Towards Realistic and Credible Wireless Sensor Network Evaluation

Kamini Garg, Anna Förster, Daniele Puccinelli, and Silvia Giordano

Networking Lab
University of Applied Sciences of Southern Switzerland
{kamini.garg,anna.foerster,daniele.puccinelli,
silvia.giordano}@supsi.ch

Abstract. This position paper explores the problem of realistically evaluating wireless sensor network (WSN) applications, algorithms and protocols. It surveys the currently available techniques, such as simulators, testbeds and real world deployments and compares their properties and challenges. While we underline the significance of simulation tools, we also observe that the state of the art simulation models at all levels (from physical to application) still lack realistic behavior. To demonstrate this gap we performed a broad study of simulation models and real world behavior of wireless links and compared those in various settings, including outdoor environments and battery-based deployments. Based on the provided survey and wireless link case study, we outline a strategy of how to enable realistic, efficient, low-cost and repeatable WSN evaluation scenarios.

Keywords: WSN, Simulation Models, Real Deployment, Wireless Propagation.

1 Introduction

This position paper presents our vision and ongoing work on the credible evaluation of Wireless Sensor Networks (WSN) using the three basic evaluation environments available in this field: real-world deployments, testbeds, and simulators. We offer two main contributions: (1) an extensive overview of the state of the art evaluation environments for WSNs and (2) a thorough comparative study of wireless link properties across different environments.

We begin by defining general credibility and usability requirements for WSN evaluation in Section 2. Section 3 presents a detailed overview of the state of the art: we explore the three basic evaluation environments in terms of wireless propagation, energy consumption and battery behavior, as well as application-level events. In Section 4 we dive deeper into one of these dimensions, namely wireless propagation, and present a comparative study between various real-world environments (indoor and outdoor), energy sources (wall power and battery power), as well as different simulation models. The main goal of this study is to highlight the huge differences between those models. Finally, based on the survey and the case study, we define our credible WSN evaluation strategy and present our ongoing work in Section 5.

D. Simplot-Ryl et al. (Eds.): ADHOCNETS 2011, LNICST 89, pp. 49–64, 2012.

2 Evaluation Requirements for Sensor Networks

The driving motivation of this work is to enable credible and convenient evaluations for WSNs. First of all, we abstract away from any specific evaluation methodologies, approaches, and models, and we identify the key properties of WSN evaluation environments and studies:

Scalability: The evaluation environment needs to support any number of network nodes and node density.

Flexibility: The environment needs to support various parameters and scenarios, such as indoor and outdoor wireless propagation, node mobility, or different hardware platforms.

Accuracy: The environment must reproduce the real-world behavior of WSN deployments.

Repeatability: Each experiment must be 100% repeatable.

Visibility: The distributed network state must be visible to the user at any given time.

Cross-Environment Validity: Studies in one environment must be comparable to studies in another environment.

Re-Usability: Implementations targeted for a given environment must be reusable in others.

This list is clearly idealistic and very hard to achieve in practice. However, it provides us with a solid basis of the goals and requirements for any evaluation scenario or environment and allows us to compare different evaluation environments to each other, such as simulation against testbeds.

The above requirements are not new and have been defined many times before [11,18]. However, as we will show in the next section, there is still no standardized evaluation environment, nor is there a consensus on which ones should be uses. The main goal of this paper is to demonstrate the gap between real world deployments and current evaluation environments and to propose a new evaluation suite, which covers most of the above requirements and enables credible and usable techniques for the evaluation of WSNs.

3 State of the Art of Evaluation Approaches

In this section, we turn our attention to existing WSN evaluation models and environments. We concentrate on three key approaches: simulators, testbeds and real-world deployments. Our approach is top-down: first, Table 1 presents a general comparison between those three environments in terms of the WSN evaluation requirements we defined in the last section. Then, we dive into each of them and break them down into their components to discuss their individual properties.

Table 1. High-level comparison of different WSN evaluation environments

WSN Evaluation Requirements	**Real world deployments**	**Indoor testbeds**	**Network Simulators**
Scalability	Scalable, but very costly (money, time, effort)	Scalable, but costly	Unlimited scalability
Flexibility	Medium/Rigid	Rigid	Flexible
Accuracy	Accurate	Less accurate	Inaccurate
Repeatability	Medium	Good	Perfect
Visibility	Low	Medium	Perfect
Cross-environment validity	High	High	Low
Re-usability	Medium/high	Medium/high	Low

Looking at Table 1, real world deployments clearly offer the most realistic evaluation environment, but are also very hard to manage, are costly, require a lot of effort and time, are typically not flexible, and are generally not conducive to repeatability. Testbeds offer a great alternative and offer better repeatability; however, even if the nodes are stationary, the environment can still change quickly and unpredictably [16]. Visibility is generally good because testbeds typically come with a dedicated infrastructure (backchannel). However, testbeds are usually wall-powered, thus lacking the complexity and the unreliable behavior of battery-deployed sensor nodes.

Simulation can offer great repeatability, visibility, unlimited scalability and flexibility. Unfortunately it suffers heavily from the poor accuracy of implemented simulation models and protocols, has very low reusability in general (especially general use network simulators) and the results gathered on one simulator are hardly comparable to results from others.

In the next paragraphs we explore the individual components of testbeds and simulators and compare them to real world deployments, which we identify as the ground truth. We argue that simulation is a currently underestimated and underused tool in the WSN community, whose problems and drawbacks could be solved by implementing cross-platform valid simulation models. The provided survey in the next paragraphs is the key to finding the right models.

The components we explore are wireless propagation, battery consumption, energy expenditure, and application-level events. We step through each of them in the next paragraphs.

3.1 Wireless Propagation

Wireless propagation in real world WSN deployments is extremely complex and depends on the radio hardware, its physical layer calibration, the orientation of the antenna, the inter-node distance (large-scale path loss), the presence of obstacles that block off the line of sight (shadowing), the presence of radio interference, the geometry of the surrounding environment (responsible for multipath fading), and the general conditions (temperature, humidity, sunlight, wind,...).

Small changes in the deployment area can change dramatically the properties of the wireless medium and lead to fundamentally different results. This complexity makes wireless links inherently unreliable and results in phenomena such as transitional links [31], asymmetric links, and burstiness. WSN testbeds are built with real hardware. They are typically deployed indoors, in office buildings that generally provide rich scattering conditions, thus accentuating the impact of multipath fading over the large-scale path loss and providing propagation conditions that are extremely different from the ones of outdoor deployments. Another key difference between real deployments and testbeds in terms of wireless propagation is the impact of the energy source. While real world deployments typically rely on off-the-shelf batteries, testbeds are wall-powered. We have shown in [13] that the vagaries of commercial batteries have a significant impact on the performance of low-power transceivers, and this observation should be taken into account in WSN evaluation and modeling.

Simulation models rely either on real wireless traces replay or on mathematical models. Well-known mathematical models [19] such as the unit disk model, the free-space propagation model, the two ray ground model, or the log-normal shadowing model fail to reproduce the real world behavior of WSNs. Much better alternatives have been also proposed, such as the Radio Irregularity Model [30], but are rarely used in practice, mostly because of their implementation complexity.

Playback of real world wireless traces has recently become a preferred simulation strategy, as it recreates the behavior of real world wireless propagation. Several variations of this model exist. One of the first trace-based models was implemented in TOSSIM [8,9,21], the standard TinyOS simulator, using real link traces to compute the empirical delivery probability and the statistical distribution of RSSI values on the individual links. Such a model accurately reproduces key properties of wireless channels, such as asymmetric links and link quality fluctuation, but it fails to represent link burstiness and the usage of a random number generator (RNG) hinders repeatability, especially across different simulators.

The problem of including burstiness in trace-based simulations has been tackled in WSNSimPy [11], a WSN simulator written in Python. It uses real-world traces by storing the real link qualities individually. When a node needs to transmit, it randomly selects one of the entries in the trace library. If the next transmission of the node is conducted soon thereafter, the next entry form the trace file is used instead of a random one. This captures relatively well the real burstiness of links. However, it does not allow perfect re-play of the experiment, again because a random number generator is used.

The work in [6] presents an algorithm called Multi-level Markov Model (M&M) to produce synthetic traces with the same statistical properties as some real trace and thus to simplify the process of gathering traces and ”stretching“ them. However, to the best of our knowledge, there is no fully deterministic implementation of wireless trace usage.

An inherent advantage of trace-driven simulation over model-based approaches is that model implementations change over time, while traces do not. When new model implementations become available, the old ones become obsolete. When new traces becomes available, the old ones still represent valid benchmarks.

In the next Section 4 we present an experimental study of the accuracy of various mathematical and trace-based models, compared to indoor and outdoor real world deployments.

3.2 Power Consumption Models

Power consumption refers to the current draw at the sensor node and its individual components (micro-controller, sensors, flash memory, external memory, LEDs, radio transceiver, etc). Some testbeds (e.g. MoteLab [25]) have limited support for measuring of the actual power draw, but typically the on-time of various components is used as the best proxy for energy consumption. Simulation provides finer-grained information.

Generally, simulators employ simple power consumption models, which differentiate between components and their current state and assign a current draw to each of those. Coarse grained models, such as the EnergyFramework of OMNeT++/MiXiM [4] consider only the radio and its main states sleep, receive and transmit. Finer grained models, such as in Cooja [3] or PowerTOSSIM [23], also consider sensors, micro-controller, and LEDs. In [13] we have shown that the most power hungry components on a system are usually the radio transmitter and the on-board flash. This is of course true for the specific sensor node hardware used for the experiments (TelosB); other platforms may behave differently.

3.3 Battery Models

Battery models are different from energy expenditure models, as they measure what is the remaining energy in the batteries over time. Relevant studies include [12] and [13], which focuses on the battery discharge behavior.

While no existing testbed employs batteries, there exist several simulation models. The most widely used model is linear and assumes that the battery is a bucket of energy units that are used up over time. A much more sophisticated non-linear model is proposed by Rakhmatov and Vrudhula [17], and captures the discharge and the recovery effects of batteries.

3.4 Application-Level Events

Application-level traces are rare in practice, both for testbeds and for simulation. Cooja is such a rare example, which use the WiseML [10] format for replaying application traces from real-world environments. The basic idea is that the application trace is a sequence of events at the application level with local timestamps. These traces should be preferably recorded at real world deployments, but can also be artificially created. Application-level traces build a basis for structured testing and evaluation, as they provide the lower communication protocols with traffic, which they need to manage.

Table 2. List of state of the art simulation models and their availability for various simulators

Simulation Model	**Cooja**	**ns-3**	**OMNeT++ / MiXiM**	**Shawn**	**TOSSIM**	**Qualnet**
Deterministic wireless link traces [6]	Trace-based, median based	Yes for ns-2, own format [14]	In progress (WiseML format)	No [22]	Non-deterministic median-based	No information
Fine-grained energy expenditure model	Yes [3]	Yes [29]	Yes, radio only [4]	No	Yes, PowerTOSSIM [23]	No information
Non-linear battery model of Rakhmatov-Vrudhula [17]	No Information	Yes	No	Yes	Own Non-linear model Power-TOSSIMz [15	Yes
Application-level traces [1]	Yes, WiseML	No Information	Yes, own format	Yes, WiseML	No	No Information
General credibility	**medium**	**medium**	**low/medium**	**low/med**	**medium**	**low**
User friendliness, support, documentation	Strong	Weak	Strong	Medium	Medium	Strong
Supported real operating systems	Contiki	TinyOS in progress [20	In progress (TinyOS, Scatter-web) [5]	Implicitly, through WiseLib [1]	TinyOS	No

3.5 Summary

Table 2 offers an overview of all described sophisticated evaluation/simulation models for some of the most popular simulation environments for WSNs. We point out that individual network simulators rarely offer a complete suite of sophisticated simulation models and model suites of different simulators are never the same. This lowers the credibility of individual simulators but also makes cross-platform comparison of simulation results impossible. The credibility grade we gave for each simulator is valid only if the most sophisticated models suite available is used. This is a crucial requirement to achieve credibility, as all simulators also offer simplified models, such as the unit disk model for wireless propagation.

Next, we present a detailed study of wireless propagation from real world deployments, testbeds and simulation models and demonstrate once again the gap between them.

4 Case Study on Wireless Links Properties

We have studied the properties of wireless links with lengths ranging from 2 to 6 meters in several different environments with various parameters: battery-powered indoor and outdoor real deployments, wall-powered indoor and outdoor testbeds, wireless trace based simulation with various noise addition under TOSSIM [9] and various mathematical models in OMNeT++/MiXiM [26]. While the main purpose of this study is to show the clear discrepancies between simulation models and reality and even between different environments (indoor and outdoor), we note that, to the best of our knowledge, a comparison between a wall-powered and a battery-powered deployment has never been carried out before.

We consider standard link metrics such as delivery rate and RSSI. In addition to the mean values, we present broad statistical values such as lower and higher quartiles, outliers, etc. We claim that the low accuracy of most simulation models is due to the fact that they are unable to completely capture the significant fluctuation of the individual metrics.

4.1 Experimental Setup

For our real world and testbed deployments, we use a basic star topology with 2 or 3m distance from the center, see Figure 1. We use TelosB nodes with minimum transmission power setting (-25dBm). The application is based on TinyOS and is taken from [6]: one node in the network is assigned the role of the sender, while all others are receivers. The sender sends packets with an inter-packet interval (IPI) of 20 ms and the receivers log the received packets with packet id, RSSI and LQI level. After some time (2-10 minutes), we switch to a different sender. The logged data is either forwarded to the testbed base station via the serial interface or is logged locally. Note that this approach avoids any kind of inter-node interference and only external interference is present.

A typical example of a code-level WSN simulator is TOSSIM, a part of the standard TinyOS [9]. TOSSIM is a discrete event simulator, where simulation events represent hardware interrupts, high-level system events and posted tasks. The basic TOSSIM wireless channel model is based on defining the large-scale path loss for each pair of nodes in both directions. Loss values can be obtained from real world traces or based on a radio propagation model. RF noise and interference from other nodes and outside sources are also simulated [8]. The Closest Pattern Matching (CPM) algorithm is used to analyze real noise trace and create a statistical model from it [21]. We map our experimental data for

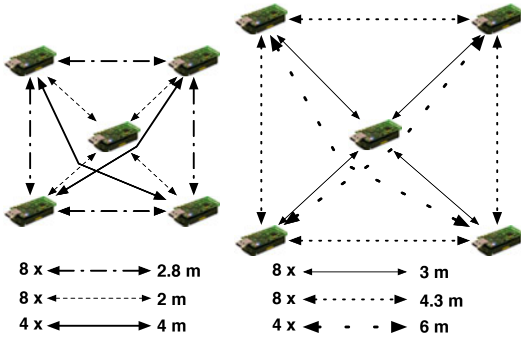

Fig. 1. Used topologies with all unidirectional links and their distances

both wall-powered and battery-powered nodes by giving the network topology information and the average directional RSSI between all node pairs. In addition to that, we also gathered noise traces around each node by using the standard *RssiSample* [8] program. Similarly to our real experimental setup, we use the same 5-node topologies with IPI of 20 ms.

We also investigate the performance of a typical general-use event-based network simulator OMNeT++ with its mobile ad hoc extension MiXiM [7]. We developed a simple communication stack, consisting of a simple CSMA MAC protocol and an application. The application is the same as our TinyOS implementation and the same topologies are reproduced. However, in contrast to TOSSIM, MiXiM implements mathematical models for the simulation of wireless propagation. The currently available models are LogNormalShadowing, SimplePathLoss and JakesFading. One immediate disadvantage of these models is the possibility to freely combine them. This makes it possible, for example, to use Jakes Fading alone (that should be used only in combination with path-loss). We employed the parameters from the real world deployments (radio frequency, transmission power, topology, etc.) as closely as possible.

4.2 Delivery Rate

Figure 2 presents the obtained results in terms of delivery rate. The top row of graphs corresponds to our hardware experiments with battery-powered and wall-powered nodes in outdoor and indoor environments. To the best of our knowledge, this study is the first to methodologically explore the differences in link quality between battery-powered and wall-powered nodes. As it can be seen from the top graphs, the indoor links exhibit a remarkably stable behavior even over different distances and there is no major difference between wall- and battery-powered deployments. However, outdoor links are completely different in their statistical values: the median of the delivery rate falls significantly with increasing distance between the nodes, and, most importantly, the battery-powered experiments almost fail to deliver any data. The consequence of this observation is clear and important:

Observation 1. *Battery-powered indoor testbeds do not mimic the behavior of outdoor battery-powered deployments.*

Next, we compare the above real world experimental data with ones obtained from TOSSIM simulations. Note that we used the real-world trace data from each of the above described deployments to mimic its behavior in TOSSIM. We present two TOSSIM settings: one with constant noise for all nodes, taken from noise measurements of the central node in the topologies, and one with individual noise traces for each node. It is interesting to note that it seems TOSSIM with individual noise traces is able to pretty well mimic the behavior of real-world links for wall-powered links, and less for battery-powered ones. The difference becomes extreme for outdoor battery-powered nodes, even if this particular data sets was used for the trace-based simulation.

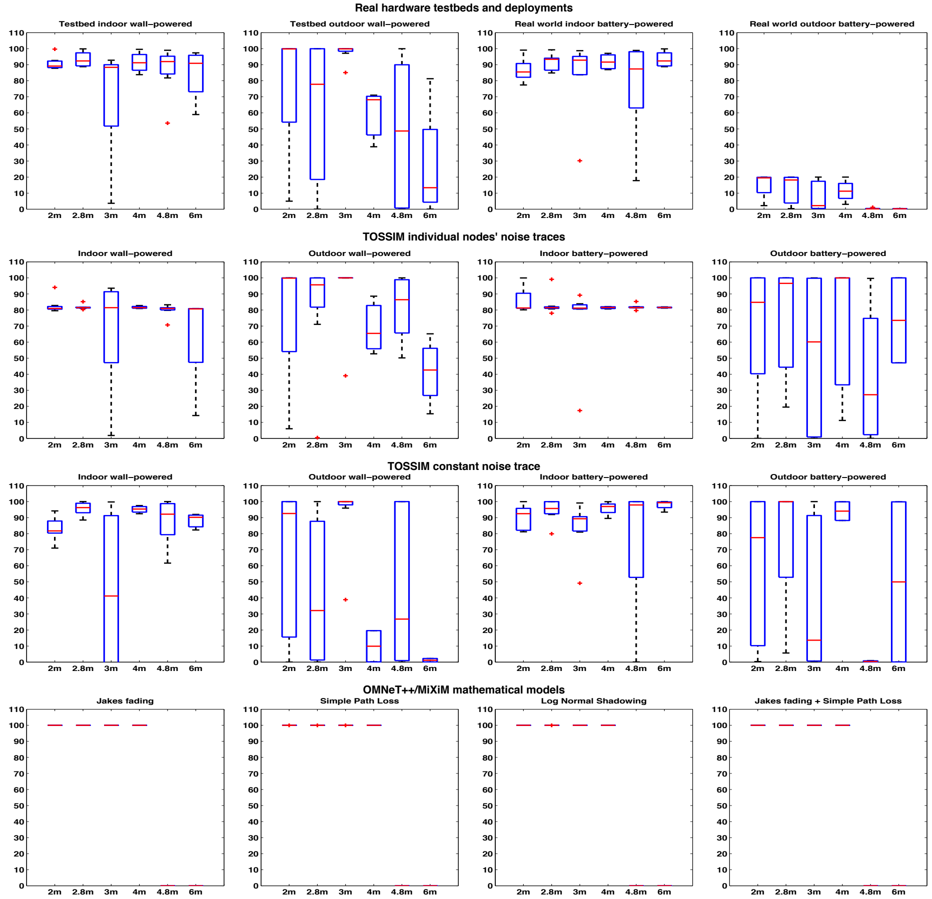

Fig. 2. Delivery rate over various environments. The box has lines at the lower quartile, median, and upper quartile values. Whiskers extend from each end of the box to the adjacent values in the data within 1.5 times the interquartile range from the ends of the box. Outliers are displayed with a + sign.

TOSSIM simulations with constant noise traces for all nodes seem to be less accurate, compared to TOSSIM with individual noise traces. The difference between simulation and reality for outdoor battery-powered links is significant.

We also ran TOSSIM experiments without any noise data, but failed to exchange even a single packet. The reason for this is the receiving threshold of TOSSIM, which is set to -72 dBm and is derived from experimental data collected using two MicaZ nodes, RF shielding, and a variable attenuator [24]. Obviously, this threshold is not very realistic for our deployments and needs to be carefully re-validated.

For OMNeT++/MiXiM and its mathematical models, only one conclusion can be drawn: even if the implemented models are considered more sophisticated and realistic than the the simple unit disk model, they completely fail to simulate the lossy nature of links and produce almost binary links.

4.3 Received Signal Strength Indicator

Figure 3 presents the data of our case study in terms of logged/calculated RSSI values for individual packets. Real-world deployments and testbed tend to have very fluctuating values even in short time intervals, as it has been shown many times before, e.g. [16,31]. However, new results present the difference between battery-powered and wall-powered links and between indoor and outdoor links. The observation is the same as for delivery rate: outdoor links have different properties than indoor ones and are generally less reliable. Some of the reasons behind *Observation 1* might be temperature [2], humidity, wind etc. Overheating of sensor nodes (due to sunlight), specially affected battery-powered nodes and reduced their PDR and RSSI.

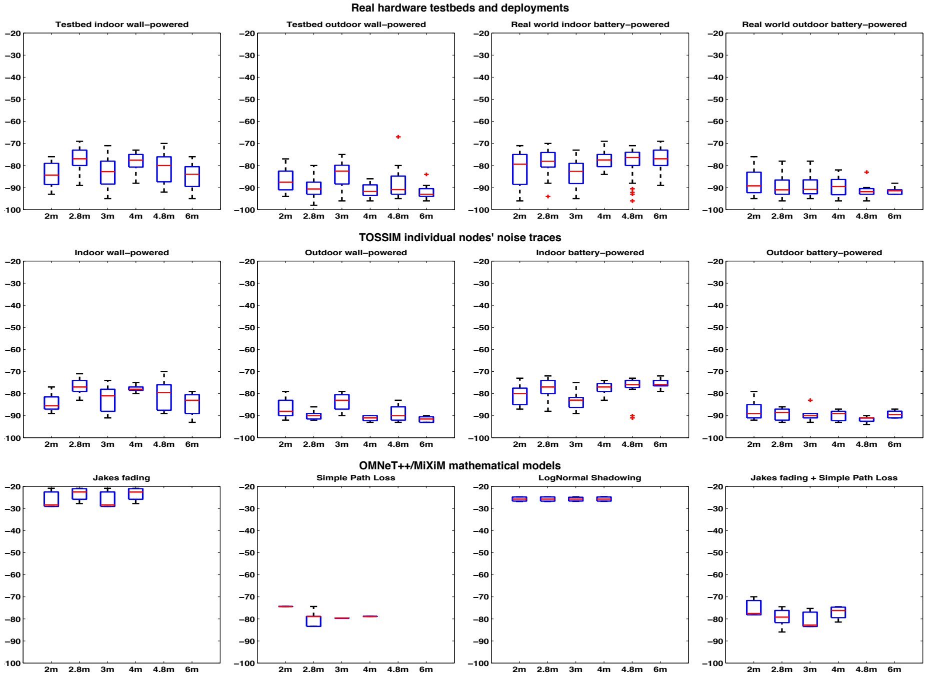

Fig. 3. RSSI values over various evaluation environments. The box has lines at the lower quartile, median, and upper quartile values. Whiskers extend from each end of the box to the adjacent values in the data within 1.5 times the interquartile range from the ends of the box. Outliers are displayed with a + sign.

TOSSIM simulations seem to mimic the real-world quite well, although the fluctuations of the values are lower. OMNeT++/MiXiM presents completely different results for the different simulation models. For example, Jakes Fading and LogNormal Shadowing produce unrealistically good RSSI values, far away of any real-world observations. On the other hand, Simple Pathloss, the simplest of the here presented models, produces much more realistic data, which almost fall in the same interval as real-world observations. The best performing combination of these models was identified to be Jakes fading and simple pathloss, which in combination produce a realistic data interval with some fluctuations. Given the discussions about delivery rate and RSSI values, it can be also generalized that parametrization and usage of simulation models is tricky and requires expertise.

Observation 2. *Parametrization of simulation models is a major challenge towards credible, realistic evaluation.*

5 Enabling Credible WSN Evaluation

This paper has presented an extensive study and comparison of WSN evaluation models and environments. We explored several important properties of WSNs, such as wireless propagation, energy expenditure, etc. in real world deployments, testbeds, and simulation. As expected, real-world deployments naturally provide the most realistic environment, simulation still leaves a lot to be desired, and testbeds lie somewhere in between (see Table 1). In the next paragraphs, we outline our vision for streamlining the evaluation process of WSNs in all environments and mainly in simulation and discuss related efforts in this area, including our own ongoing work.

5.1 Credible WSN Simulations

The main drawback of WSN simulations is their low credibility because of over-simplified simulation models. As our case study on wireless propagation in Section 4 has shown, there are great discrepancies between simulation models and reality. Furthermore, another significant challenge is the parametrization and usage of these models. TOSSIM's trace-based simulation of wireless links presents a credible environment, but TOSSIM users need to be aware of all possible additional models, such as add-on noise traces or the sensitivity threshold, in order to achieve maximum credibility. A novice or non-expert can easily miss some details and use unconsciously a simplified, less-credible environment. We claim that full and safe simulation credibility can be achieved only by implementing deterministic, parameter-free simulation models. This approach will not only simplify the work with simulators and make them credible, but also enable cross-platform comparison of simulation results. The most important models to be implemented are:

Wireless Propagation Model. The most deterministic and realistic model is the playback of real wireless traces. This model has two main requirements: implementing the model itself and building a wireless trace database with sufficient number of traces from different environments and different topologies. The important implementation challenges of trace driven wireless propagation model include inter-node interference model, interference from distant nodes, spatial and temporal extension of available traces etc. To the best of our knowledge, there is no deterministic playback model implemented yet for any simulator (see Section 3). Our own work in progress includes such a model for OMNeT++/MiXiM and TOSSIM. In terms of data format for the captured wireless traces, WiseML [10] is a perfect candidate, since a small database of traces in this format is already available [28].

Fine-Grained Energy Expenditure Model. Energy expenditure is a vital metric of WSNs and thus needs to be monitored carefully. Simulation is a great tool for this, as it offers fine-grained state observation of individual components. A credible energy expenditure model will include at least the radio transceiver, sensors, and processing. It must capture at least their main states, e.g. sleep, receive, transmit for the radio. A significant component of this model are realistic, fine-grained energy expenditure measurements from real platforms, such as PowerTOSSIMz [15]. The remaining challenge is to perform fine-grained real-world experiments of energy consumption of individual components on a wide range of sensor platforms.

Non-Linear Battery Model. Battery models are closely related to energy expenditure models and enable predictions about the lifetime of sensor nodes. Credible models capture the non-linear behavior of batteries like self-discharge, fluctuating output voltage, etc. However, even sophisticated battery models like Rakhmatov-Vrudhula [17] remain to be validated for a complete battery lifetime on real sensor nodes.

Application Model. This is probably the least complex model, which we require for credible simulations. However, this upper layer dictates when events occur in the network and how they are disseminated. Thus, we also dictate the data traffic in the whole network, which can be periodic, bursty, event-based, etc. Such models exist for OMNeT++, Cooja and Shawn. The last two support WiseML for reading application events, OMNeT++ supports its own simple table format.

5.2 Optimized WSN Testbeds

WSN testbeds have proved to be a handy tool for evaluating WSNs. One of their main advantages is the direct portability of code between a testbed and a real environment. However, as we discussed in Section 3, there exists a gap between a testbed and a real world deployment, because testbeds are wall-powered, deployed indoors, and typically completely stationary. In order to counteract these challenges, we propose the following further optimizations of existing testbeds:

Wireless Propagation Model. There is a clear need to reproduce the vagaries and features of outdoor deployments. Deploying testbeds outdoors is generally not practical. We propose the implementation of a trace-based simulation wireless channel, identical to the above described deterministic wireless propagation model for simulations, enabled through the backchannel infrastructure of the testbed. This will allow two important novelties: perfect repeatability of experiments and cross-platform comparison between simulation and testbeds. Of course, the main goal is to be able to playback real world wireless traces from any environment. Note that this model does not eliminate the other hard challenges of testbeds, such as real hardware, clock drift, processing time, etc.

Energy Consumption Model. An energy consumption model needs to monitor the work of individual hardware components and to estimate on-line their individual energy consumption. Such a functionality is already provided by the embedded operating system Contiki [3]; TinyOS implementations are not readily available, but trivial to add.

Battery Model. Using real batteries on remote testbeds is very inconvenient and does not mimic outdoor battery-powered deployments, see Section 4. We propose the adoption of a battery simulation model, as described for simulations above. This can be easily combined with the energy consumption model, where the testbed server monitors the energy consumption of all nodes and automatically shuts down the ones whose simulated battery dies. To the best of our knowledge, this simple idea has not been considered so far.

5.3 Real World Deployments

The main disadvantage of real-world deployments is their restricted visibility that makes debugging very challenging. Furthermore, energy consumption measurements are typically out of scope. Much effort has been already invested in improving the visibility and debugging tools for real world deployments, for example Marionette [27] for TinyOS, which enables remote function call invocation on sensor nodes. Our own efforts in developing FLEXOR [5] also target more visibility and control over remote sensor nodes.

5.4 Summary

A generalized view of our implementation and research strategy towards credible WSN evaluation is depicted as a work flow graph in Figure 4. The center of the graph build the three evaluation environments: real world, testbeds and simulation. They are surrounded by cross-platform operating systems, programming languages, and software architectures in order to enable portability and re-usability of code across different environments. The rest of the graph consists of the four models described above: non-linear batteries, energy consumption, wireless propagation and application events. For each of them the necessary steps are depicted, with input coming mostly from real world deployments (the bold lines). The grey vertices underline future work, while white vertices represent already available tools and methodologies, which might need some extensions only.

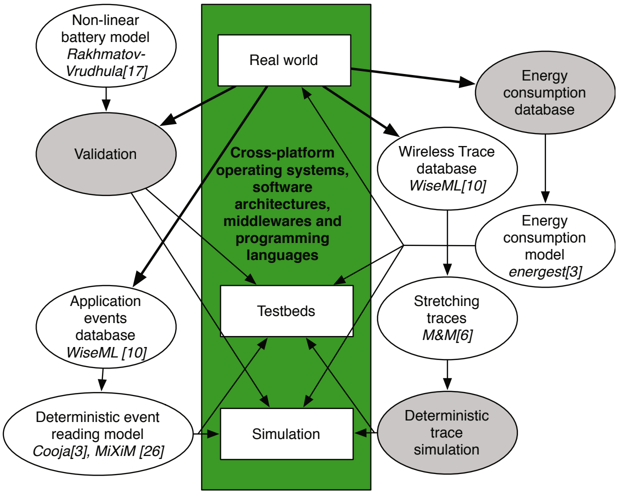

Fig. 4. Future directions for enabling credible WSN evaluation. See inline for detailed explanation.

6 Conclusion

In this paper, we presented our vision and strategy towards enabling credible, realistic and convenient WSN evaluation. We presented the basis of our strategy, consisting of broad state of the art survey of evaluation approaches and models and a rigorous case study to identify and demonstrate the gaps between real world deployments and current evaluation and simulation models. Our immediate future plans include the implementations of deterministic wireless trace based simulation model for OMNeT++/MiXiM and TOSSIM and WiseML-support to those two simulators. Discussions and collaborations are highly welcome.

References

1. Baumgartner, T., Chatzigiannakis, I., Fekete, S., Koninis, C., Kröller, A., Pyrgelis, A.: Wiselib: A Generic Algorithm Library for Heterogeneous Sensor Networks. In: Silva, J.S., Krishnamachari, B., Boavida, F. (eds.) EWSN 2010. LNCS, vol. 5970, pp. 162–177. Springer, Heidelberg (2010)
2. Boano, C.A., Brown, J., He, Z., Roedig, U., Voigt, T.: Low-Power Radio Communication in Industrial Outdoor Deployments: The Impact of Weather Conditions and ATEX-Compliance. In: Komninos, N. (ed.) SENSAPPEAL 2009. LNICST, vol. 29, pp. 159–176. Springer, Heidelberg (2010)

3. Dunkels, A., Osterlind, F., Tsiftes, N., He, Z.: Software-based on-line energy estimation for sensor nodes. In: Proceedings of the 4th Workshop on Embedded Networked Sensors (EmNets), Cork, Ireland (2007)
4. Feeney, L.M., Willkomm, D.: Energy framework: an extensible framework for simulating battery consumption in wireless networks. In: Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques, Torremolinos, Malaga, Spain (2010)
5. Förster, A., Förster, A., Leidi, T., Garg, K., Puccinelli, D., Ducatelle, F., Giordano, S., Gambardella, L.: Poster abstract: Motel: Towards flexible mobile wireless sensor network testbeds. In: Proceedings of the 8th European Conference on Wireless Sensor Networks (EWSN), Bonn, Germany (February 2011)
6. Kamthe, A., Carreira-Perpiñán, M.A., Cerpa, A.E.: M&M: multi-level Markov model for wireless link simulations. In: Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, SenSys 2009, pp. 57–70. ACM, Berkeley (2009)
7. Köpke, A., Swigulski, M., Wessel, K., Willkomm, D., Haneveld, P.T.K., Parker, T.E.V., Visser, O.W., Lichte, H.S., Valentin, S.: Simulating wireless and mobile networks in omnet++ the mixim vision. In: Proceedings of the 1st International Conference on Simulation Tools and Techniques, Marseille, France (2008)
8. Lee, H., Cerpa, A., Levis, P.: Improving wireless simulation through noise modeling. In: Proceedings of the 6th International Conference on Information Processing in Sensor Networks, IPSN 2007, Cambridge, Massachusetts, USA, pp. 21–30 (2007)
9. Levis, P., Lee, N., Welsh, M., Culler, D.: Tossim: accurate and scalable simulation of entire tinyos applications. In: Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys), Los Angeles, CA, USA, pp. 126–137 (2003)
10. Li, Q., Österlind, F., Voigt, T., Fischer, S., Pfisterer, D.: Making wireless sensor network simulators cooperate. In: Proceedings of the 7th ACM Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks (PeWASUN), Bodrum, Turkey (2010)
11. Marchiori, A., Guo, L., Thomas, J., Han, Q.: Realistic performance analysis of wsn protocols through trace based simulation. In: Proceedings of the 7th ACM Workshop on Performance Evaluation of Wireless Ad Hoc, Sensor, and Ubiquitous Networks, PE-WASUN 2010, pp. 87–94. ACM, Bodrum (2010)
12. Menzel, T., Willkomm, D., Wolisz, A.: Improving battery-efficiency of embedded devices by favorably discharging only towards end-of-life. In: Proceedings of the CONET 2011 Workshop in Conjunction with CPSWeek 2011, Chicago, USA (April 2011)
13. Nguyen, H.A., Förster, A., Puccinelli, D., Giordano, S.: An experimental study of sensor node lifetime. In: Proceedings of the 7th IEEE International Workshop on Sensor Networks and Systems for Pervasive Computing (PerSens). Seattle, WA, USA (2011)
14. NS-2 (May 2011), `http://nsnam.isi.edu/nsnam/index.php`
15. Perla, E., Catháin, A., Carbajo, R.S., Huggard, M., Mc Goldrick, C.: Powertossim z: realistic energy modelling for wireless sensor network environments. In: Proceedings of the 3nd ACM Workshop on Performance Monitoring and Measurement of Heterogeneous Wireless and Wired Networks, Vancouver, British Columbia, Canada (2008)
16. Puccinelli, D., Gnawali, O., Yoon, S., Santini, S., Colesanti, U., Giordano, S., Guibas, L.: The Impact of Network Topology on Collection Performance. In: Marrón, P.J., Whitehouse, K. (eds.) EWSN 2011. LNCS, vol. 6567, pp. 17–32. Springer, Heidelberg (2011)

17. Rakhmatov, D., Vrudhula, S.: Energy management for battery-powered embedded systems. ACM Transactions on Embedded Computing Systems 2, 277–324 (2003)
18. Raman, B., Chebrolu, K.: Censor networks: A critique of "sensor networks" from a systems perspective. ACM SIGCOMM Computer Communication Review 38(3), 75–78 (2008)
19. Rapapport, T.: Wireless Communications: Principles and Practice. Prentice Hall (2001)
20. Riliskis, L., Osipov, E., Maróti, M.: Tos-ns3: a framework for emulating wireless sensor networks in the ns3 network simulator. In: Proceedings of the 3rd International Workshop on NS3, in Conjunction with SimuTOOLS, Malaga, Spain (2010)
21. Rusak, T., Levis, P.A.: Investigating a physically-based signal power model for robust low power wireless link simulation. In: Proceedings of the 11th International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems, pp. 37–46 (2008)
22. Shawn: `https://www.itm.uni-luebeck.de/shawnwiki`
23. Shnayder, V., Hempstead, M., Chen, B.R., Allen, G.W., Welsh, M.: Simulating the power consumption of large-scale sensor network applications. In: Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, SenSys 2004, Baltimore, MD, USA, pp. 188–200 (2004)
24. TOSSIM: Simulating TinyOS Networks: `http://www.cs.berkeley.edu/~pal/research/tossim.html`
25. Werner-Allen, G., Swieskowski, P., Welsh, M.: Motelab: a wireless sensor network testbed. In: Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN), pp. 483–488 (April 2005)
26. Wessel, K., Swigulski, M., Köpke, A., Willkomm, D.: Mixim: the physical layer an architecture overview. In: Proceedings of the 2nd International Workshop on OMNeT++, Rome, Italy, pp. 1–8 (2009)
27. Whitehouse, K., Tolle, G., Taneja, J., Sharp, C., Kim, S., Jeong, J., Hui, J., Dutta, P., Culler, D.: Marionette: using rpc for interactive development and debugging of wireless embedded networks. In: The Proceedings of the 5th International Conference on Information Processing in Sensor Networks, IPSN 2006, pp. 416–423 (2006)
28. Wisebed (2011), `http://www.wisebed.eu`
29. Wu, H., Nabar, S., Poovendran, R.: An energy framework for the network simulator 3 (ns-3). In: Proceedings of the 4th International Conference on Simulation Tools and Techniques (SimuTOOLS), Barcelona, Spain (2011)
30. Zhou, G., He, T., Krishnamurthy, S., Stankovic, J.A.: Models and solutions for radio irregularity in wireless sensor networks. ACM Transactions on Sensor Networks 2(2), 221–262 (2006)
31. Zuniga, M., Krishnamachari, B.: An analysis of unreliability and asymmetry in low-power wireless links. ACM Transactions on Sensor Networks 3 (June 2007)