Experimental Evaluation of Simulation Abstractions for Wireless Sensor Network MAC Protocols

G.P. Halkes K.G. Langendoen
Faculty of Electrical Engineering, Mathematics and Computer Science
Delft University of Technology, The Netherlands
{g.p.halkes,k.g.langendoen}@tudelft.nl

Abstract— **The evaluation of MAC protocols for Wireless Sensor Networks (WSNs) is often performed through simulation. These simulations necessarily abstract away from reality in many ways. However, the impact of these abstractions on the results of the simulations has received only limited attention. Moreover, many studies on the accuracy of simulation have studied either the physical layer and per link effects or routing protocol effects. To the best of our knowledge, no other work has focused on the study of the simulation abstractions with respect to MAC protocol performance.**

In this paper we present the results of an experimental study of two often used abstractions in the simulation of WSN MAC protocols. We show that a simple SNR-based reception model can provide quite accurate results for metrics commonly used to evaluate MAC protocols. Furthermore we provide an analysis of what the main sources of deviation are and thereby how the simulations can be improved to provide even better results.

I. INTRODUCTION

To evaluate a MAC protocol for a Wireless Sensor Network (WSN) requires that one performs several experiments with different representative topologies. Furthermore, these experiments should ideally be repeated several times to obtain statistically relevant results. Performing these experiments in the real world is exceedingly time consuming and costly. Therefore, MAC protocol designers normally resort to using a simulator to evaluate their protocols. Using a simulator is a cheap and quick way to perform many experiments with different topologies and parameter settings.

Simulators necessarily abstract away from reality in many ways. For example, radio propagation is not simulated by simulating the EM radiation through the air and obstacles from one antenna to the next, but by using a formula to calculate the received signal strength at the receiving radios. It is clear that these abstractions are required to make simulation feasible, and it is likely that many details can be ignored because of their limited impact on the simulation results. However, limited work has been done to validate the abstractions commonly used in simulators for WSN MAC protocols evaluation.

In this paper we study the impact two abstractions commonly used in simulations of WSNs. These abstractions are different ways to model the reception of signals at WSN nodes. First we evaluate the binary reception model that is used in the Unit Disk Graph (UDG) model. In this reception model a signal is either received by a node at sufficient strength that perfect reception is guaranteed, or it is not received at all. Furthermore, all signals have equal strength, so if two signals arrive at the same node at the same time the node will not be able to receive either signal. This model is sometimes extended with an interference range. The signals arriving at nodes within the interference range are assumed not to be decodable, but strong enough to interfere with all other signals and therefore prevent reception.

The second reception model we evaluate is the SNR-based reception model. In this model each signal is given a signal strength. If some signal arriving at a node is stronger than sum of all other signals at the node by at least the SNR ratio the node can properly receive the signal. If the strength of the strongest signal versus the other signals is below the threshold the node will only receive a garbled message. This model is commonly used in combination with Free space and Two Ray propagation models. However, for our evaluation of the SNR abstraction we use measured signal strength from the testbed we use for validation.

We evaluate the accuracy of the physical layer abstractions within the context of MAC protocols for WSNs. Therefore we focus on the performance metrics commonly used in evaluating MAC protocols. These are packet delivery ratio (a.k.a. packet reception rate or goodput), and energy consumption which is usually derived from the time spent in different radio states. Furthermore we investigate the average packet latency. Latency is a less often used metric for the evaluation of MAC protocols as it is usually assumed that latency is not important and can be traded off for better energy consumption.

The rest of this paper is organised as follows: in Section II we give an overview of relevant prior research in the area of simulation validation. In Section III we describe the setup of the experiments we performed, followed by the results in Section IV. Finally, in Section V we present our conclusions.

II. RELATED WORK

Wireless simulation accuracy has been studied mostly in the context of Mobile Ad Hoc Networks (MANETs). Within this context the work of Ivanov et al. [1] and Liu et al. [2] is most similar to our work. Ivanov compares the results of a real experiment with the results of the ns-2 wireless simulator for the same experiment, concluding that the simulated delivery ratio is quite accurate but latency results show much deviation from the real experiment. Furthermore, the study is limited to a single routing and MAC protocol. This limits the value as other studies have shown that different routing protocols are affected differently by different physical models [3].

The work of Liu et al. [2] provides some validation of the physical layer models used in the SWAN simulator, by using connectivity traces from a real experiment to drive the simulator. This study focuses solely on packet delivery ratio and parameter sensitivity, using different routing protocols.

Kotz et al. [4] provides a list of assumptions used by many MANET network simulators and provides a few small experiments to show that these assumptions can lead to erroneous results.

More specific studies into the accuracy of WSN simulators have been performed by Colesanti et al. [5], Lee et al. [6] and Wittenburg et al. [7]. Colesanti studied the OMNeT++ MAC Simulator, but again only looked at packet delivery ratios and a single MAC protocol. The MAC Simulator uses the Unit Disk Graph (UDG) model. Colesanti et al. showed that by introducing probabilistic packet corruption derived from real-world experiments, the results of the UDG model could be made to approach the real-world experiments.

The experiments done by Wittenburg et al. [7] focus on single link behaviour. The results show that given a reasonable propagation model, similar packet loss rates can be achieved as in real-world experiments. Lee et al. [6] provides a new trace-based noise model for wireless simulations. Through several experiments Lee et al. show that their model can simulate single links more accurately with respect to packet delivery ratio than existing models. However, because both studies only consider a single link, effects such as collisions and the capture effect are unknown and no conclusions can be derived with respect to MAC protocol behaviour.

Because in MANETs energy efficiency is not an important metric, none of the MANET studies have considered energy consumption. The study by Colesanti, although focused on WSNs, also did not consider energy consumption. Heidemann at al. [8] have considered energy consumption but only to show that the energy consumed by nodes when waiting for packets to arrive is a significant factor and must be taken into account in simulations.

All MANET validation studies have used the 802.11 MAC protocol. As this is the de facto standard in MANETs, this is perfectly reasonable. However, as we show in this paper, not all MAC protocols are affected equally by the choice of physical layer abstraction. Therefore it is important to specifically study the impact of simulation abstractions on different MAC protocols.

III. EXPERIMENT SETUP

To evaluate the simulation abstractions, we compare simulation results with results from our PowerBench testbed [9]. The testbed consists of 24 nodes installed in our offices. By configuring the send power to its lowest setting we can create a multi-hop network. However, when using this setting we can only usefully employ 22 nodes. The nodes in our testbed are Tnodes, which use the same components as the mica2 nodes (Chipcon CC1000 radio, Atmel ATmega 128L processor).

On our testbed we use our TinyOS 2.x λMAC framework, for which we have implementations of several MAC protocols. The MAC protocols using the λMAC framework represent different points in the MAC protocol design space. For our experiments we use the B-MAC [10], T-MAC [11], Crankshaft [12], and LMAC [13] protocols. The B-MAC protocol is representative of the Low-Power-Listening class of protocols. T-MAC is also a carrier-sense based protocol, but instead uses frames with active and idle periods to reduce energy consumption. LMAC is an example of a TDMA protocol, and finally Crankshaft is a hybrid protocol using the slotted structure of TDMA protocols in combination with carrier sensing to achieve high energy efficiency. It should be noted that the LMAC implementation for the λMAC framework uses a static slot assignment and uses a timer to detect the absence of packets in a slot rather than a carrier sense mechanism.

In order to limit as much as possible the influence of modelling differences between the software running on the real hardware and the simulation models, we have chosen to use the TinyOS 2.x simulator TOSSIM. The standard TOSSIM however does not provide a model for the CC1000 radio. Therefore, we used and modified the PowerTOSSIM extension for TinyOS 2.x as a basis to implement different reception models.

A. *Traffic Pattern and Metrics*

In our evaluation we consider the convergecast or to-sink traffic pattern. In this pattern all nodes in the network send messages to a single sink node. This pattern is representative of data collection in WSNs. Because we are considering a multi-hop network a routing tree needs to be set up. To eliminate the influence of other components than the MAC protocol as much as possible, we use a fixed routing tree that we created off-line based on link quality measurements. The sink was positioned in the middle of the network and the average hop count was approximately 1.86.

As metrics we consider both delivery ratio and energy consumption. The delivery ratio is simply defined as the fraction of messages sent by all nodes that arrive at the sink node. The energy consumption in simulation is usually derived from the time the radio spends in transmit, receive and idle state. In previous work [9] we have shown that using this simple three-state model yields accurate results for energy consumption. In this paper we therefore use the time spent in the different states as our metric, rather than the combined energy consumption number. Using the separate states allows us to more precisely determine the causes of inaccuracy of the simulator.

Finally, we also provide average packet latency as metric. Packet latency is usually traded for energy consumption in WSN MAC protocols, and therefore not considered a very important metric. However, it is interesting to see how much latency is incurred because of the trade off, and this does make latency an interesting metric.

B. *Abstractions*

In this paper we study two reception models commonly used in WSN MAC protocol simulation. The first model is the binary reception model employed by the Unit Disk Graph (UDG) model. In this model nodes either receive a signal perfectly, or not at all.

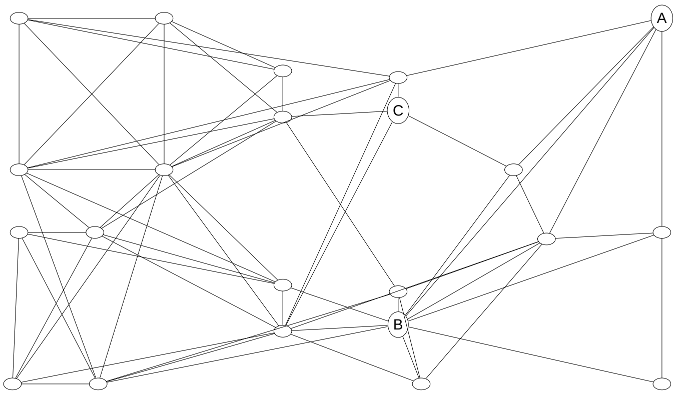

Fig. 1. Connectivity in the PowerBench testbed. Links shown have at least 95% packet reception in both directions.

To arrive at a simulation of the binary reception model which can be compared with the results from our testbed, we cannot simply derive a connectivity graph from the node positions. As Figure 1 shows, the connectivity in our testbed network is very irregular. For example, there is good connectivity between node A and node B, while the link between A and C, which is much shorter, allows virtually no messages to get through. Therefore we first measured all link reception rates in our testbed. From this we extracted the subset of links that show (near) perfect reception, i.e. those links with 95% reception. These links are then taken to be usable for signal transmission, while all other links are discarded. In our experiments we use the extended reception model that also implements an interference range. The links on which interference can occur are the links for which the signal strength is above a threshold. The threshold has been tuned to provide simulation results as close as possible to the real-world results. We have verified that the extended binary model yields slightly better results than the simple binary model.

The second popular reception model is the Signal to Noise Ratio (SNR) based model. In this model all signals have a different signal strength. To determine whether a signal can be received, it is compared to the sum of all other signals and noise. If the signal is stronger than the combined signal by at least the SNR threshold it is assumed to be decodable. Collisions therefore only occur when two signals arrive that are close in signal strength.

As with the binary reception model, we cannot simply use the positions of the nodes in our testbed to calculate signal strengths as there is too little correlation between received signal strength and distance. This phenomena has already been shown in [4]. We have therefore measured the (average) signal strengths between all pairs of nodes in our testbed. We use this information as the signal strengths in our simulations. Furthermore, we have experimentally determined the SNR value of the CC1000 radio on the Tnodes to be approximately 5 dB.

IV. RESULTS

We now present the results of our experiments. All experiments were repeated between 5 and 8 times, and the graphs show the mean and standard deviation.

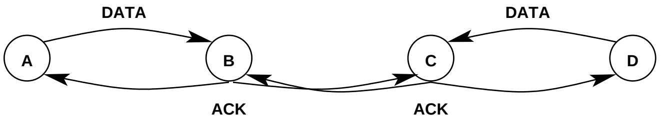

Fig. 2. Potential collision situation for the Crankshaft protocol.

A. Delivery Ratio

Figure 3 shows the delivery ratio for the different protocols. From the graphs it is immediately clear that the binary reception model does not provide a good simulation abstraction. For all protocols except LMAC the simulated delivery ratio is much worse than the measured delivery ratio. It is not surprising that the delivery ratio in LMAC is not affected by the binary reception model, as transmissions in LMAC are scheduled not to collide.

The large differences in delivery ratio for the B-MAC, T-MAC and Crankshaft protocols is due to the all or nothing nature of the binary reception model. In real life fewer collisions occur because weak signals do not interfere with strong signals. In the binary model there is no distinction between weak and strong signals, which means all concurrent transmissions arriving at a single node will always cause a collision.

The delivery ratio for the SNR-based simulation for the most part approach the measured results quite closely. Notable exceptions are the B-MAC protocol at high message rates and the Crankshaft protocol at low message rates. The reason that B-MAC diverges at high message rates is because the real implementation detects more carriers than the simulator. Even though both use the same code for carrier detection, the fluctuations in (measured) signal strength that occur in real life are not simulated and therefore fewer carriers are detected when the signal strength is close to the detection threshold. We found that this abstraction is the cause of most differences between the measured results and the SNR-based simulation model.

The cause of the difference between the SNR-based simulations of the Crankshaft protocol and the measured results is very different. The sending of messages in the Crankshaft protocol is synchronised to a particular time in each slot. When equal length messages are sent as shown in Figure 2 both the actual messages and the acknowledgements are received error free. However, when node A is slightly ahead of node D, the acknowledgement sent by node B in response to A's message will collide with the message from D to C. In simulation the clocks of different nodes run exactly at the same rate. Therefore, once properly synchronised there is no chance of such a collision. However, in reality clocks on different nodes drift, which means collisions of this type are likely to occur. At higher message rates the extra carrier detections found in the real-life implementation prevent some hidden terminal problems, which compensates for the synchronisation problem.

B. Energy Consumption

Next we consider the time spent in the different radio states. We do not show the time spent in transmit mode because the

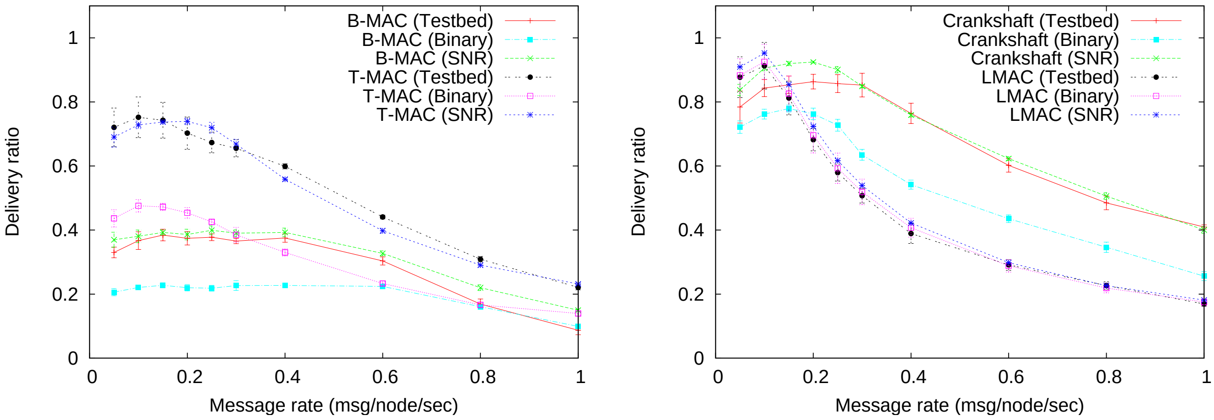

Fig. 3. Delivery ratio for simulated and real convergecast experiment.

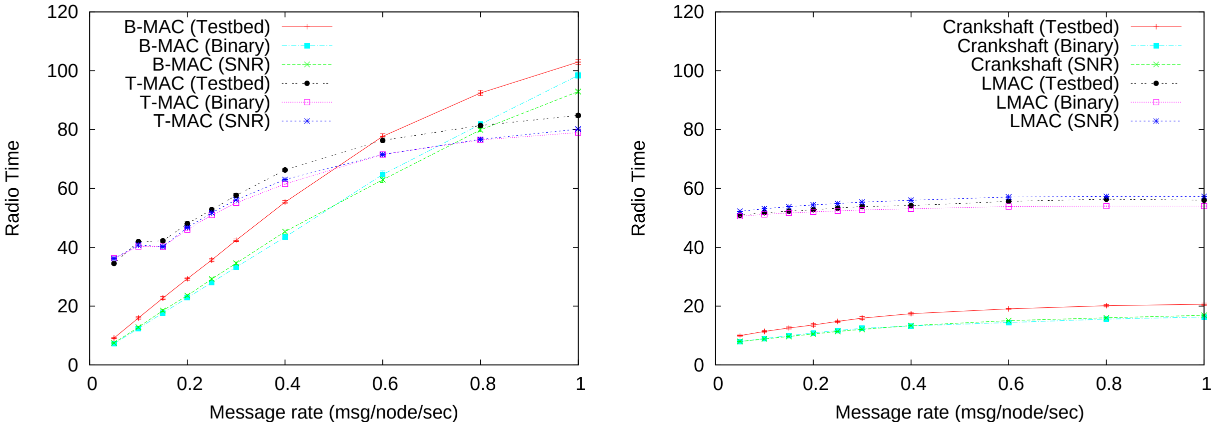

Fig. 4. Radio receive times for simulated and real convergecast experiment.

deviations in transmit time are very small. Figure 4 shows the time spent in receive mode. For this metric the difference between the binary reception model and the SNR-based model are very small. Although this may at first seem contradictory given the delivery ratio graphs, one should take into account that collisions do cause the radio to remain in receive state.

Although the time spent in receive mode for the simulated experiment are very similar to the real-life situation, there are some differences that warrant explanation. The time the real B-MAC and T-MAC spend in receive mode is higher than in the respective simulations. This is again caused by the extra carrier detections. A similar explanation holds for Crankshaft. However, the difference here is that the offset between the real and simulated protocol is much more constant. Because Crankshaft separates unicast and broadcast traffic, the time synchronisation protocol will not consider the unicast packets as suitable for time synchronisation. Therefore, even when there is unicast traffic being sent, the time synchronisation protocol will still send broadcast packets at regular intervals. This, combined with the reduced overhearing for unicast messages in the Crankshaft protocol, results in an almost constant offset as the extra receive time is nearly completely caused by the broadcast messages.

C. Latency

Finally, Figure 5 shows the average packet latency. These graphs do not show the binary model anymore, because the divergence in packet delivery is too large to make these results sensible.

For T-MAC the SNR-based latency results seem quite similar to the real-world results, except at low message rates. However, there are two effects here that cancel each other out at higher data rates. The already mentioned extra carrier detections on the one hand cause extra latency in the real-world experiment. On the other hand they also prevent hidden terminal collisions around the sink, which effectively reduces latency. At lower data rates hidden terminal collisions are less of a problem. Therefore the latency in the real-world experiment is higher at low data rates.

B-MAC suffers from using long preambles. The real-world latency is much higher because each extra carrier that is detected defers the sending of a packet by a significant amount of time.

The difference between the real-world Crankshaft results and the SNR-based Crankshaft results is a direct consequence of the difference in the source of packet loss and retries. The packets lost in the real-world experiment are lost throughout the network. However, the SNR-based simulation experiences packet loss mostly around the sink. Because in Crankshaft the sink is assumed to be mains powered and therefore listens in every slot, a retry to the sink can be done in the next slot. However, a packet lost elsewhere in the network has to be retried in the next frame, and therefore has to be deferred for a lot longer.

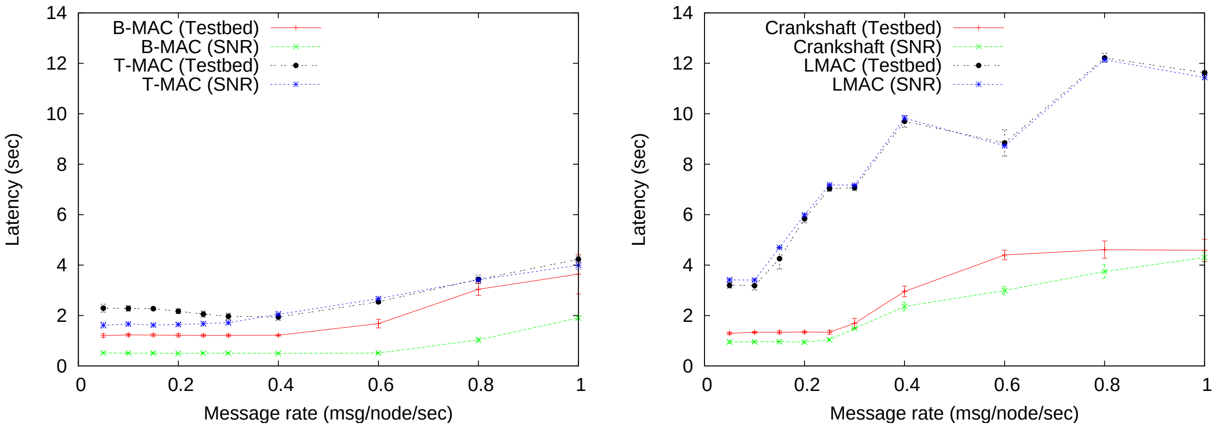

Fig. 5. Average packet latency for simulated and real convergecast experiment.

D. *Other Experiments*

Besides the experiment described in detail in the previous sections we have performed two more experiments. Due to lack of space we can not present an in depth analysis of these experiments, so we only present the most important findings.

First we performed another experiment using the convergecast traffic pattern, but using a different routing tree. For this experiment we used a node at the edge of the network as the sink node. With the sink at the edge we created a routing tree with an average hopcount of 2.48. In this routing topology the hidden terminal problem is much less of an issue. The results of this experiment are mostly the same as the experiment shown, except that the reduced hidden terminal collisions do not compensate other differences as much.

Finally we also performed an experiment with the broadcast flood pattern. In this pattern one node (usually the sink) injects a message into the network that must be sent to all other nodes in the network. The broadcast flood pattern is often used to disseminate data through the network. The SNR-based simulation results were within a few percent of the real-world experiments, except for B-MAC. For B-MAC the delivery ratio was matched for low traffic rates, but at high traffic rates the SNR-based simulations showed higher delivery ratio than in the real world. This can again be explained by extra carrier detections in the real world, which reduce the maximum available bandwidth.

V. CONCLUSIONS

In this paper we have presented the results of an experimental study of two often used reception models in the simulation of WSN MAC protocols. We have compared simulations that use these models with data from a testbed. The results show that the binary reception model used in, for example, the Unit Disk Graph simulation model results in significant deviations from the real-world results. The SNR-based model, however, can provide quite accurate results for metrics commonly used to evaluate MAC protocols, i.e. delivery ratio and energy consumption. Most results are within 5%, while virtually all are within 15%. For latency the results are less accurate. The main cause of remaining deviation between real-world results and the SNR-based model are the unmodeled fluctuations in (measured) signal strength.

REFERENCES

[1] S. Ivanov, A. Herms, and G. Lukas, “Experimental validation of the ns-2 wireless model using simulation, emulation, and real network,” in *4th Workshop on Mobile Ad-Hoc Networks (WMAN 2007)*, Berne, Switzerland, Mar. 2007.

[2] J. Liu, Y. Yuan, D. M. Nicol, R. S. Gray, C. C. Newport, D. Kotz, and L. F. Perrone, “Empirical validation of wireless models in simulations of ad hoc routing protocols,” *Simulation*, vol. 81, no. 4, pp. 307–323, Apr. 2005.

[3] M. Takai, J. Martin, and R. Bagrodia, “Effects of wireless physical layer modeling in mobile ad hoc networks,” in *2nd ACM Int. Symp. on Mobile Ad Hoc Networking & Computing (MobiHoc 2001*. ACM Press, Oct. 2001, pp. 87–94.

[4] D. Kotz, C. Newport, R. S. Gray, J. Liu, Y. Yuan, and C. Elliott, “Experimental evaluation of wireless simulation assumptions,” in *ACM/IEEE International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, October 2004, pp. 78–82.

[5] U. M. Colesanti, C. Crociani, and A. Vitaletti, “On the accuracy of OMNeT++ in the wireless sensor networks domain: Simulation vs. testbed,” in *4th ACM Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PE-WASUN 2007)*, Crete Island, Greece, Oct. 2007, pp. 25–31.

[6] H. Lee, A. Cerpa, and P. Levis, “Improving wireless simulation through noise modeling,” in *6th Int. Conf. on Information Processing in Sensor Networks (IPSN07)*, Cambridge, Massachusetts, Apr. 2007, pp. 21–30.

[7] G. Wittenburg and J. Schiller, “A quantitative evaluation of the simulation accuracy of wireless sensor networks,” in *6. Fachgesprch “Drahtlose Sensornetze” der GI/ITG-Fachgruppe “Kommunikation und Verteilte Systeme”*, Aachen, Germany, July 2007, pp. 23–26.

[8] J. Heidemann, N. Bulusu, J. Elson, C. Intanagonwiwat, K. chan Lan, Y. Xu, W. Ye, D. Estrin, and R. Govindan, “Effects of detail in wireless network simulation,” in *SCS Multiconference on Distributed Simulation*, Phoenix, Arizona, Jan. 2001, pp. 3–11.

[9] I. Haratcherev, G. Halkes, T. Parker, O. Visser, and K. Langendoen, “PowerBench: A scalable testbed infrastructure for benchmarking power consumption,” in *Int. Workshop on Sensor Network Engineering (IWSNE)*, Santorini Island, Greece, June 2008, pp. 37–44.

[10] J. Polastre and D. Culler, “B-MAC: An adaptive CSMA layer for low-power operation,” UC Berkeley, Tech. Rep. cs294-f03/bmac, Dec. 2003.

[11] T. van Dam and K. Langendoen, “An adaptive energy-efficient MAC protocol for wireless sensor networks,” in *1st ACM Conf. on Embedded Networked Sensor Systems (SenSys 2003)*, Los Angeles, CA, USA, Nov. 2003, pp. 171–180.

[12] G. Halkes and K. Langendoen, “Crankshaft: An energy-efficient MAC-protocol for dense wireless sensor networks,” in *4th European conference on Wireless Sensor Networks (EWSN’07)*, Delft, The Netherlands, Jan. 2007, pp. 228–244.

[13] L. van Hoesel and P. Havinga, “A lightweight medium access protocol (LMAC) for wireless sensor networks,” in *1st Int. Workshop on Networked Sensing Systems (INSS 2004)*, Tokyo, Japan, June 2004.