distributed uniformly across a 300m x 300m plane with a sink node in one corner. We used the OMNeT++ network simulator [6] and the mobility framework [7], which supports simulations of wireless ad hoc and mobile networks. It calculates SNR (Signal-to-Noise) ratios according to a free space propagation model. For performance evaluation of power saving MAC protocols, one has to carefully model the transceiver's energy consumption in its respective operation modes and state transition phases as well as the transition delays and their respective costs. We used an energy consumption and state transition model with three operation modes (sleep, receive and transmit) and applied the respective energy consumption values, transmission rate and state transition delays of the TR1001 low-power transceiver [8]. This radio transceiver chip is used in our real-world sensor hardware test bed and many other existing platforms. CPU processing costs are not taken into account. The simulation parameters of the simulation environment and the WiseMAC implementation are listed in the tables below.

transmission rate	19'200 bps
transmitter power	0.1 mW
communication range	50 m
carrier sensing range	100 m
path loss coefficient á	3.5

WiseMAC interval duration T	250 ms
WiseMAC duty cycle	5%

MAC Header	10 byte
Payload	10 byte
Simulation runs	100

Every node calculates the shortest path to the sink and forwards its packets along its gateway node. Node start reporting data towards the sink after an initialization phase according to a Poisson traffic model. By linearly increasing the traffic rate ë, we observed how the wake-up patterns on the MAC impact the service characteristics of the network.

Fig. 3a illustrates the throughput with increasing traffic rate. Comparing the original WiseMAC wake-up pattern with the moving intervals approach, we can claim a slight performance improvement with increasing traffic rate in the range of 10-15%. The payoff of the mechanism incorporating moving wake intervals is only measurable with increasing traffic. As long as there is not much traffic, the situation that two stations with a similar wake-pattern concurrently need to handle traffic does not yet occur. With increasing traffic the problem leads to congestion problems, which impact the throughput and delay. With the fixed static wake-up pattern of WiseMAC, this impact obviously occurs earlier. As one can clearly see in Fig. 3a, the introduction of moving intervals leads to a higher throughput with increased traffic intensity.

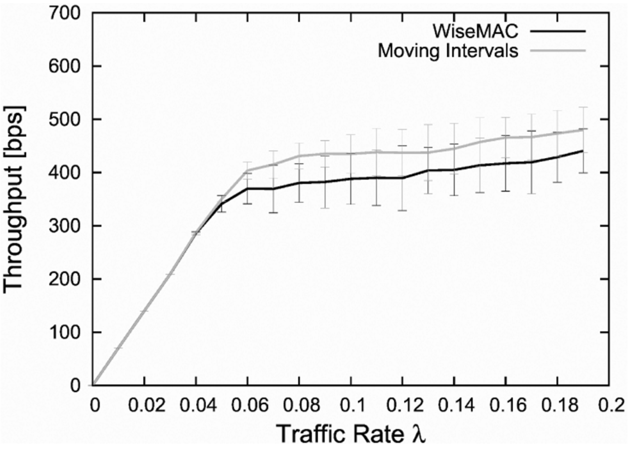

Fig. 3a Throughput in OMNeT++

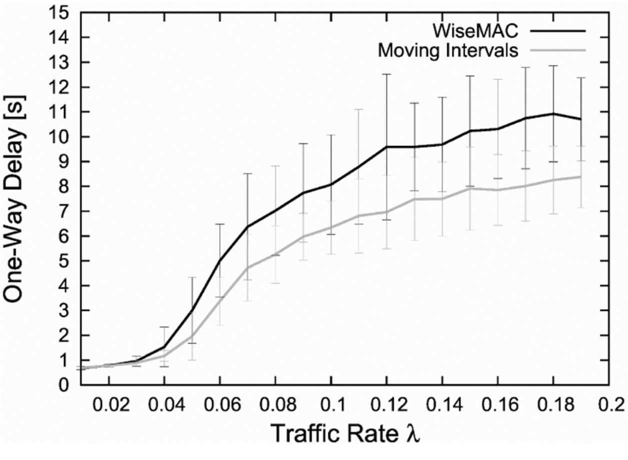

Fig. 3b One-Way Delay in OMNeT++

Fig. 3b further depicts the average one-way delay of the packets arriving at the sink station. The fixed static wake-up pattern of WiseMAC performs worse than the moving intervals approach. The better overhearing avoidance between any two neighboring nodes is accountable for the efficiency and performance gain, which reaches 20-30% with increasing traffic rate.

b) Moving Intervals Experiment on the Embedded Sensor Boards

We implemented the moving intervals scheme with one moving interval moving inside a tact window on the Embedded Sensor Boards (ESB) sensor platform. ESBs run the sensor node operating system ScatterWeb [8] and are equipped with a microcontroller MSP430, various sensors and a TR1001 low power radio transceiver [7]. We prove that the moving intervals wake-up pattern provides a more reliable overhearing avoidance. With a simple experiment, we show that the fixed static WiseMAC tact leads to bad performance if two particular nodes' wake intervals are near each other.

The experiment involves two sender nodes A and D transmitting packets towards the receiver nodes B and C, as illustrated in Fig. 2. In a multi-hop topology with many nodes, the situation is quite probable to occur. We assume that node C's wake interval slightly precedes node B's interval, as depicted in Fig. 2. The probability that this problem arises is determined by parameters of the WiseMAC protocol, the state transition delays of the transceiver and the bit rate. With our default settings of the ESB prototype (T = 500 ms, 1% duty cycle, 19'200 bps) we ran 100 tests by independently booting two nodes and testing if their wake pattern hinders each other from receiving messages. We found that the probability that this problem happens is significant. Out of 100 independent tests, the problem occurred 23 times.

The results in Fig. 4a and 4b show that the problem can be solved or at least softened with the moving intervals approach. We generated traffic from node A to B and from node D to C and measured the service characteristics of the traffic arriving at the sooner node C and the later node B. Traffic is generated according to a Poisson process of increasing rate ë ranging from 0.1 to 0.8 packets per second. Fig. 4a and 4b depict the delay of packets generated by the application layer in node A and D arriving at nodes B and C with both approaches. The delay is measured as the time the application generates the packet and the time the receiver node application layer receives and decapsulates the packet. Fig. 4a depicts the delay when the nodes apply the WiseMAC fixed static tact. In any case node D transmits a packet to C, node A can not deliver own packets to B and has to back off and wait for the next cycle. The delay of packets received by node B therefore increases steeply with increasing traffic rate, as node B's wake-up is shadowed by node C's wake-up. Fig. 4b depicts the delay when all nodes' wake intervals behave according to the moving intervals concept. Obviously this MAC scheme indeed performs better. As the wake interval shifts inside a cycle with respect to the movement function, chances are low that the two receiving nodes B and C systematically hinder each other from receiving messages. In contrast to the fixed static pattern of WiseMAC, it is less probable that both the start of the cycle and the initial configuration of the nodes' wake intervals lead to the problem of permanently near wake intervals. It leads to the property that the wake intervals come near each other and cross each other infrequently but do not always hinder each other from receiving messages. The delay of packets destined to node B does not suffer from steep in-

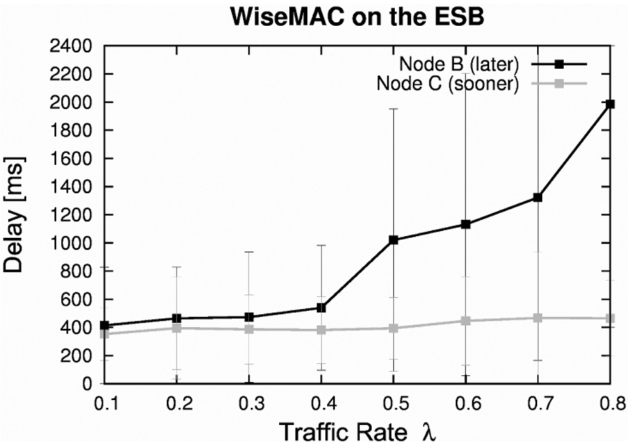

Fig. 4a Delay with WiseMAC static intervals on the ESB

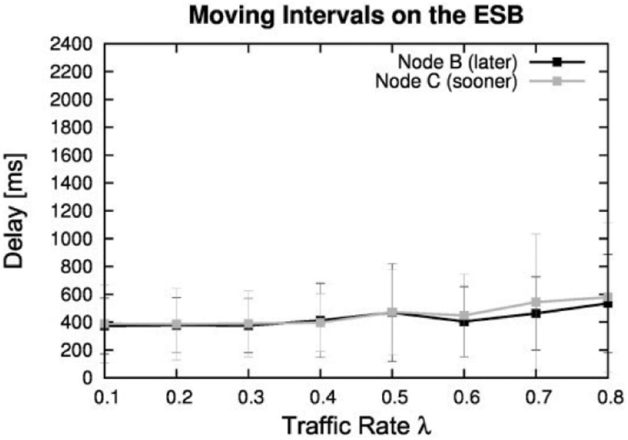

Fig. 4b Delay with moving intervals on the ESB

crease but remains independent from the examined considerably low traffic rates.

4 INVESTIGATION ON TRAFFIC ADAPTIVITY / THROUGHPUT INCREASE

WiseMAC is very energy-efficient for low traffic rates. Performance evaluations show that the energy consumption increases linearly with the traffic rate. To increase the maximum throughput in case of packet bursts and higher traffic load, WiseMAC suggests an optional fragmentation scheme called *more bit* mode. WiseMAC sets a flag (more bit) in a unicast MAC frame whenever a node has *more packets* to send. The more bit in the frame header signals to the receiving node that it shall not turn off the transceiver after receiving the frame, but switch to the receive mode again after frame acknowledgement in order to receive the next packet. However, even when using the more bit, the throughput of WiseMAC is rather limited and packet loss occurs with rather low traffic rates already. The reason is that in tree-based wireless sensor network scenarios, nodes receiving traffic from several sources might become bottlenecks that have to forward data of large subtrees.

We have developed a mechanism to improve the traffic-adaptivity of the WiseMAC protocol in cases of traffic between multiple senders and one receiver basing on a so-called *stay-awake promise*. Cases with multiple nodes aiming to forward data over certain receivers are likely to occur in wireless sensor network topologies. Nodes shall automatically stay awake for a certain time when more traffic has to be handled and tell this to all nodes waiting to forward traffic to it in the acknowledgement.

Fig. 5a and 5b depict the original WiseMAC more bit and our stay-awake promise scheme in a scenario where two sources SRC1 and SRC2 simultaneously aim to transmit packets to the same node DST. If SRC1 and SRC2 both aim to reach DST in the same wake interval, the contention mechanism will decide who is first. SRC1 wins the contention and sends its first two frames with the more bit set. The destination node acknowledges the more bit in the ACK packet and promises to stay awake for a certain time. As SRC2 has lost the contention, it will wait and overhear the transmission from SRC1 to DST. By overhearing the stay-awake promise in the ACK, SRC2 knows that it can start sending its own data frames right after SRC1 has finished its transmissions. The advantage of this scheme is that no time is wasted with an idle channel. The mechanism is only activated when there is a node buffering more than one frame that requests its destination to stay awake for one next packet, which is an indication of increased load.

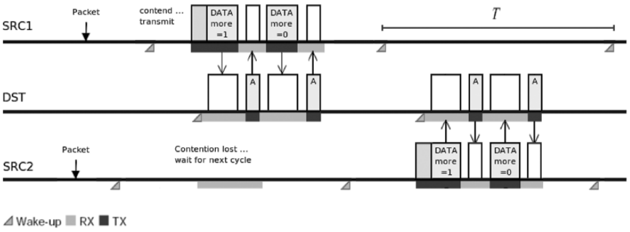

Fig. 5a WiseMAC more bit scheme

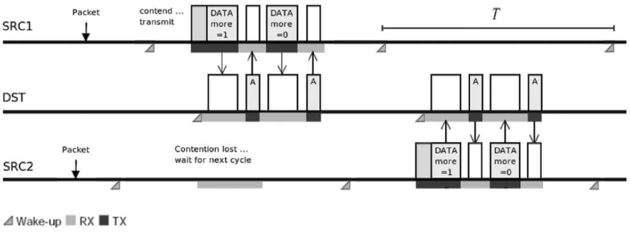

Fig. 5b Stay-Awake Promise Scheme

The scheme shows benefits in terms of throughput, packet loss and delay, which however comes at the cost of slightly increased energy consumption. Fig. 6a depicts the throughput increase in a simulation experiment. It illustrates the throughput increase when applying the same experiment setup as in Section 4. Nodes report data towards the sink the scenario with 90 nodes uniformly distributed across a plane.

In Fig. 6b, we measured the throughput when generating traffic of equal rate from two senders to one receiver. With our proposed stay-awake promise scheme, the receiver node promises to stay awake for at least T = 500 ms and communicates this by setting a bit in the acknowledgement frame. The x-axis constitutes the traffic generated by each of the two nodes.

The results obtained in simulation and on the real sensor test bed confirm that the stay-awake promise scheme is superior to the original WiseMAC more bit scheme in respect to the achieved maximum throughput. The superior performance of roughly 20% has been found similar in both simulation and real-world experiments. Advantages and drawbacks of the scheme are further elaborately analyzed in simulation and on real sensor hardware in [10] and [11].

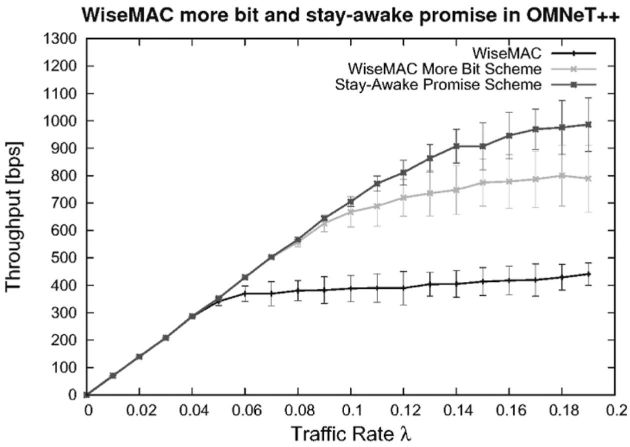

Fig. 6a Comparison of the schemes in OMNeT++

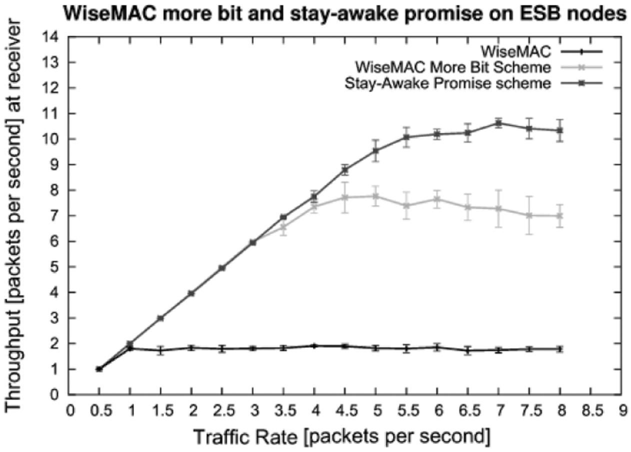

Fig. 6b Comparison of the schemes on the ESB

5 CONCLUSIONS

This paper is an excerpt from the investigations conducted in the context of [1]. We show that in energy-efficient MAC protocols such as WiseMAC, problems may arise when two neighboring nodes share a similar wake patterns. We propose an alternative allocation and arrangement scheme of the node wake-ups to avert performance degrading systematic overhearing and to avert fairness effects of the WiseMAC fixed static wake-up pattern. The paper suggests a scheme to let the node's wake-up intervals move inside fixed cycles with respect to a linear movement-function. This scheme yields a lower danger of systematic overhearing than the WiseMAC fixed wake-up pattern and yet retains the deterministic nature of the wake-ups. The paper shows that the approach notably increases throughput and decreases latency in a simulation study and a real-world experiment.

The paper further investigates mechanisms to improve the traffic-adaptivity of unsynchronized energy-efficient MAC protocols. A novel scheme to improve the traffic-adaptivity and throughput in scenarios with multiple senders and bottleneck destination nodes is proposed. The results obtained in simulation and sensor test bed confirm that the so-called stay-awake promise performs better than the original WiseMAC more bit scheme. The superior performance of 20% has been found similar in both simulation and real-world experiments.

LITERATURE

[1] Philipp Hurni: Unsynchronized Energy-Efficient Medium Access Control and Routing in Wireless Sensor Networks, Master thesis, University of Bern, Switzerland, Nov. 2007.

[2] El-Hoiydi, A.; Decotignie, J.-D.: WiseMAC: An Ultra Low Power MAC Protocol for Multihop Wireless Sensor Networks, ALGOSENSORS, 2004.

[3] El-Hoiydi, A.: Energy Efficient Medium Access Control for Wireless Sensor Networks, PhD Thesis, EPF Lausanne, 2005.

[4] J. Polastre; Hill, J.; Culler, D.: Versatile Low Power Media Access for Wireless Sensor Networks, ACM Conference on Embedded Networked Sensor Systems (SenSys '04).

[5] Buettner, M.; Yee, G.V.; Anderson, E.; Han, R.: X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks, 4th ACM Conference on Embedded Networked Sensor Systems, 2006.

[6] Varga, A: The OMNeT++ Discrete Event Simulation System, European Simulation Multiconference, 2001.

[7] Drytkiewicz, W.; Sroka, S.; Handziski, V.; Koepke, A.; Karl, H.: A mobility framework for omnet++. 3rd Intl. OMNeT++ Workshop, 2003.

[8] RF Monolithics: TR1001 hybrid transceiver, http://www.rfm.com/products/data/TR1001.pdf.

[9] Schiller, J.; Liers, A.; Ritter, H.; Winter, R.; Voigt, T.: ScatterWeb: Low Power Sensor Nodes and Energy Aware Routing, 38th Annual Hawaii International Conference on System Sciences, 2005.

[10] Philipp Hurni; Torsten Braun: Increasing Throughput for WiseMAC, IEEE/IFIP WONS 2008, Garmisch-Partenkirchen, 2008.

[11] Philipp Hurni; Torsten Braun: Evaluation of WiseMAC on Sensor Nodes 10th IFIP International Conference on Mobile and Wireless Communications Networks (MWCN2008), Toulouse, 2008.