Modeling energy consumption of wireless communications in OMNeT++

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

ABSTRACT

We present an energy model for the simulation framework OMNeT++ that has been calibrated using several measurements for real sensor nodes. This energy model allows to study sensor network algorithms and applications in a simulation model with high quality energy estimations. The energy model can be adapted for arbitrary sensor nodes based on respective measurement data. We show the applicability of the energy model based on two scenarios: the analysis of the energy performance of IEEE 802.15.4 and the evaluation of the advantages of on-demand sensor network reprogramming.

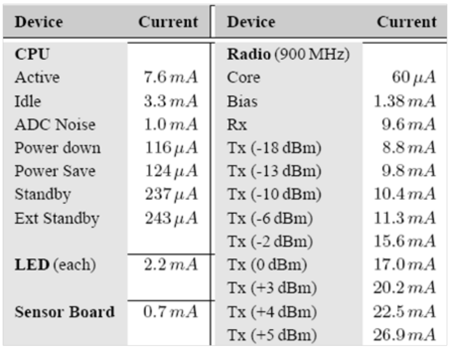
1. INTRODUCTION

In the last decade, many approaches have been proposed that improve the performance of sensor networks. 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.

All the individual algorithms and techniques have been analyzed regarding their performance, e.g. the speed of adaptation to environmental changes, the end-to-end performance, the produced overhead, and the energy consumption. Studying especially solutions optimizing the energy performance – or the entire network lifetime [3] –, we realized that varying techniques are used for evaluation and analysis of the developed solution.

Basically, we need to distinguish between experimentation and simulation as evaluation techniques. In general, it seems that in most cases, only one of these techniques has been used. This has a number of drawbacks that will become obvious in a short comparison. Simulation allows to study developed methods and techniques without the need of really deploying sensor nodes – that may not yet exist. In many evaluated scenarios, experimentation will be too expensive or not possible at all, e.g. for deployment scenarios in hazardous environments. Also, simulation allows the evaluation of really large networks, which will be infeasible in a lab. On the other hand, experimentation allows to study sensor networks in a real world environment facing typical radio transmission problems and others. Therefore, performance evaluation is usually based on simulation models. Nevertheless, measurements are necessary to calibrate simulation models.

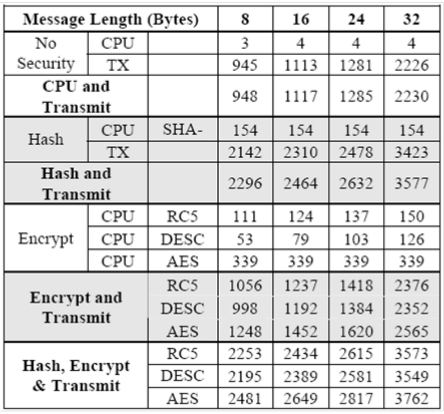
In the last years, a number of research groups started to provide basis measures to be used to improve the quality of sensor network simulation. This includes performance measures of the typical micro controllers, wireless transmission, and energy measures. Examples are the measurements for Mica2 sensor nodes by Landsiedel et al. [6] as depicted in Figure 1 and the analysis of various communication energy measures for normal and encrypted communication in a security-enhanced scenario by Chang et al. [1] as shown in Figure 2.



Device	Current	Device	Current
CPU		**Radio** (900 MHz)	
Active	$7.6\,mA$	Core	$60\,\mu A$
Idle	$3.3\,mA$	Bias	$1.38\,mA$
ADC Noise	$1.0\,mA$	Rx	$9.6\,mA$
Power down	$116\,\mu A$	Tx (-18 dBm)	$8.8\,mA$
Power Save	$124\,\mu A$	Tx (-13 dBm)	$9.8\,mA$
Standby	$237\,\mu A$	Tx (-10 dBm)	$10.4\,mA$
Ext Standby	$243\,\mu A$	Tx (-6 dBm)	$11.3\,mA$
		Tx (-2 dBm)	$15.6\,mA$
LED (each)	$2.2\,mA$	Tx (0 dBm)	$17.0\,mA$
		Tx (+3 dBm)	$20.2\,mA$
Sensor Board	$0.7\,mA$	Tx (+4 dBm)	$22.5\,mA$
		Tx (+5 dBm)	$26.9\,mA$

Figure 1: Energy consumption of Mica2 sensor nodes [6]

In general, energy consumption of sensor networks has been studied manifold. One of the most important observations was that there is a strong contrast between energy consumption for communication and computation. Depending on the source in the literature a factor of 1.000 up to 100.000 needs to be considered. In many current simulation models, only sending activities are counted. Nevertheless, depending on the used duty-cycle and the message rate, message reception and idle listening can be even more expensive than sending. Thus, we need better models for energy consump-



Message Length (Bytes)			8	16	24	32
No Security	CPU		3	4	4	4
	TX		945	1113	1281	2226
CPU and Transmit			948	1117	1285	2230
Hash	CPU	SHA-	154	154	154	154
	TX		2142	2310	2478	3423
Hash and Transmit			2296	2464	2632	3577
Encrypt	CPU	RC5	111	124	137	150
	CPU	DESC	53	79	103	126
	CPU	AES	339	339	339	339
Encrypt and Transmit		RC5	1056	1237	1418	2376
		DESC	998	1192	1384	2352
		AES	1248	1452	1620	2565
Hash, Encrypt & Transmit		RC5	2253	2434	2615	3573
		DESC	2195	2389	2581	3549
		AES	2481	2649	2817	3762

Figure 2: Energy consumption of normal and encrypted communication [1]

tion in our current simulation tools.

In this paper, we contribute to the current research by presenting an energy model for the OMNeT++ simulation framework [7]. We calibrated the model using reference measures provided by other groups as described below. We also show two application scenarios, in which we used the model to analyze the performance of the IEEE 802.15.4 model and the advantages of on-demand sensor network reprogramming, respectively.

2. OUR ENERGY MODEL

In this section, we describe the functionalities and implementation details of the energy model developed for OMNeT++. This simulation framework provides a discrete event simulation environment with support for many network protocols such as WLAN (IEEE 802.11), 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 the ad hoc routing protocol DYMO (dynamic MANET on demand) and IEEE 802.15.4, the MAC protocol for ZigBee.

Our energy model is developed as a protocol-independent module and serves as a plug-in to various wireless protocol models in OMNeT++. It adopts the initial battery energy, the radio power in different working state, and the CPU power as its input parameters. Based on these configurations, the model continuously performs the calculation of the energy consumption both on radio and CPU in real-time. If needed, it displays the remaining energy level in animations during the simulation running. Depending on the purpose of the study, the energy model can be configured to execute one of the following two actions upon exhaustion of battery power:

- The simulation is terminated when the first node exhausts its battery power.

- The simulation keeps running until a specified node (e.g. the central node) dies or all nodes in the network die. The dead nodes have to be cut from the network communication, which is maintained by all other active nodes. 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 our energy model, energy consumption on both radio and CPU is considered. Since the energy consumption of the wireless communications is differentiated depending on the current radio state, a proper radio model defining various working states is necessary. Our energy model supports a usual four-state radio interface that are known to lead to a different energy consumption of the node for most hardware platforms:

- Idle
- Sleeping
- Transmitting
- Receiving

To calculate the energy consumed by the radio in real-time, the energy model tracks every state switch in the PHY module using the OMNeT++ notification board. This board allows to centrally observe distributed events. Upon receiving state switch event from the notification board, the energy model updates the accumulated time for each radio state and recalculates the current energy consumption. If no battery power is found left after the recalculation, one of the above mentioned two actions will be executed.

Estimating and modeling energy consumption on the CPU is much more difficult than doing this on the radio, because the activity of the CPU is complex and depends on a couple of factors. For instance, CPU will be busy while processing a packet just received by the MAC or executing some encryption or decryption algorithms. It can also be idle while the radio is busy transmitting or sleeping. Therefore, we can only consider a rough approximation. In our model, we define two CPU states, active and inactive. It is assumed 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.

Finally, the model needs to be calibrated for specific systems (hardware modules). This is done based on measurements as presented before. Here, we need to mention that the degree of details strongly depends on requirements because the energy model consumes processing time in the simulation. Further details of the energy model are discussed in the following section that outlines two application examples.

3. APPLICATION EXAMPLES

3.1 Energy performance of IEEE 802.15.4

Based on a new simulation model of the ZigBee MAC protocol IEEE 802.15.4, we analyzed the performance of this protocol. Some results from these measurements are presented in the following. The simulation model itself is described in [2].

IEEE 802.15.4 defines MAC and physical layers for low-rate wireless personal area networks (LR-WPANs) [5] and the upper layers to form a complete network stack built are specified by ZigBee. The objective of IEEE 802.15.4 is to enable low-cost communication between devices. In particular, the physical layer allows data rates up to 250 kBit/s. The MAC layer provides collision avoidance with CSMA/CA as well as real-time support by reservation of guaranteed time slots. Beaconing is used for synchronization between devices.

At the MAC layer, a superframe structure may be defined by the PAN coordinator that controls an entire network. The structure of a superframe is depicted in Figure 3.

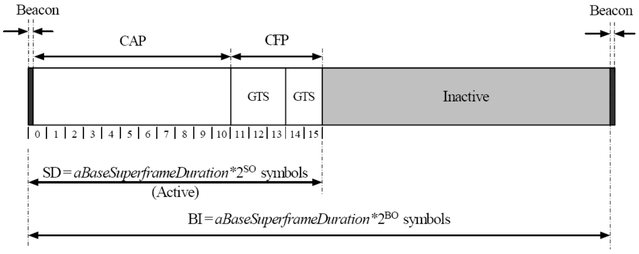


Figure 3: Superframe structure of the MAC layer of IEEE 802.15.4

The implementation in OMNeT++ is outlined in Figure 4. It consists of a PHY and a MAC model plus several supporting models including an interface queue (IFQ) and our energy model.

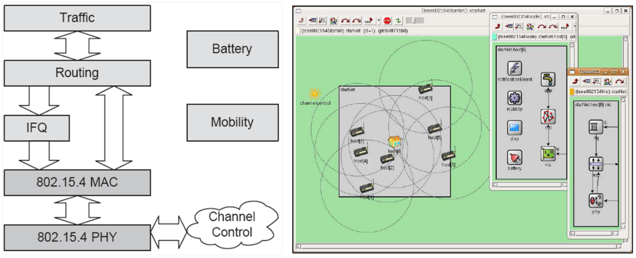


Figure 4: Scheme (left) and implemented models (right) of the IEEE 802.15.4 model

As an example, we analyzed a three node network topology. We connected two devices to a PAN coordinator. The first device is generating packets and sending them to the second device via the PAN coordinator. We analyzed different combinations of BO/SO. The results are shown in Figure 5. Please note that the duty cycle remains constant at 50% for the given BO/SO combinations. Shown is the energy consumption per successfully transmitted byte for different traffic rates ($0.01 \dots 10 s^{-1}$) on a log scale. Obviously, the energy consumption is higher, the lower the traffic rate is. The reasons lies in the long active periods in which no data is transmitted. Thus, this example shows that modeling the CPU energy consumption is essential for evaluating network protocols.

3.2 Network lifetime

In another experiment, we studied the lifetime of dynamically reprogrammed sensor networks. The complete setup including a description of the concept and the main ideas are presented in [4]. In short, we prepared a sensor network with three types of sensors. All the sensor nodes gather data and forward them to a central base station (using WLAN and the ad hoc routing AODV). Mobile robot systems are used for on-demand sensor node reprogramming. In particular, the robots continuously check the application requirements

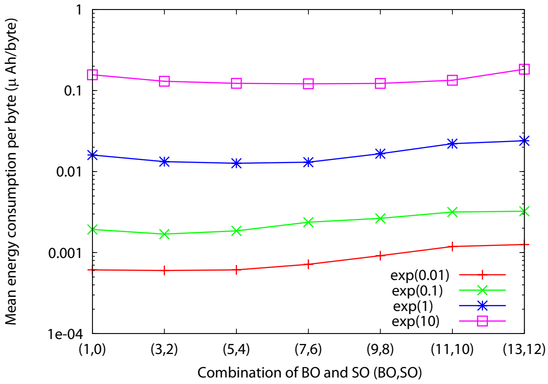


Figure 5: Energy consumption of IEEE 802.15.4 for different traffic rates and different (BO,SO) combinations

and, if necessary, they identify a spare sensor node that can be reprogrammes. We analyzed the sensor network lifetime according to a set of different setups. The programmed application requirement is that each of the four sectors of the network needs at least two of each sensor type to guarantee a sufficient degree of coverage. The scenario is shown in Figure 6.

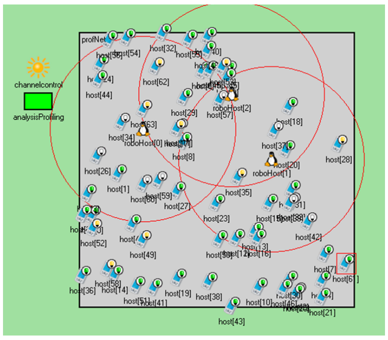
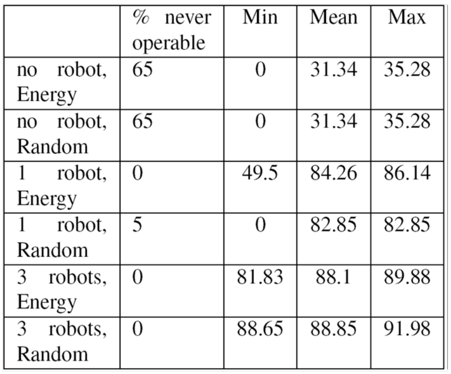


Figure 6: Simulation setup for the lifetime measurements

Some selected measurement results are shown in Figure 7. In order to demonstrate the features of the energy model, we selected three different node programs that lead to a different energy consumption of the node per time:

- P0: simple sensor, measurement cycle is 60 s
- P1: simple sensor, measurement cycle is 10 s
- P2: complex sensor (additional energy consumption for each measurement), measurement cycle is 10 s

Two reprogramming strategies were used by the mobile robot systems: random selection of a nearby node and selection of the node with the most remaining energy. Also, we changed the number of robots (0, 1, 3) and the initial programming of the network was randomly chosen. We analyzed the percentage of network operable time compared to the complete simulation time (the simulation terminated if the application demands cannot be fulfilled any further).



	% never operable	Min	Mean	Max
no robot, Energy	65	0	31.34	35.28
no robot, Random	65	0	31.34	35.28
1 robot, Energy	0	49.5	84.26	86.14
1 robot, Random	5	0	82.85	82.85
3 robots, Energy	0	81.83	88.1	89.88
3 robots, Random	0	88.65	88.85	91.98

Figure 7: Lifetime of the sensor network

4. CONCLUSION

We described a new energy model for use within the OMNeT++ simulation framework. This model allows to evaluate the energy performance and, thus, the network lifetime for arbitrary sensor network applications. In particular, we presented two application examples that we analyzed in related work using the described model. Currently, we calibrated the energy model for sensor nodes of type Mica2 according to measurement results available in the literature. Further types of hardware can be supported according to adequate measurement results. Currently, the model supports four different radio states and we also support the modeling of CPU intensive operations. Future work include fine granular CPU state modeling and support for a wider range of sensor systems.

5. REFERENCES

[1] 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 (WSNS'07)*, Pisa, Italy, October 2007.

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

[3] I. Dietrich and F. Dressler. On the Lifetime of Wireless Sensor Networks. Technical Report 04/06, University of Erlangen, Dept. of Computer Science 7, December 2006.

[4] F. Dressler and I. Dietrich. Lifetime Analysis in Heterogeneous Sensor Networks. In *9th EUROMICRO Conference on Digital System Design - Architectures, Methods and Tools (DSD 2006)*, pages 606–613, Dubrovnik, Croatia, August 2006.

[5] 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.

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

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