Simulating Intermittently Powered Embedded Networks

Muhammad Hamad Alizai, Qasim Raza*, Yasra Chandio, Affan A. Syed†, Tariq M. Jadoon
SysNet Lab, SBA School of Science and Engineering, LUMS, Lahore, Pakistan
†PLUMgrid Inc. and FAST NU, Islamabad, Pakistan
{hamad.alizai,jadoon}@lums.edu.pk, {qasim.raza,yasra.chandio}@sysnet.org.pk, asyed@plumgrid.com

Abstract

With the promise of delivering immortality, energy harvesting and wireless energy transfer have become the next research frontier for pervasive computing and networks. The challenge still is to adapt operations along all aspects of a sensing system (sensing, computation, communication), to deal with the varied, unpredictable or possibly intermittent supply of such energy. To understand these challenges, we currently lack sufficient tools to evaluate the impact of different energy harvesting and transfer models - which is further exacerbated due to the high cost of deploying such systems at a large scale. We present a generic, TOSSIM-based simulation framework to model energy harvesting and energy transfer, enabling rapid development of harvesting- and transfer-aware applications, protocols, and system software. Our evaluation shows that even an abstract simulation model can provide useful insights, such as frequent power outages and node reboots due to intermittent energy supply. Based on these insights, we further establish the utility of this framework by demonstrating how high level simulations can lead to a better choice of energy scheduling algorithms.

Categories and Subject Descriptors

C.2 [**Computer-Communication Networks**]: Miscellaneous; I.6 [**Simulation and Modeling**]: Miscellaneous

General Terms

Experimentation

Keywords

Wireless energy transfer, energy harvesting, simulation

1 Introduction

Energy harvesting (EH) is the conversion of ambient energy into electric energy. An environment can posses ambient energy either due to the presence of natural sources, such as sunlight or vibrations, or due to intentional provisioning through wireless energy transfer (WET) using, for example, a laser or radio transmitter. Such energy sources replace the assumption of a finite energy budget in embedded networks, such as IoT and WSN, with that of potentially infinite, but possibly low power (RF requiring buffers) or intermittent supply (due to diurnal or other scheduled transfer) [1, 15]. Both these scenarios require computation that survives power blackouts—*intermittent computing*. Due to this potential of enabling perpetual deployments, in recent years EH and WET have received an overwhelming attention: new hardware platforms [4, 7, 12] and corresponding software solutions [1, 15] have been developed, the number of publications has rapidly increased exploring different types of ambient energy (natural [8, 12, 16] or dedicated WET [3, 9, 13]), and new workshops on this topic have also been initiated.

Not long ago, we witnessed a similar transition of the Internet from wired to the wireless medium, where the resulting push for simulation tools to model the unpredictable nature of wireless communication provided an important impact on networking research by providing quick and early feedback. While the early tools lacked accuracy, they were iteratively improved by the community as their benefit was undeniable. The WSN community also embraced a similar effort to identify protocols that deal with a wireless nature and limited energy using simulation tools such as TOSSIM [11], COOJA/MSPSim [6], and others [10, 14, 17]. We believe that a similar effort is essential to appropriately focus efforts of the community on energy harvesting/transfer technologies, and the resulting protocols or architectures that deal with their unique characteristics. This effort is emphasized by the high cost and complexity of deploying and calibrating EH and WET-based sensor technologies [3, 12].

We thus need to develop relevant system and environment models and incorporate them in concrete simulation tools that can help us understand the nature of different EH and WET technologies and their impact on network operation. Although fine-granular modeling of this phenomena is complicated, as the relevant environment features are difficult to ascertain, an abstract simulation model can provide useful insights. For example, in Section 4.3 we show how simulations can lead to a better choice of energy scheduling algorithms. These insights do not necessarily require detailed modeling but a high level simulation platform, with abstract models,

*Current affiliation: Department of Computer Science, University of New Mexico.

EWSN'16, February 15–17, 2016, Graz, Austria.

such as the one presented in this paper.

We make two contributions to facilitate the research in this area. **First**, we develop a generic, TOSSIM [11] based simulation framework for modeling EH and WET (Section 2). The framework allows plug-in models entirely based on the user's needs: ranging from high level abstract to fine-grained detailed models. Furthermore, they may be based on empirical data or theoretical analysis.

Second, as a proof of concept, we integrate this framework with a laser energy model gleaned from empirical data (Section 3). Using this model we show how the challenges of an intermittent energy supply mechanism, such as frequent node reboots, and their impact on network operation can be captured and investigated in simulations (Section 4).

2 Simulation Framework

There are three main reasons that motivate our choice of TOSSIM for the simulation framework. First, our goal is to facilitate early studies in high level simulations without stipulating a minimum granularity of the simulation model, which is not possible for example in instruction-set simulators. Second, the TOSSIM architecture is inherently very extensible due to the component based nesC language. Finally, because TOSSIM replaces hardware drivers with simulation wrappers providing the exact same interfaces, it enables direct deployment of the simulated application as opposed to other simulators such as ns-3.

2.1 Simulation Architecture

We embed our simulation framework in the existing TOSSIM architecture as shown in Figure 1. Besides the wireless *energy propagation* model, three new components namely *source*, *harvester*, and *energy buffer*, as well as a new event type, *energy event*, are registered. Our APIs for each of these components are highly flexible: while defining a very rich set of interfaces allowing fine grain modeling of all relevant characteristics, the APIs do not necessitate modeling of any particular characteristic. This allows the users to decide whether to wire an interface that models a certain characteristic or abstract from such details by leaving that component out of the simulation. For example, a user may want to model intrinsic details of energy losses during harvesting or simply include a static loss coefficient.

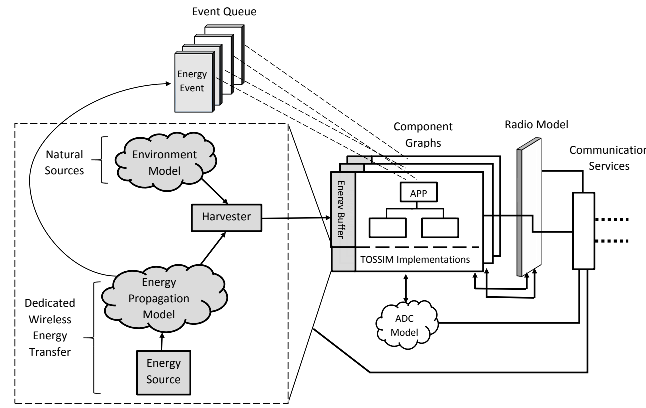


Figure 1. Extending TOSSIM architecture [11] for modeling intermittently powered embedded networks (grey): extensions include *energy source* and *harvester* components for modeling respective phenomena, wireless *energy propagation* model, and a new event type for registering energy transfer operation into simulation.

2.2 Models

In the following we discuss the need for and the desired functionality of each of the architectural components and their relevant APIs.

2.2.1 Energy Source

The source component simulates an ambient energy source, which can be natural or based on dedicated WET. To model a natural source, we need an appropriate *environment model* of ambient energy one may exploit to harvest energy. However, for a WET source we need to model specific characteristics, such as the *energy consumption* and the *efficiency* of the transmitter in converting electric energy into wireless energy. These characteristics are particularly relevant if the source itself is transiently powered and not through an uninterrupted power supply such as a wall socket. An additional aspect of wireless energy is that of *directionality*. Existing solutions may operate in a directional fashion - enjoying greater efficiency and range - as opposed to a omnidirectional mode, which facilitates deployments by alleviating the need to orient the transmitters. For directional WET, we additionally need to model the *movement control* characteristics, such as movement delay and accuracy, for orienting the transmitter towards a particular node in the network.

The main application API for a WET source is `eSend(uint16_t dest, uint32_t duration, uint8_t type)`, where *dest* is the ID of the node receiving the energy, *duration* is the length of the recharging interval, and *type* represents the energy source, e.g. light or radio. The implementation of source, and all other framework components, is further subdivided in multiple nesC components wired through interfaces. This allows including or leaving out from the model, a certain WET functionality based on a developer's needs.

We note that the main difference between a natural or a WET based energy source model is that the former is self driven, i.e., automatically inserts *energy events* in the simulation queue and does not provide any application API.

2.2.2 Energy Propagation

The energy propagation model simulates wireless energy transfer through a certain medium. The main characteristics of propagation include; *permeability*: the ability to travel through mediums of different types; *path loss*: reduction in the power density of the transmitted signal as it propagates through a medium.

The API for energy model is `transmitEnergy(uint16_t dest,uint32_t duration, uint8_t type, uint32_t delay, uint32_t energy)`, where *delay* is the movement delay of a directional WET source that needs to be added to the simulation time before inserting an event, and *energy*, for example, represents the power in watts of the transmitted signal. The environmental characteristics of a certain medium can be incorporated in the energy propagation model through a `configuration`.

Besides general permeability characteristics of the medium, the configuration also allows to input specific path characteristics, such as distance, line-of-sight, or an opaque object between nodes. This is similar to how TOSSIM uses configurations to import wireless data transmission characteristics into the simulation.

2.2.3 Harvester

The harvester component models the technical solution to convert an environmental phenomena, natural or provisioned by a WET source, into electric energy. Every source bears an intrinsic content of energy that may only be partly reaped, depending upon the conversion technique employed. Hence, the most important characteristic that needs to be modeled is the energy loss incurred during this extraction process. While modeling a certain harvesting technique, such as piezoelectric or thermoelectric, is complicated, the energy conversion efficiency of most harvesting techniques is documented and can be used to derive an abstract simulation model. In our simulation framework, the harvester model has to implement the handlers for events inserted by the energy model.

2.2.4 Energy Buffer

The energy buffer component models energy storage, such as capacitors or rechargeable batteries, at the receiving node. An energy buffer is used for storing energy to serve a node when ambient energy is insufficient, due to the limited efficiency of existing EH solutions, or unavailable. The most important characteristics for modeling are the *capacity*, the rate of *charging*, and the depth of *discharging*. While the capacity and charging behavior are specific to a certain storage technology, the discharging behavior largely depends on the external load which may vary based on the energy state of hardware components at a certain time. Thus, the accuracy of buffer discharging model is a function of the node model provided by a simulation platform.

For example, TOSSIM currently provides models of varying granularity for MICAz platform: it accurately translates hardware interrupts, such as timer fires, into simulation events but only provides an abstract, platform independent implementation of the radio model. PowerTOSSIM [14, 17] and TimeTOSSIM [10] extend it to provide detailed energy consumption models of different hardware components of the node. Although MICAz is not among the latest platforms for embedded network sensing, developing new node models is beyond the scope of this paper and is future work. Nonetheless, since our framework targets high level simulations, the need for a particular node model may only arise to capture the precise discharging behavior of the energy buffer.

In our simulation framework, a node remains active in simulation as long as the energy level of its buffer is above the minimum operational threshold of the modeled platform. Otherwise, it is shutdown until its energy level again rises above a threshold.

3 Modeling Laser based Wireless Energy

We model laser technology because it is the first published and practically demonstrated solution that can autonomously run a WSN. This facilitates its seamless integration in TOSSIM, a simulator for WSN class of devices.

Table 1. Laser components and modeling technique based on empirical data-set [3].

Components	Model Input (in evaluation)
Transmitter	laser signal energy in watts ($0.8W$)
Movement	mechanism: servo motors controlled pan/tilt
	minimum angular movement: in degrees ($0.49°$)
	movement delay: in *ms* (3.78 *ms* per $0.49°$)
Propagation	laser propagation traces from [3]
Harvester	monocrystalline solar panel
	efficiency: $\approx 20\%$
Buffer	capacitor in μF (100μF)
	charging: standard RC circuit ($E = \frac{1}{2}CV^2$)
	discharging: radio load- Idle, Tx and Rx ([14])

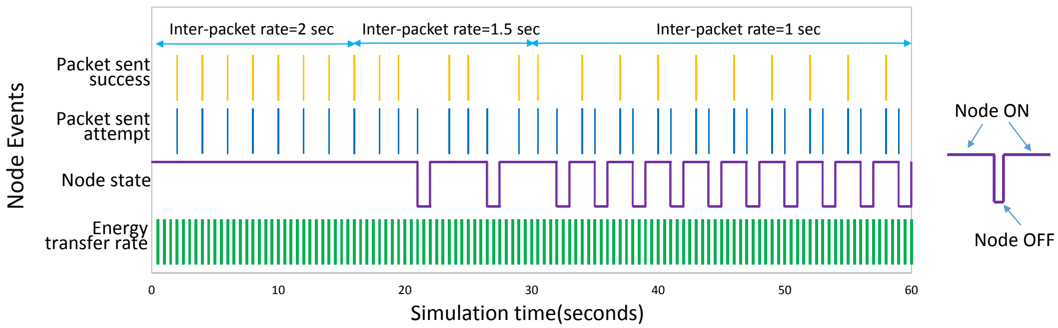
Moreover, the characteristics of laser transmission are more predictable and hence simpler to model in comparison with other technologies such as radio. This helps us keep our first model simple for a better understanding and evaluation of concepts. However, we note that our framework is generic and independent of any particular WET technology and model granularity. Other models, theoretical or from empirical data, can easily be plugged into the framework. Our model is mostly based on the traces obtained from a laser deployment [3] providing ample data to create a simulation model (summarized in Table 1).

To model the energy source, we need to model the transmitter and its movement control mechanism. For this purpose, we model a commodity laser module mounted on a pan-tilt mechanism controlled by servo motors. We model exact characteristics, such as minimum angular movement and delay, of the pan-tilt mechanism employed in [3]. The energy events (cf. Section 2) are added with the corresponding movement delays incurred while refocusing from one node to other based on the recharging schedule, which is programmed as an application software. We assume that the source is unconstrained in terms of energy, i.e., powered through a wall socket, and hence, its energy consumption can be ignored.

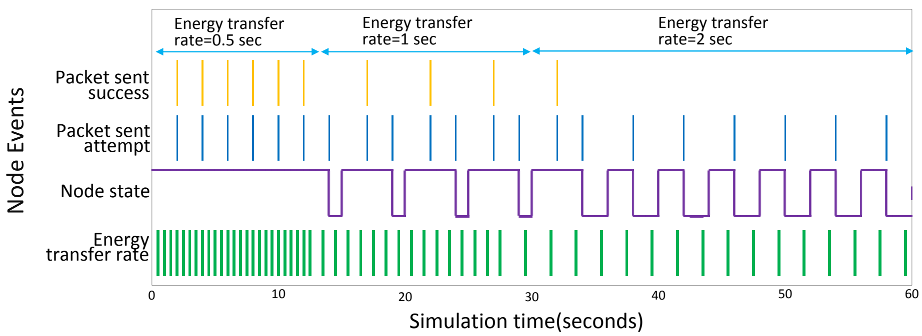
The laser source transmits a high intensity beam resulting in near uniform attenuation at a certain range. Our laser propagation model is entirely based on empirical traces [3]. The path loss (or attenuation) is dependent upon the medium characteristics and the distance between the source and the receiver, both input via a configuration file.

The harvester is perhaps the most difficult component to model as it involves complex energy conversion processes. However, barring a specific need, we can arguably abstract from intrinsic hardware details and use a static energy-loss coefficient in high level simulations. It is relatively easy to determine this coefficient for lasers, as a directional power-beam focuses on the harvester (solar panel) resulting in a steady supply of energy being harvested. The harvester model is thus based on an energy-loss coefficient, determined empirically from the amount of electrical energy that can be extracted from the beam and provided to the energy buffer.

Finally, we model the energy buffer as a capacitor. The charging rate of capacitor is faster at the start and then tapers off as the capacitor takes on additional charge at a slower rate depending upon the time constant τ. The discharging



(a) Varying packet transmission rate



(b) Varying energy transfer rate

Figure 2. Node behavior: frequent shutdowns and reboots occur either by increasing the workload or by decreasing the recharging rate.

rate depends upon the applied load, i.e., energy consumption model. For our initial evaluation, we integrate the energy consumption model of the CC2420 radio chip [14].

4 Evaluation

Since the laser model is based on traces from a practical deployment, the source, energy propagation, and harvester models are inherently accurate. However, inaccuracies, which we believe can be neglected in an initial exploratory model for high level simulations, might arise due to a lack of complete energy consumption models of the simulated node platform: we only model energy consumption of the radio hardware and abstract from other components[1], such as the processor and sensors. To minimize the impact of this abstraction, in the following evaluation we use applications which primarily use radio hardware for communication.

We evaluate three aspects: First, to establish the correctness of our implementation, we observe individual node behavior by varying workload and recharging schedules. Second, we observe network behavior by varying the number of nodes in the network, which impacts the recharging schedule, in different topologies. This leads to a better understanding of the challenges introduced by intermittent energy supply. Finally, we show how this understanding can help us develop and evaluate new algorithms. All our experiments assume a lossless wireless communication model so that we can evaluate the effects of EH and WET models only.

[1] Updating implementations of TOSSIM extensions [10, 14] is a future work.

4.1 Node Behavior

We perform two experiments to highlight node behavior when powered using a dedicated, laser-based WET. In the first experiment, we maintain a periodic energy transfer schedule while varying the workload, i.e., packet transmission rate of the node. We simulate two types of nodes: a WET source mounted with laser transmitter, also acting as the base station, and an energy receiver with a solar panel. Figure 2(a) shows the behavior of the receiver with a periodic recharging schedule of 500ms and an intermission of the same length. The packet transmission rate is gradually increased during the simulation run. With a large inter-packet interval, all packets are successfully transmitted and the receiver does not experience any shutdowns. However, the gradual increase in the packet transmission rate results in packet failures due to frequent shutdowns and reboots, a peculiarity of the intermittent energy supply. In the second experiment, the packet transmission rate is kept constant whilst the energy schedule is changed. Figure 2(b) shows a similar overall trend: packet loss due to increasing shutdown and reboots when the interval between two successive recharging epochs of the receiver is increased. We can conclude that the simulation framework reveals peculiar aspects of node behavior under a fundamentally different set of constraints due to the change in energy supply mechanism.

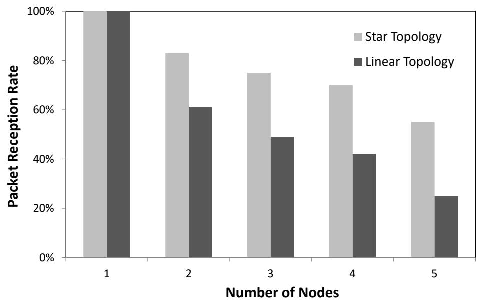


Figure 3. Network behavior: packet reception rate for linear and star topologies when more nodes are added into the simulated network.

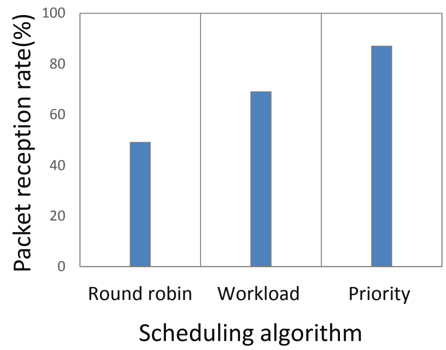


Figure 5. Multihop packet reception rates with different energy scheduling algorithms.

4.2 Network Behavior

To highlight network behavior, we simulate two types of network topologies, star and linear. The star topology represents clustered deployment with receivers directly connected to and radially spread across the energy source. The linear topology represents a collection tree based multihop communication scenario where the base station (energy source) is placed at one end of the linear topology and the data is forwarded hop-by-hop.

Figure 3 shows the packet reception rates for both topologies. The simulation starts with a single receiver and further additions are made during the simulation run. The source recharges the receivers in a round robin fashion. The addition of more receivers burdens the source, decreasing the recharging rate and increasing the intensity of shutdowns and reboots. This ultimately reduces packet reception rates at the base station. The results for the linear topology are worse as shown in Figure 3. Here, we enforce a collection tree based multihop packet forwarding in which node n forwards its packets to node $n-1$. In a round-robin energy transfer schedule, node 1 suffers with most shutdowns and reboots as it has to receive and forward more packets than any other node.

Although the behavior depicted in Figure 3 is predictable, existing simulation tools are unable to automatically generate these behaviors for evaluating algorithms before deployment.

4.3 Evaluating Algorithms

After observing network behavior and developing noteworthy insights, we now show how the simulation framework further helps us in developing and evaluating new algorithms. For example, here we try to evaluate different energy scheduling algorithms and see the overall response from the network. We evaluate three scheduling algorithms, round robin, workload based, and priority based. Figure 4, which is divided in three sections separated by vertical dotted lines, shows the results of simulating these three algorithms with the collection tree based multihop topology. The first section shows the network behavior, i.e., node 1 suffers most shutdowns, for round robin scheduling. At 2000s we dynamically activate workload based energy scheduling: the amount of energy transferred is proportionate to the work load of a node. We can clearly see the changed network response as now all the nodes enjoy a fair share of lifetime. Finally, at 4000s, we activate priority scheduling which is an enhanced version of workload based scheduling: critical node, such as node 1, is prioritized for energy as it is closest to the base station and responsible for packet delivery from multiple nodes as well as maintaining a connected network topology. Generalizing, the priority of node $n-1$ is higher than node n. We can see that the priority scheduling results in (i) nodes closer to the base station enjoying longer lifetimes as compared to the ones farther away, and (ii) reduced node shutdowns and reboots in the network. Figure 5 clearly shows the positive impact of priority scheduling on data collection performance, i.e., improved packet reception.

Overall, this evaluation demonstrates the utility of the simulation framework in revealing challenges peculiar to intermittent energy supply and developing and evaluating algorithms under a fundamentally different set of constraints. Recent related works, such as on intermittent computing [1, 15] and DTNs, can significantly benefit from such simulation tools.

5 Discussion and Related Work

The presented simulation framework that is *open* (exports interfaces for model integration), *flexible* (can support any model granularity), and *comprehensive* (includes Wireless Energy Transfer as well). This framework along with its large code-base enriches TOSSIM, undoubtedly a mainstream simulator for the vast majority of WSN community, with the necessary widgets for EH and WET. This will simplify a developer's task to merely plug-in preferred models, using the (now) well-defined interfaces and the wiring in-between, eliminating the need to grapple with complex low-level customizations of the simulation architecture.

A simple, abstract laser model is intentionally used to demonstrate the efficacy of the simulation framework. This is very much in-line with the original spirit of TOSSIM: a comprehensive radio framework built around just a couple of simplistic models. Over time researchers plugged-in models of increasing complexity for tailored simulations. System-level-accuracy of the laser model can only be validated with the inclusion of node energy consumption models, where we cannot claim novelty. Nonetheless, recognizing the importance of energy consumption models, their up-gradation and

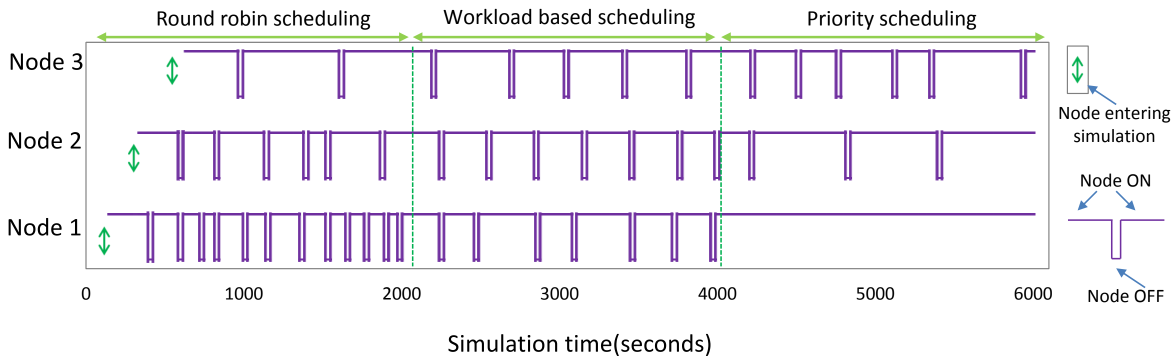


Figure 4. Impact of different energy scheduling algorithms on node and network behavior.

integration in TOSSIM is primary future work.

While there is huge body of work on simulating energy consumption in WSN, here we focus only on EH and WET: With the required support now available in TOSSIM, EH models [2, 18] for high level simulation platforms, such as ns-3 and OMNeT++, can easily be imported to an OS supported platform for simulating deployment-ready applications. SensEH [5] is the most related work as it extends COOJA, a Contiki simulator, with EH capabilities. It provides models for photovoltaic harvester evaluated using sunlight and artificial light traces from a WSN deployment inside a road tunnel. However, in contrast to our simulation framework, it does not include support for simulating WET sources. Moreover, our focus in this work has been on providing a generic model which (i) allows integration of models of varying granularity, and (ii) generates behaviors—such as computation failure, packet drops, node shutdowns and reboots—to facilitate studies evaluating the impact of intermittent energy supply on the node and network behavior.

6 Conclusion

We presented a simulation framework and integrated it with a laser model for simulating EH and WET in high level simulations. Our evaluation demonstrates the utility of this framework in providing useful insights such as the impact of an intermittent energy supply on collection routing performance. We believe that the utility of such a simulation framework goes a long way in developing algorithms for this new, fundamentally different class of embedded networks. The framework easily integrates new models providing a potential platform for the community to evaluate sophisticated models of various EH and WET technologies that they are researching. Our future work includes improving and evaluating the accuracy of existing models, developing models for other types of ambient energy sources, as well as developing and integrating energy consumption models of contemporary devices used in embedded wireless networks.

7 References

[1] D. Balsamo, A. S. Weddell, G. V. Merrett, B. M. Al-Hashimi, D. Brunelli, and L. Benini. Hibernus: Sustaining computation during intermittent supply for energy-harvesting systems. *Embedded Systems Letters*, 7(1):15–18, 2015.

[2] D. Benedetti, C. Petrioli, and D. Spenza. Greencastalia: An energy-harvesting-enabled framework for the castalia simulator. In *ENSSys '13*, 2013.

[3] N. A. Bhatti, A. A. Syed, and M. H. Alizai. Sensors with lasers: Building a wsn power grid. In *IPSN*, 2014.

[4] M. Buettner, R. Prasad, A. Sample, D. Yeager, B. Greenstein, J. R. Smith, and D. Wetherall. Rfid sensor networks with the intel wisp. In *SenSys*, 2008.

[5] R. DallOra, U. Raza, D. Brunelli, and G. Picco. Senseh: From simulation to deployment of energy harvesting wireless sensor networks. In *IEEE SenseApp*, 2014.

[6] J. Eriksson, F. Österlind, N. Finne, N. Tsiftes, A. Dunkels, T. Voigt, R. Sauter, and P. J. Marrón. Cooja/mspsim: Interoperability testing for wireless sensor networks. In *SIMUTools*, 2009.

[7] M. Gorlatova, P. R. Kinget, I. Kymissis, D. Rubenstein, X. Wang, and G. Zussman. Energy harvesting active networked tags (enhants) for ubiquitous object networking. *IEEE Wireless Commun.*, 17(6):18–25, 2010.

[8] J. Gutierrez, J. F. Villa-Medina, A. Nieto-Garibay, and M. A. Porta-Gandara. Automated Irrigation System Using a Wireless Sensor Network and GPRS Module. *IEEE Transactions on Instrumentation and Measurement*, 63:166–176, 2014.

[9] G. Hancke and N. Vorster. The feasibility of using resonant inductive power transfer to recharge wireless sensor network nodes. In *IEEE Wireless Power Transfer Conference (WPTC)*, 2014.

[10] O. Landsiedel, H. Alizai, and K. Wehrle. When timing matters: Enabling time accurate and scalable simulation of sensor network applications. In *IPSN*, 2008.

[11] P. Levis, N. Lee, M. Welsh, and D. Culler. Tossim: Accurate and scalable simulation of entire tinyos applications. In *SenSys*, 2003.

[12] V. Liu, A. Parks, V. Talla, S. Gollakota, D. Wetherall, and J. R. Smith. Ambient backscatter: Wireless communication out of thin air. In *SIGCOMM*, 2013.

[13] Y. Peng et al. Prolonging sensor network lifetime through wireless charging. In *IEEE RTSS*, 2010.

[14] E. Perla, A. O. Catháin, R. S. Carbajo, M. Huggard, and C. Mc Goldrick. Powertossim z: Realistic energy modelling for wireless sensor network environments. In *ACM PM2HW2N '08*, 2008.

[15] B. Ransford, J. Sorber, and K. Fu. Mementos: System support for long-running computation on rfid-scale devices. *SIGARCH Comput. Archit. News*, 39:159–170, 2011.

[16] L. Rizzon, M. Rossi, R. Passerone, and D. Brunelli. Wireless sensor networks for environmental monitoring powered by microprocessors heat dissipation. In *ENSSys '13*, 2013.

[17] V. Shnayder, M. Hempstead, B.-r. Chen, G. W. Allen, and M. Welsh. Simulating the power consumption of large-scale sensor network applications. In *SenSys*, 2004.

[18] C. Tapparello, H. Ayatollahi, and W. Heinzelman. Energy harvesting framework for network simulator 3 (ns-3). In *ENSsys '14*, 2014.