An Energy Model for Simulation Studies of Wireless Sensor Networks using OMNeT++

Feng Chen, Isabel Dietrich, Reinhard German and Falko Dressler
Computer Networks and Communication Systems
University of Erlangen, Germany
{feng.chen,isabel.dietrich,german,dressler}@informatik.uni-erlangen.de

ABSTRACT

Simulation is frequently used to evaluate the performance of networking algorithms and techniques in wireless communication networks. However, performance aspects such as the transmission delay, the channel utilization, or the throughput provide only limited information about the feasibility of the particular approach. This is especially the case when investigating Wireless Sensor Networks (WSNs), which stress energy-efficiency due to extremely limited power source in sensor nodes. Thus, a precise evaluation of the energy efficient performance is demanded. In this paper, we present a generic energy model developed for the simulation framework OMNeT++. The model allows to accurately evaluate the energy performance (in terms of energy consumption or network lifetime) of sensor networks (or in principle any wireless network), taking into account the energy consumption of both the radio transceiver and the CPU. The energy model can be calibrated to arbitrary types of sensor nodes if power measurements are available for the node type. The applicability of the developed energy model is demonstrated in a selected application scenario: the analysis of the energy consumption in IEEE 802.15.4 star networks. Additionally, we outline the need for more complex metrics compared to the energy consumption of single nodes to evaluate the lifetime of the whole sensor network.

1. INTRODUCTION

In the last decade, significant advances have been achieved in the domain of Wireless Sensor Networks (WSNs).

Some of the most challenging issues that have been studied are the medium access, routing strategies, clustering schemes, and application layer dynamics. All these approaches contribute to the final objective to enable designers to develop and to deploy applications under various environmental conditions. The idea is to provide a broad range of design variants that can be chosen and combined in order to provide the optimal behavior of the sensor network. Most of the approaches are targeted to improve the performance of the wireless communication with respect to the quality of service. Therefore, all the individual algorithms and techniques have been analyzed with regard to their performance, e.g. the speed of adaptation to environmental changes, the end-to-end performance, and the resulting overhead.

A key objective for most wireless communication networks and especially for sensor networks is to reduce the energy consumption. Since the radio transceiver is the most critical part contributing to the energy consumption on an individual sensor node, much effort has been put into designing proper network protocols that turn the radio off as long as possible while still maintaining network connectivity. Such energy preserving techniques have been developed in all the layers of the communication stack. For example, to reduce energy waste in idle listening and overhearing, MAC protocols for WSNs have largely adopted schedule-based techniques with fixed or varying duty-cycles. Thus, all nodes periodically turn the radio on and off and the problem of saving energy is being reduced to finding a perfect scheduling scheme [15, 17].

Another example is the Low Power Listening (LPL) approach [12], which has emerged as a physical layer technique that utilizes preamble sampling to minimize idle listening. LPL has recently attracted considerable research interest [1, 9, 11]. Consequently, this technique also receives increasing hardware support and a number of off-the-shelf radio products are being developed for application in WSNs.

The evaluation and validation of the developed energy-efficient solutions is usually relying on either experimentation or simulation techniques. Experimentation allows to study WSNs in a real-world environment which provides accurate measurements, especially for energy consumption, for the hardware equipment used during the experiment. However, simulation also plays a major role in most performance studies due to a number of reasons. First, simulation techniques allow to study novel methods and techniques without the need of real deployments, which may be infeasible due to hardware limitations or just impractical due to the necessary implementation efforts. Furthermore, evaluating large networks with hundreds or even thousands of sensor nodes can typically only be done by means of simulation. Experimentation might be too expensive for such setups or infeasible due to physical limitations. Thus, in most cases performance evaluation is based on simulation models.

Basically, two levels for sensor network simulations can be distinguished: the single-node level and the network level. Simulators that work at the level of single sensor nodes are, for example, PowerTOSSIM [13] and Avrora [14]. They can generate detailed estimates for CPU cycle usage and energy consumption of sensor nodes. Their main disadvantage is, however, that they work on executable sensor node programs, which might not exist at the early stages of development. This also means that they are usually tailored towards only very few hardware platforms, and are difficult to adapt to other platforms. More complex modeling approaches are also being developed, still focusing on a single sensor node [8].

Simulators at the network level, like OMNeT++ [16], are completely independent from the sensor node hardware. This means that the estimates produced by them depend strongly on the specific resource consumption models used in a simulation. In most simulations, CPU usage is not considered at all. Besides the classical network performance metrics like throughput and delay, the energy performance of WSNs has recently been evaluated in many simulation studies – primarily using the *network lifetime* as a metric [5].

We have observed a number of problems in most of these simulation experiments. Firstly, many energy models have not been built accurately to reflect real operations in the radio hardware. For example, some energy models assume that the radio consumes the same power in idle listening as in receiving state and they are ignoring the energy consumption in sleeping state. Furthermore, few simulation models take into account the transition energy cost for switching between the radio operational states. Secondly, the energy models deployed in many simulation studies have not been calibrated using measurement results obtained from experiments on real hardware platforms. Finally, the energy consumption of the micro controller is frequently not considered.

Aiming to overcome the described drawbacks in existing energy models, we contribute to the current research by presenting a more accurate energy model for the OMNeT++ simulation framework [16]. Our energy model distinguishes the different power consumption rates in each radio state and it explicitly considers the necessary transition energy. A simple CPU model has been built to estimate the energy consumption for computationally intensive operations. Benefiting from published measurement results from hardware experiments, our energy model can be easily calibrated to a number of radio transceivers and micro controllers.

The remainder of the paper is organized as follows. Section 2 introduces the energy model that we developed for OMNeT++. In Section 3, we demonstrate its applicability in a case study for IEEE 802.15.4-based star networks. Section 4 outlines the necessary metrics for evaluating the network lifetime based on the energy measurement of single sensor nodes. Finally, Section 5 concludes the paper.

2. ENERGY MODEL IN OMNET++

In this section, we describe the functionalities and implementation details of our energy model in OMNeT++.

2.1 Principles and Operation

We developed the energy model with respect to the operation principles of the OMNeT++ simulation framework. OMNeT++ is a public-source, component-based discrete-event simulation environment. Its INET framework, a set of modules for standard network protocols, provides support for many network protocols such as IEEE 802.11 (WLAN), TCP/IP, and many others. Additionally, mobility models and traffic models are available. Recently, a number of models for ad hoc and sensor network protocols have been integrated including our model of the IEEE 802.15.4 protocol [4].

Figure 2 shows the interior of a typical wireless node in OMNeT++. The energy (or battery) model has been placed next to the protocol stack because the consumption of energy can occur at different places in a node. Communication with the other modules in the node is realized using a notification board mechanism. The notification board is a module provided by the INET framework. The board allows to model cross-layer communication and crosscutting concerns such as energy without breaking the layered architecture.

In that way, the energy model can serve as a plug-in to various models of wireless protocols in OMNeT++. Its parameters are the initial battery energy, the radio power in different working states, and the CPU power. Based on these configurations, the model accounts for energy usage due to radio transmissions and receptions, CPU consumption (based on radio activity), and additional CPU consumptions based on varying sensor programs (changes in sensor programs are received via the notification board). For demonstration purposes, the remaining energy on each node can be displayed in animations during the simulation run.

Depending on the purpose of the study, the energy model can be configured to execute an arbitrary action upon exhaustion of the battery power. Currently, the model can be configured to terminate the simulation in one of the following two cases:

- When the first node exhausts its battery power, or
- when a specified node (e.g. the central node) dies or all nodes in the network die.

For the second case, the dead nodes have to be excluded from network communication. In our model, we implement this by dynamically disconnecting the `radioIn` gate of the dead node from the channel module and reconnecting it to an empty gate. In addition, all other activities including statistics recording in each module of the dead node need to be terminated as well. To achieve this, the event of battery exhaustion is put on the notification board of the dead node, so that all relevant modules of the node can act accordingly. In the following, we introduce the implementation details in the energy model.

2.2 Implementation Details

For the implementation, we have to distinguish between energy modeling for the radio transceiver and for the CPU.

2.2.1 Radio transceiver

The energy model for the radio is built by considering two sources of energy consumption: the steady state energy and the transition energy. The steady state energy is consumed while the radio is working in one of its working states. The transition energy is consumed by the radio during the switching process from one working state to another. Our energy model supports a typical radio transceiver that defines four working states and eight state transitions as listed in Table 1.

Table 1: Radio working states and state transition

Radio working state	State transition
Idle	SLP $\rightleftarrows$ Idle
Sleeping (SLP)	Idle $\rightleftarrows$ Tx
Transmitting (Tx)	Idle $\rightleftarrows$ Rx
Receiving (Rx)	Tx $\rightleftarrows$ Rx

In order to calculate the energy consumed by the radio, the energy model tracks every radio state switch that is controlled in the Physical Layer (PHY) module using the notification board. Upon receiving state switch events from the notification board, the energy model updates the accumulated time for each radio working state and recalculates the current energy consumption. If the measurements of radio transition energy are available for the specific radio, e.g. the measured transition energy of the Chipcon CC2420 radio [6], the corresponding energy for current radio state switch needs to be further subtracted from the remaining battery power. If the battery is completely exhausted after the update, one of the above mentioned actions will be executed.

2.2.2 CPU or micro controller

Estimating and modeling energy consumption on the CPU is more difficult compared to the radio transceiver because the activity of the CPU is complex and depends mainly on the running application. For instance, the CPU will be busy while processing a packet just received by the MAC or while executing some encryption or decryption algorithms. It can also be idle while the radio is busy transmitting or sleeping. Therefore, we consider only a rough approximation by defining two CPU states, active and inactive, in our current implementation. We also assume that the CPU will follow the same sleeping schedule of the radio interface, which means that CPU is inactive only during the radio sleeping period.

2.3 Model Calibration

Finally, the energy model needs to be calibrated to a specific type of radio transceiver and micro controller. Obviously, this step must be based on intensive measurements of the hardware modules. An example for calibrating the energy module in the IEEE 802.15.4 model is presented in Section 3. Furthermore, the degree of detail in the energy model depends on the implemented functions of the protocol stack models and application specific information. For example, if security mechanisms have been implemented, the energy model can be extended to calculate the CPU consumption for execution of security algorithms.

We currently rely on three published energy measurements for sensor nodes (mainly Mica2 motes). Measurement data is available for the energy consumption of two different radio transceivers: the Chipcon CC1000 (the main radio transceiver used in the last generation of sensor nodes) [10] and the Chipcon CC2420 (the radio transceiver in forthcoming IEEE 802.15.4 based ZigBee networks) [6]. Furthermore, measurement results of energy consumption of the Atmel ATMEGA micro controller for typical security algorithms in sensor networks have been presented in [2].

3. ENERGY EVALUATION OF IEEE 802.15.4 STAR NETWORKS

Based on our simulation model of the IEEE 802.15.4 protocols in OMNeT++, we analyzed the (energy) performance of an IEEE 802.15.4-based star network. The IEEE 802.15.4 model including the energy model is described in [3,4]. Some selected simulation results for energy consumption are presented in the following.

3.1 A Short Overview of IEEE 802.15.4 and Simulation Model in OMNeT++

The IEEE 802.15.4 standard [7] defines Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs) that address the need for low-rate, low-power, and low-cost wireless networking. The IEEE 802.15.4 PHY operates in one of three Industrial, Scientific, and Medical (ISM) frequency bands at a maximum data rate of 250 kbps. The standard supports two network topologies, a star and a peer-to-peer topology.

The IEEE 802.15.4 MAC utilizes a superframe structure to operate in the so called *beacon-enabled* mode as shown in Figure 1. Each superframe is bounded by periodically transmitted beacon frames, which allow nodes to associate with and synchronize to their coordinator. There are two parts in the superframe, an active portion and an inactive portion. Nodes may enter a low-power (sleep) mode during the inactive portion. Two MAC attributes, the *macBeaconOrder* (BO) and the *macSuperframeOrder* (SO), determine the length of the beacon interval (BI) and the length of the active portion of the superframe (SD), respectively. The active portion of the superframe is further divided into three parts: the beacon, the Contention Access Period (CAP), and the Contention-Free Period (CFP). In the CAP, all nodes contend for the channel using a slotted Carrier Sense Multiple Access With Collision Avoidance (CSMA/CA) algorithm. In the CFP, a Time Division Multiple Access (TDMA) medium access scheme utilizing Guaranteed Time Slots (GTS) provides guaranteed service for low-latency applications.

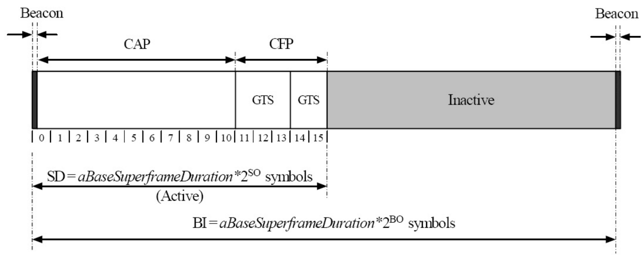

Figure 1: Superframe structure of the MAC layer of IEEE 802.15.4

The implementation in OMNeT++ is outlined in Figure 2. It consists of a PHY and a MAC model plus several supporting models including an Interface Queue (IFQ) and our energy model. For the simulations presented in this paper, we calibrated the energy model based on the measurements for Mica2 motes contributed by Landsiedel et al. [10] and considered no radio transition energy. The deployed power parameters for the radio and the CPU are listed in Table 2 in the form of current draw. In our further ongoing work on evaluation of energy performance of IEEE 802.15.4 based WSNs, we have adapted our energy model to the standard IEEE 802.15.4 radio CC2420 according to its hardware measurement results [6].

3.2 Simulation Scenarios and Results

In a first experiment, we analyzed energy consumption of a star network with one Personal Area Network (PAN) coordinator and 20 devices as shown in Figure 3. Each device sends packets, which are generated with an exponentially distributed interarrival time, to the PAN coordinator. The packet interarrival mean is varied between 0.01 s and 100 s. We fixed SO at each node to 0, which results in a constant active period of 0.015 s. The BO is chosen at 1, 3, 5, and 7, which correspond to beacon intervals of 0.03 s, 0.12 s, 0.49 s, and 1.97 s, respectively.

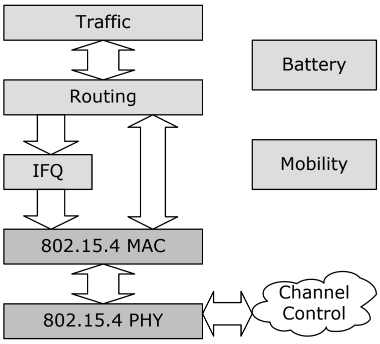

Figure 2: Implemented models of the IEEE 802.15.4 model

Table 2: Energy measurements of Mica2 motes for calibration of energy model in OMNeT++ (data from [10])

Measurements of radio power	
Sleep	60 μA
Idle	1.38 mA
Rx	9.6 mA
Tx (-18 dBm)	8.8 mA
Tx (-13 dBm)	9.8 mA
Tx (-10 dBm)	10.4 mA
Tx (-6 dBm)	11.3 mA
Tx (-2 dBm)	15.6 mA
Tx (0 dBm)	17.0 mA
Tx (+3 dBm)	20.2 mA
Tx (+4 dBm)	22.5 mA
Tx (+5 dBm)	26.9 mA
Measurements of CPU power	
Active	7.6 mA
Inactive	237 μA

The measured mean energy consumption per payload byte is depicted in Figure 4. In the area of heavy traffic load on the left-hand side of the graph, the energy consumption under the same traffic load increases with the increasing length of BI. For example for a traffic interval of 0.01 s, due to the same length of the active period, the average number of packets transmitted per BI are almost the same for various SO. This means that almost the same amount of energy is consumed in the active period. Therefore, the effect that longer BI results in more energy consumption is caused by more energy consumption in the inactive period. Under heavy traffic, the energy consumption of each curve remains constant independent of the traffic load, because the MAC is queuing packets and the channel is completely loaded. Thus, the total energy consumed in the active period reaches its peak value and the number of transmitted packets per BI is saturated. The total energy averaged to each payload byte is constant.

As the traffic load keeps decreasing, the energy curve drops first and then ascends monotonously. The drop in energy consumption is contributed by the decrease in the number of collisions per BI, which reduces the energy wasted in resending. The bottom value of the energy consumption appears when the collision rate and the packet drops at the IFQ are minimized while the channel is still saturated, which also corresponds to the conditions for the maximal throughput of the network.

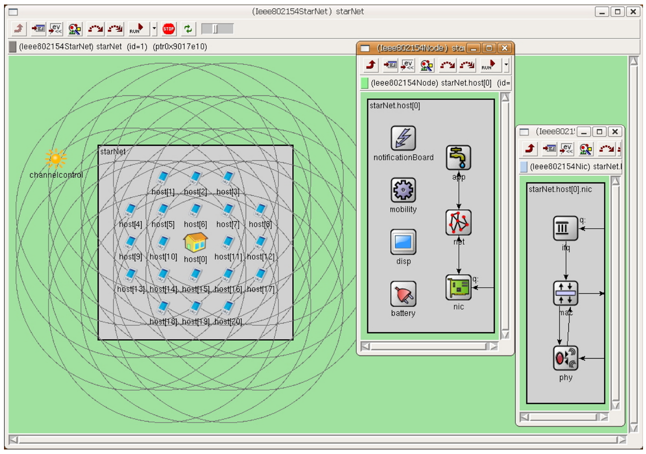

Figure 3: Star network in IEEE 802.15.4 and implemented models in OMNeT++

The increasing trend in energy consumption on the right-hand side of the graph can be explained as follows. As the traffic load gets lighter, fewer packets are transmitted per BI and the portion of energy consumed on idle listening increases. When idle listening starts to contribute the most to the percentage of the overall energy consumption, the mean energy consumption per payload byte will increase inversely proportional to the packet generating rate. In the area of energy ascending on all the curves, the smallest BO consumes the most energy, because with the same SO, higher duty cycle under light traffic means more energy consumption per unit of time. However, the number of packets transmitted per time unit are almost the same.

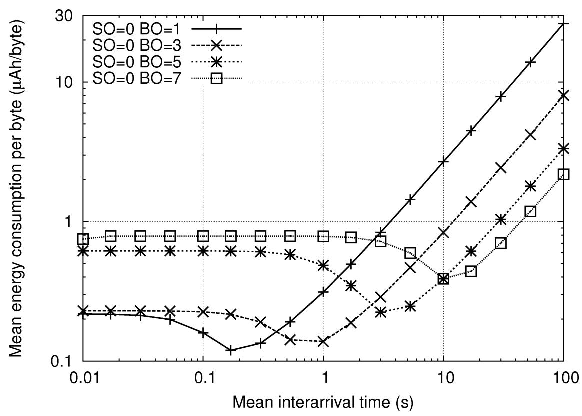

Figure 4: IEEE 802.15.4 CSMA/CA: energy consumption for SO=0 and various values of BO under various traffic loads

In a second experiment, we evaluated the GTS scheme of IEEE 802.15.4, which has been proposed to provide guaranteed service for real-time applications. Considering the deterministic characteristics of the TDMA-like scheme, we investigate the allocation of one GTS in a star network with exactly one device and one PAN coordinator. According to our implementation, a device desiring a GTS can directly request one GTS from the PAN coordinator at the starting stage of the simulation. In our current GTS model, neither the allocation nor the deallocation process is modeled as defined in the specification.

In this experiment, the length of the GTS is allocated with the minimum number of superframe slots that can accommodate at least one complete transaction for transmitting one alarm message with a payload of one Byte. Because no packet loss due to collisions will occur within the GTS in our current model implementing only an ideal channel, the GTS model is configured to run in the none-acknowledgment mode. However, such settings are not reliable for practical applications with existence of interference on lossy channel. We are currently working on an improved channel model that takes into accounts more complicated fading effects in real industrial environments. Using the same (BO,SO) combinations as in Figure 4, we fixed SO to 0 and explored various BO settings at 1, 3, 5 and 7. The measured energy consumption is depicted in Figure 5. The energy curves for GTS follow a similar trend as shown in Figure 4 without producing the described "wave" because in the collision-free GTS no retransmissions will occur and consume energy.

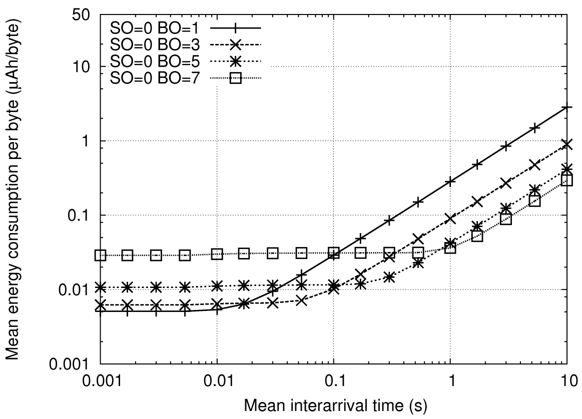

Figure 5: IEEE 802.15.4 GTS: energy consumption for SO=0 and various values of BO under various traffic loads

4. NETWORK LIFETIME

As our application examples in the previous section show, our energy model enables us to produce estimates of the energy consumption of single sensor nodes. This also means that we are able to predict how long a single sensor node can provide its services to the network before it finally fails. However, what our model does not allow is to estimate how long the entire network will be able to provide its services to the user. In other words, our model can predict the lifetimes of single nodes, but not the lifetime of the network as a whole.

A recent literature survey presented in [5] shows that researchers have used a wide range of different definitions of *network lifetime*. The common notion is that the lifetime of a network corresponds to the period of time during which it is able to provide its services to the user. As this definition is informal and therefore too vague to calculate the actual value of network lifetime, many researchers have "invented" their own definitions.

The most common definition found in the literature is to calculate network lifetime as the time when the first node fails. This metric is often too pessimistic because most networks will be able to provide useful services a long time after the failure of the first node. Another common definition takes this into account and allows for the failure of a certain percentage of nodes. A drawback of both of these definitions is that they consider only node numbers, and not the real service provided by the nodes. The most important services delivered by a sensor network are sensing and transmission of data to the user. Therefore, a number of authors include the percentage of covered area (as a measure for the sensing quality) and the connectivity to a base station (as a measure for the possibility of data transmission) in their lifetime definitions.

In effect, this led to a myriad of network lifetime definitions, basically resulting in incomparable performance studies due to different metrics used in each study. To overcome this problem and to provide a way in which comparable and reproducible results concerning network lifetime can be published, we introduced a new metric for sensor network lifetime in [5]. It combines the existing approaches, e.g. percentage of nodes, coverage, connectivity, and some others, and introduces a few new concepts:

- *Service disruption* indicates that a network does not need to deliver its service continuously, but that service disruptions can be tolerated to some degree.
- The idea behind *connected coverage* is that the nodes providing sensing coverage and the nodes providing connectivity in a network may be different.
- *Time-integration* indicates that a criterion does not need to be fulfilled at every point in time, but that it is sufficient if it is fulfilled at least once in each time interval.
- *Graceful degradation* provides a measure of the quality of service delivered by the network after its failure.

There are parameters for each of the existing criteria, and also for the four new ones. These parameters may be used to adjust each criterion's influence on network lifetime to the needs of different applications. Based on the parameter settings, it is possible to evaluate the liveliness of the network at every point in time. The network is considered lively if it fulfills all criteria according to their parameters.

Two different metrics for the network lifetime are distinguished, both giving the network lifetime in seconds. The first metric, accumulated network lifetime, gives the sum of all periods during which the network provided its service, i.e. the sum of all periods during which the network was lively. The second metric, the total network lifetime, gives the first point in time when the liveliness of the network is lost for a longer period than the service disruption tolerance.

We are currently working on a simulation model for network lifetime. In combination with the energy model presented here, we will then be able to accurately estimate the lifetime of a sensor network under various conditions. In particular, this means that we will be able to compare the influence of different protocols or protocol variations on network lifetime.

5. CONCLUSION AND FUTURE WORK

We developed a new energy model for use within the OMNeT++ simulation framework. This model allows to evaluate the energy performance for arbitrary sensor network applications. The energy model provides accurate energy measurements by considering both energy consumption of the radio transceiver for communication and of the CPU for computational effort. Based on a four-state radio model, the energy consumption of the radio transceiver both in working states and during state transitions can be measured by our model. In the current implementation, we define only two working states, active and inactive, in the CPU model and assume that the CPU has the same sleeping schedule as the radio interface. In order to demonstrate the applicability of our work, we presented a simulation study of IEEE 802.15.4-based star networks, in which the energy model was calibrated based on measurements for Mica2 sensor motes. The simulation results for the investigated energy metric are analyzed to reveal the effect of the various traffic loads and the different settings of the protocol parameters on the energy performance.

In addition, we argue that evaluating the energy performance of single sensor nodes is not enough to assess the performance of a complete network. We introduced an ongoing study of network lifetime in WSNs which presents two new metrics for network lifetime. In this context, our energy model may be used as a basis for evaluating the lifetime of WSNs. Future work include fine granular CPU state modeling and support for a wider range of sensor systems.

6. REFERENCES

[1] M. Buettner, G. V. Yee, E. Anderson, and R. Han. X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. In *4th ACM Conference on Embedded Networked Sensor Systems (ACM SenSys 2006)*, pages 307–320, New York, NY, 2006. ACM.

[2] C.-C. Chang, D. J. Nagel, and S. Muftic. Assessment of Energy Consumption in Wireless Sensor Networks: A Case Study for Security Algorithms. In *4th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (IEEE MASS 2007): 3rd IEEE International Workshop on Wireless and Sensor Networks Security (IEEE WSNS 2007)*, Pisa, Italy, October 2007.

[3] F. Chen and F. Dressler. A Simulation Model of IEEE 802.15.4 in OMNeT++. In *6. GI/ITG KuVS Fachgespräch Drahtlose Sensornetze, Poster Session*, pages 35–38, Aachen, Germany, July 2007.

[4] F. Chen, N. Wang, R. German, and F. Dressler. Performance Evaluation of IEEE 802.15.4 LR-WPAN for Industrial Applications. In *5th IEEE/IFIP Conference on Wireless On demand Network Systems and Services (IEEE/IFIP WONS 2008)*, pages 89–96, Garmisch-Partenkirchen, Germany, January 2008. IEEE.

[5] I. Dietrich and F. Dressler. On the Lifetime of Wireless Sensor Networks. *ACM Transactions on Sensor Networks (TOSN)*, 5(1):1–39, February 2009.

[6] I. Howitt, R. Neto, J. Wang, and J. Conrad. Extended energy model for the low rate WPAN. In *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference*, volume 0, page 322. IEEE Computer Society, 2005.

[7] IEEE. Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs). IEEE Sandard 802.15.4-2006, 2006.

[8] D. Jung, T. Teixeira, and A. Savvides. Sensor Node Lifetime Analysis: Models and Tools. *ACM Transactions on Sensor Networks (TOSN)*, 5(1):1–33, February 2009.

[9] R. Jurdak, P. Baldi, and C. Lopes. Adaptive Low Power Listening for Wireless Sensor Networks. *IEEE Transactions on Mobile Computing*, 6(8):988–1004, 2007.

[10] O. Landsiedel, K. Wehrle, and S. Götz. Accurate Prediction of Power Consumption in Sensor Networks. In *Second IEEE Workshop on Embedded Networked Sensors (EmNetS-II)*, Sydney, Australia, May 2005.

[11] S. Moon, T. Kim, and H. Cha. Enabling Low Power Listening on IEEE 802.15.4-Based Sensor Nodes. In *IEEE Wireless Communications and Networking Conference (WCNC 2007)*, pages 2305–2310, March 2007.

[12] J. Polastre, J. Hill, and D. Culler. Versatile low power media access for wireless sensor networks. In *2nd ACM Conference on Embedded Networked Sensor Systems (ACM SenSys 2004)*, pages 95–107, New York, NY, 2004. ACM.

[13] V. Shnayder, M. Hempstead, B. Chen, G. W. Allen, and M. Welsh. Simulating the Power Consumption of LargeScale Sensor Network Applications. In *2nd ACM Conference on Embedded Networked Sensor Systems (ACM SenSys 2004)*, pages 188–200, Baltimore, MD, November 2004.

[14] B. L. Titzer, D. K. Lee, and J. Palsberg. Avrora: scalable sensor network simulation with precise timing. pages 477–482, 2005.

[15] T. van Dam and K. Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. In *1st ACM Conference on Embedded Networked Sensor Systems (ACM SenSys 2003)*, pages 171–180, New York, NY, 2003. ACM.

[16] A. Varga. The OMNeT++ Discrete Event Simulation System. In *European Simulation Multiconference (ESM 2001)*, Prague, Czech Republic, 2001.

[17] W. Ye, J. Heidemann, and D. Estrin. Medium Access Control with Coordinated Adaptive Sleeping for Wireless Sensor Networks. *IEEE/ACM Transactions on Networking (TON)*, 12(3):493–506, June 2004.