Energy-Harvesting Active Networked Tags (EnHANTs): Prototyping and Experimentation

ROBERT MARGOLIES, MARIA GORLATOVA, JOHN SARIK, GERALD STANJE,
JIANXUN ZHU, and PAUL MILLER, Electrical Engineering Department, Columbia University
MARCIN SZCZODRAK, Computer Science Department, Columbia University
BARADWAJ VIGRAHAM, Electrical Engineering Department, Columbia University
LUCA CARLONI, Computer Science Department, Columbia University
PETER KINGET, IOANNIS KYMISSIS, and GIL ZUSSMAN, Electrical Engineering
Department, Columbia University

This article focuses on a new type of wireless devices in the domain between RFIDs and sensor networks—Energy-Harvesting Active Networked Tags (EnHANTs). Future EnHANTs will be small, flexible, and self-powered devices that can be attached to objects that are traditionally not networked (e.g., books, furniture, toys, produce, and clothing). Therefore, they will provide the infrastructure for various tracking applications and can serve as one of the enablers for the Internet of Things. We present the design considerations for the EnHANT prototypes, developed over the past 4 years. The prototypes harvest indoor light energy using custom organic solar cells, communicate and form multihop networks using ultra-low-power Ultra-Wideband Impulse Radio (UWB-IR) transceivers, and dynamically adapt their communications and networking patterns to the energy harvesting and battery states. We describe a small-scale testbed that uniquely allows evaluating different algorithms with trace-based light energy inputs. Then, we experimentally evaluate the performance of different energy-harvesting adaptive policies with organic solar cells and UWB-IR transceivers. Finally, we discuss the lessons learned during the prototype and testbed design process.

CCS Concepts: ● **Networks** → **Network protocols;**

Additional Key Words and Phrases: Energy adaptive networking, energy harvesting, ultra-low-power communications, organic solar cells, ultra-wideband (UWB), cross-layer

1. INTRODUCTION

Energy-Harvesting Active Networked Tags (EnHANTs) will be a new type of ultra-low-power flexible devices that harvest indoor light energy and can be attached to objects

ACM Transactions on Sensor Networks, Vol. 11, No. 4, Article 62, Publication date: November 2015.
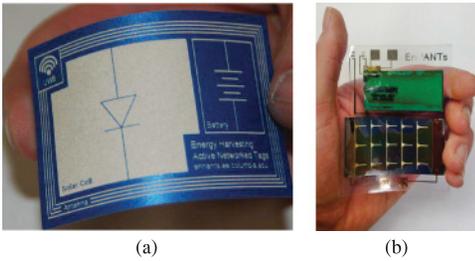
62

Fig. 1. (a) The intended EnHANT form factor, and (b) an EnHANT mock-up integrated with a flexible solar cell and a thin-film battery.
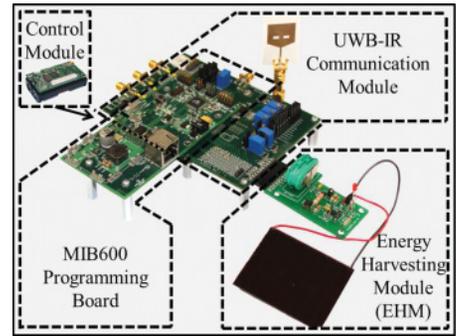

Fig. 2. An EnHANT prototype.

that are traditionally not networked. The realization of the EnHANTs vision is based on recent advances in energy harvesting [Bandyopadhyay and Chandrakasan 2012; Peumans et al. 2003; Taneja et al. 2008; Paradiso and Starner 2005], which allows for dramatic reduction in the size and weight of a wireless node battery. In particular, advances in the the area of organic semiconductors for energy harvesting allow the fabrication of organic photovoltaics[1] (OPVs) on flexible substrates [Peumans et al. 2003], thereby enabling the pervasive use of future flexible EnHANTs. Another enabling technology is ultra-low-power Ultra-Wideband Impulse-Radio (UWB-IR) communications [Wentzloff et al. 2007; Daly et al. 2010; Crepaldi et al. 2011]. UWB-IR spends substantially less energy than other low-power wireless technologies due to its pulse-based nature.

The advances in these areas will allow small, flexible, lightweight tags to communicate and network in environments with limited energy availability (e.g., while harvesting indoor light energy).[2] The form factor for the envisioned EnHANT is shown in Figure 1(a) and a recent mock-up is shown in Figure 1(b). A tag of this size (less than $25cm^2$) will support continuous data rates of nearly 3kbps [Gorlatova et al. 2013].

EnHANTs will be one of the enablers for the *Internet of Things* [Atzori et al. 2010] and will support a variety of tracking and monitoring applications beyond what the current tracking technology, RFIDs, permits. While RFIDs make it possible to *identify* an object, EnHANTs will make it possible to *search* for an object, and to continuously track objects' whereabouts, including their proximity to each other.

In Gorlatova et al. [2010a] we described a representative EnHANT application: *locating a misplaced book in a library*. In this application, EnHANTs will be attached to library books, harvest indoor light, and wirelessly exchange IDs with the neighboring books. If a book is misplaced, its ID will be significantly different from the IDs of its neighbors, and this information will be forwarded to the librarian. Similar applications were recently discussed and demonstrated in Wang and Katabi [2013] and Liu et al. [2013]. They include finding specific items in a store or a warehouse, locating misplaced items, and continuous monitoring of merchandise in transit.

Realizing the EnHANTs vision requires the collaborative design of cross-layer hardware, software, and algorithms. A few solar- (sunlight-)powered sensor networks have been deployed and tested [Yang et al. 2009; Taneja et al. 2008]. Additionally, networking protocols for energy-harvesting nodes recently started gaining attention (e.g., Fan et al.

---

[1]We use the term "photovoltaic" rather than "solar cell" to emphasize the use for harvesting *indoor light*, rather than sunlight.
[2]Indoors, the amount of energy available is a *thousand times lower* than outdoors [Gorlatova et al. 2010a].

Fig. 3. A software-based light control system, along with four EnHANT prototypes.
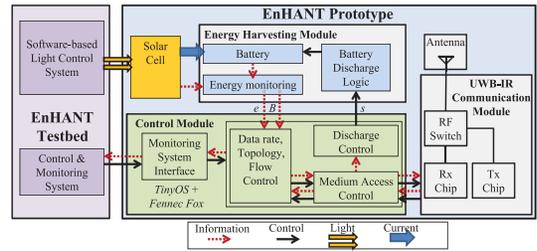

Fig. 4. A block diagram of the EnHANT prototype and its interactions with the testbed.

[2008], Liu et al. [2010], and Chen et al. [2011]). However, the challenges imposed by the cross-layer interactions between indoor energy harvesting, RF transceiver design, communications, and networking have not been studied in depth.

Developed over the past 4 years through a series of design phases, in this article we present the cross-layer design considerations of the EnHANT prototypes. The current prototypes (shown in Figure 2) are much larger than the envisioned node that appears in Figure 1(a). Yet, they already harvest indoor light energy using custom-designed OPVs (which can be made flexible) and communicate wirelessly using ultra-low-power UWB-IR transceivers. The prototypes form small networks and adapt their communications and networking patterns to the battery and energy-harvesting states using an Energy-Harvesting Module (EHM). The prototype design has been driven by the needs of the higher layers (e.g., flow control, multihopping), while the higher-layer protocols (e.g., MAC and flow control) had to adapt to the physical layer (OPV and UWB-IR) characteristics.

Moreover, we present the EnHANTs testbed,[3] which enables controllable and repeatable experiments with communications and networking algorithms for energy-harvesting nodes. The testbed allows observing the states of the prototypes in real time and includes a software-based light control system (see Figure 3) that can expose solar cells to controllable light conditions based on real-world light energy traces. To the best of our knowledge, this environmental input control system is the first of its kind. The approach to testbed design and development we pioneer in this work, as well as the design of the environmental energy inputs, is applicable not just to EnHANTs, but to all future energy-harvesting devices.

Algorithms on all layers of the protocol stack need to be redesigned in order to support networking energy-harvesting devices. However, due to space constraints, designing new energy-harvesting adaptive algorithms is out of the scope of this article. Rather, using the testbed and the prototypes, we conducted experiments regarding the performance of the algorithms, some of which were proposed in Gorlatova et al. [2013], Fan et al. [2008], and Kansal et al. [2007]. Specifically, we considered (i) algorithms that determine the energy-spending rate of a node, (ii) the energy-related tradeoffs of the UWB-IR transceivers in realistic environments, (iii) link layer algorithms that determine the data rates of communicating nodes, and (iv) flow control and collection tree selection algorithms for a small network. Our evaluation method is unique, since we provide the nodes various light energy scenarios and conduct repeatable experiments. Insights obtained in the experiments may apply beyond the EnHANTs to energy-harvesting sensor networks [Raghunathan et al. 2005; Jiang et al. 2005; Zhu et al.

---

[3]A video showing the testbed from the demonstration presented in Stanje et al. [2011] is available at youtu.be/QFCf62lBATI.

2009] and networks of computational RFIDs [Gummeson et al. 2010]. Throughout the evaluation of experiments, we demonstrate numerous phenomena which are typically difficult to capture in simulations and nearly impossible to model in analytical work.

The prototypes and the testbed were developed over the past 4 years through a series of integration phases. At the end of each integration phase, the prototype and testbed functionality were demonstrated at a conference demonstration session (e.g., Stanje et al. [2011], Zhu et al. [2011], and Gorlatova et al. [2010b, 2010c]). We summarize the lessons that were learned throughout the different phases of the prototype and testbed design process. For example, we discuss gaps between theoretical assumptions and practical constraints as well as the importance of facilitating repeatable experiments in an energy-harvesting testbed. We believe that these lessons would be useful for the designers of many different types of energy harvesters and energy-harvesting adaptive networks.

1.1. Contributions and Article Organization

To conclude, the main contributions of this article are (i) the design and development of EnHANT prototypes, (ii) the design of a testbed for the evaluation of energy-harvesting algorithms, and (iii) the experimental evaluation of such algorithms. To the best of our knowledge, the EnHANT prototypes are the first wireless devices that harvest energy using OPVs, the first to demonstrate multihop data forwarding over the ultra-low-power UWB-IR physical layer, and the first to adapt to the harvesting states in real time. We also note that this work is the first attempt to evaluate energy-harvesting adaptive policies in a controllable experimental environment.

Compared to the previous descriptions of the EnHANTs prototypes at various phases [Stanje et al. 2011; Zhu et al. 2011; Gorlatova et al. 2010b, 2010c] and a preliminary conference paper [Gorlatova et al. 2013], this article presents many important details that were not described in the past. Specifically, this article describes *detailed* design considerations of the EnHANTs prototype and testbed including descriptions of the UWB-IR Wireless Communications and the Energy-Harvesting Module. Additionally, this article describes experimentation with energy-harvesting adaptive flow control and collection tree selection functionality. Finally, this article summarizes the lessons learned through the development of the prototypes and the testbed.

The article is organized as follows. In Section 2, we discuss the related work. Sections 3 presents an overview of the prototypes and the testbed. Section 4 introduces the notation and the experimental settings. Section 5 discusses the OPVs, the EHM, and energy allocation policies for energy-harvesting nodes. Section 6 discusses the UWB-IR transceiver, the MAC layer, and the link layer energy-harvesting adaptive policies. Section 7 discusses energy-harvesting adaptive policies for networks of energy-harvesting nodes. Section 8 presents the lessons learned, and Section 9 presents ongoing and future work. We conclude the article in Section 10.

2. RELATED WORK

Combining the advances in energy harvesting and ultra-low-power communications has attracted attention from industry [EnOcean Alliance 2014; Texas Instruments 2012] and academia. Energy harvesting can be used to power sensor nodes [Texas Instruments 2012; Taneja et al. 2008], computational RFIDs [Gummeson et al. 2010; Zhang and Ganesan 2014], and to realize mm^3-scale wireless devices [Dutta 2014]. There are also energy-harvesting sensors [EnOcean Alliance 2014]; for example, NFC-WISPs [Dementyev et al. 2013] harvest RF-energy from a nearby smartphone to power an e-display. To the best of our knowledge, all of the prior platforms operate as part of a single-hop network.

The prototypes are integrated with a UWB-IR communication module that supports higher-layer MAC and networking functionalities. An alternative approach for device communications is explored in Liu et al. [2013]. There have been other low-power UWB-IR implementations [Wentzloff et al. 2007; Daly et al. 2010; Solda et al. 2011; Thoppay et al. 2008; IMEC 2013]. A few UWB-IR product development efforts target higher data rate applications such as wireless microphones [Audio-technica 2012], real-time location systems [Zebra Technologies 2013], and radar [Time Domain 2013]. However, to the best of our knowledge, none have implemented a network of more than two nodes, nor has any experimental MAC or higher-layer communications been reported.

MAC protocols for UWB-IR based on a time-hopping physical layer (which differs from our transciever) can be found in Win and Scholtz [2000], Le Boudec et al. [2004], Di Benedetto et al. [2006], Cuomo et al. [2002], and Karapistoli et al. [2012]. However, to the best of our knowledge, these MAC protocols have not been experimentally evaluated. For each of these protocols, carrier sensing, which is difficult to implement due to the low transmit power and the lack of a carrier waveform for UWB-IR communications [Zhang et al. 2009; August et al. 2004], is not utilized. This complicates the implementation of even simple CSMA schemes. Hence, the 802.15.4.a standard [IEEE 2007] provides a UWB-IR physical layer and an Aloha MAC layer with optional carrier sensing based on preamble detection [Miscopein and Gorce 2009].

The EnHANT prototypes include an EHM, which provides real-time energy awareness, and OPVs, which are specifically designed to harvest energy from low-intensity artificial light sources[Yang et al. 2013]. Existing sensor network nodes that harvest energy from sunlight [Kansal et al. 2007; Taneja et al. 2008; Yang et al. 2009] and indoor light [Yerva et al. 2012] typically use monocrystalline or amorphous silicon solar cells, rather than OPVs. Similar systems [Powercast 2014; IPS 2014; Raghunathan et al. 2005] offer only limited energy awareness and do not provide real-time harvesting rate information. Accurately estimating the state-of-charge (SOC) for the battery is essential for our application. Typically this is estimated by applying a model to measurements of the battery voltage, current, and temperature [Castagnetti et al. 2012; Park et al. 2005; Pop et al. 2005; Rao et al. 2003]. However, these models have an inherent tradeoff between SOC accuracy and power consumption [Buchli et al. 2013]. The EHM uses a Coulomb counter that provides a high-accuracy estimate of the battery state of charge. Other solar energy harvesters (e.g., Texas Instruments [2012]) cannot accurately track the incoming energy rate, nor the battery SOC.

Using the prototypes and the testbed, we experiment with energy-harvesting adaptive policies [Fan et al. 2008; Gorlatova et al. 2013; Liu et al. 2010; Noh et al. 2009; Kansal et al. 2007]. Many such policies were proposed for different layers of the networking stack [Fan et al. 2008; Noh et al. 2009; Liu et al. 2010; Chen et al. 2011; Zhu et al. 2009; Yang et al. 2009; Michelusi et al. 2012; Tutuncuoglu et al. 2013]. However, the majority of the proposed policies were only evaluated via simulations.

While several wireless network testbeds exist (e.g., Yang et al. [2009], Taneja et al. [2008], and Dutta et al. [2006]) and a few are under development (e.g., Johansen et al. [2011] and Dang et al. [2011]), to the best of our knowledge, our testbed is the first to support trace-based *light energy control* functionality. Solar simulators (used for testing solar cells) can provide precisely controlled illumination (approximating sunlight) but cannot create trace-based dynamic light environments.

3. PROTOTYPE AND TESTBED

In this section, we present an overview of the prototypes and the testbed used throughout the article for all experiments. Note that the version presented below corresponds
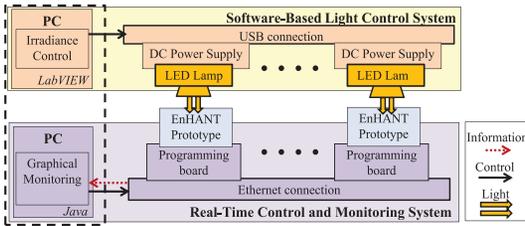
Fig. 5. A schematic diagram of the EnHANTs testbed.

Table I. The Hardware Configuration for the Light Control System

Control	Agilent E3631A DC power supply with LabVIEW-controlled LED current
Lamps	Engin LZ4-00CW00 component LEDs
Enclosure and Mounting	Custom dark box with 3D printed solar cell mounting fixtures

to Phase V of the prototype and the testbed, and we will describe its evolution in Section 8.1.

3.1. Prototype

The EnHANT prototype is shown in Figure 2 and its block diagram, including the different modules and their interactions, is shown in Figure 4. The current prototypes are larger than the envisioned nodes that appear in Figure 1, yet they are already integrated with the enabling technologies.

The prototype's Energy-Harvesting Module (EHM) contains a rechargeable battery and energy-monitoring circuitry. It interfaces with the solar cells, which are specifically designed to harvest *indoor light energy*. The current EHM is not powering the EnHANT prototype. The harvested energy is stored in the EHM's battery, and the energy spending on communications and other activities is emulated by discharging a load.

The prototypes communicate with each other wirelessly using ultra-low-power Ultra-Wideband Impulse-Radio (UWB-IR) Communication Modules, based on UWB-IR transmitter and receiver chips (described in Crepaldi et al. [2011]). The custom chips are mounted onto a printed circuit board that interfaces with the other prototype components. A Complex Programmable Logic Device is used to realize the "glue logic" between the radio chipset and the rest of the prototype.[4]

The prototype Control Module (CM) is based on a legacy off-the-shelf MICA2 mote that runs TinyOS with an added Fennec Fox software framework [Szczodrak and Carloni 2011; Szczodrak et al. 2013]. The CM implements the MAC and forwarding protocols tailored for the UWB-IR transceivers. The Communication Module is integrated with the CM such that the packets, which contain node ID and network control information, originate in the TinyOS application layer and are sent *wirelessly via the UWB-IR transceiver*. The CM adapts the prototype's networking and communication patterns, based on the energy states reported by the EHM.

3.2. Testbed

Our small-scale testbed, shown schematically in Figure 5, includes *6 EnHANT* prototypes, a *control and monitoring* system, and a *software-based light control system*. For control and monitoring, the prototypes are placed on MIB600 programming boards and accessed from a PC via Ethernet. On the PC, a Java-based graphical monitoring

---

[4]We note that while the design of the UWB-IR transceiver chips were described in Crepaldi et al. [2011], the integration of the transceiver with the higher layers, and evaluations in realistic environments (sending data packets, multiuser interference) have not been presented before.
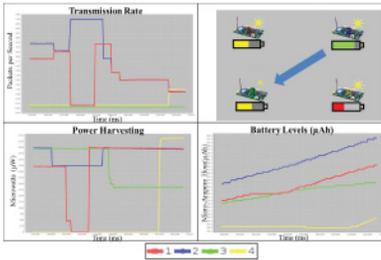
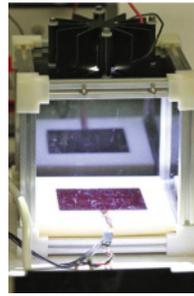Fig. 6. A screenshot of the EnHANT testbed monitoring system.

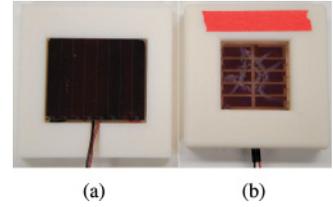Fig. 7. A dark box enclosure used in the light control system.

Fig. 8. Solar cells integrated with EnHANT prototypes: (a) an amorphous silicon (a-Si) solar cell, and (b) a custom-fabricated organic photovoltaic (OPV).

system shows in real time the network topology, data rates, energy harvested, battery levels, and the individual packets transmitted (they are shown as flashing "arrows"). A screenshot of the monitoring system is shown in Figure 6.

The software-based light control system allows exposing individual prototypes to repeatable light energy conditions *based on real-world irradiance (light energy) traces*. The system is shown in Figure 3, and the hardware configuration for the system is shown in Table I. To ensure full control over the photovoltaics' orientation and to eliminate external light sources, the photovoltaics are placed inside custom-designed dark box enclosures, shown in Figure 7.[5] The light sources are component cool white LEDs, mounted on heat sinks and attached to the enclosures. A LabVIEW script on a PC controls the irradiance inside each enclosure by controlling the current supplied to the LEDs from a DC power supply. The irradiance produced by the system was calibrated using a NIST-traceable photodiode (Newport UV-818). The system can produce over 3,000 distinct irradiance levels between 0 and 14mW/cm^2 (an effective resolution of less than 5μW/cm^2); the irradiance levels can be changed with time steps of under 0.1sec.

This system allows replicating real-world irradiance[6] traces with remarkable repeatability. For example, energy-harvesting rates over 4–10 repetitions of a light energy trace are shown in Figures 11(b), 15(a), 16(b), and 17(b), with variations in energy harvested less than 1.9%, 6.4%, 4.1%, and 3%, respectively. This ensures that experimental evaluations of energy-harvesting adaptive policies are *based on the same energy inputs*.

4. NOTATION AND EXPERIMENTAL SETTINGS

The notation used in following sections is summarized below. We assume that time is slotted, and we denote the time slot index by i. We denote the energy-harvesting rate of node u in time slot i by $e_u(i)$, the desired energy-spending rate by $s'_u(i)$, the actual energy-spending rate by $s_u(i)$, and the data rate by $r_u(i)$. We denote the battery charge level of node u in slot i by $B_u(i)$. Prior to the experiments, the battery is conditioned to an initial charge point using a multimeter and a power supply. Throughout this article,

---

[5]In practice, photovoltaics are effectively a connected series of modules; if a photovoltaic is partially covered, one of the modules will be darkened and the cell will produce significantly less power.

[6]Additionally, the system is capable of mapping real-world energy-harvesting measurements from other sources (e.g., vibration [Gorlatova et al. 2015]) to equivalent irradiance values to enable experiments based on other energy-harvesting sources.
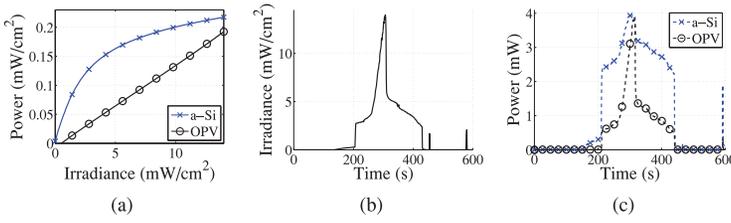
Fig. 9. Energy harvesting for a prototype with an a-Si solar cell and an organic photovoltaic (OPV): (a) generated power as a function of irradiance (light intensity), (b) time-varying irradiance based on a light energy trace from Gorlatova et al. [2011], (c) power harvested by an a-Si solar cell and an OPV.

$B_u(i)$ is reported in Coulombs.[7] In determining node data rates, we consider the energy spent on transmitting and receiving a bit, denoted by c_{tx} and c_{rx}, respectively. The c_{tx} and c_{rx} for the current prototype transceiver/MAC are discussed in Section 6.3.

We experimentally evaluate several energy-harvesting adaptive policies. For these evaluations, we provide the prototypes light inputs based on indoor light energy traces available in [Gorlatova et al. 2011] using the trace identifiers specified therein (e.g., L-1, L-2). The light control system allows "replaying" light energy traces at different time and light intensity scales. In the experimental evaluations, we "compress" traces in time, downsampling original day-long traces to "play" in 6–12 minutes. To capture the corresponding dynamics in energy storage behavior, we additionally scale the traces by a factor indicated in the corresponding sections. The scale-factor is chosen such that the irradiance of the light trace falls within the range of the light control system of 0–14mW/cm^2.[8] In some sections, we compare experimental results to simulations, conducted in MATLAB.

5. ENERGY HARVESTING

In the subsections below, we briefly describe the photovoltaics and the EHM, and then evaluate several energy allocation policies for energy-harvesting nodes.

5.1. Photovoltaics

The prototypes are integrated with two different photovoltaic technologies (shown in Figure 8): commercially available amorphous silicon (a-Si) cells, and *custom-designed polymer OPVs*.[9] The a-Si cells are commonly used for indoor harvesting applications [Yerva et al. 2012; Texas Instruments 2012]. This is due to their higher efficiency over the conventional monocrystalline silicon cells under the reduced intensity and blue-shifted spectrum of indoor fluorescent or LED lighting. The OPVs have energy-harvesting characteristics that are similar to those of the a-Si cells under indoor light, and they can be manufactured on *flexible substrates* using low temperature, roll-to-roll processing techniques.

Different types of photovoltaics perform differently under the same light conditions. For example, Figure 9(a) shows the power generated by two different photovoltaics: an a-Si cell and an OPV (both biased at 3.1V, the minimum voltage required to charge the

---

[7]Battery levels and battery capacities are often reported in various different units, units of charge (Coulombs, Ampere-hours), and units of energy (Joules). Since $1C = 1A \cdot 1sec$, the conversion between Coulombs and Ampere-hours is straightforward. To calculate the energy, the charge is multiplied by the voltage of the battery, V_{BAT}. In our system, $V_{BAT} = 2.4V$; thus, the charge of 1C (or, equivalently, $278\mu Ah$) corresponds to 2.4J of energy.

[8]In some cases, we do not scale the irradiance traces to the maximum light level of 14mW/cm^2.

[9]The a-Si cell is Sanyo AM-1815 with a 5.61x4.52cm^2 active area. The OPVs area is 5.0x5.0cm^2. A description of the process we followed to fabricate the OPVs can be found in PreparingPlexcore [2014].

Table II. EHM Configuration

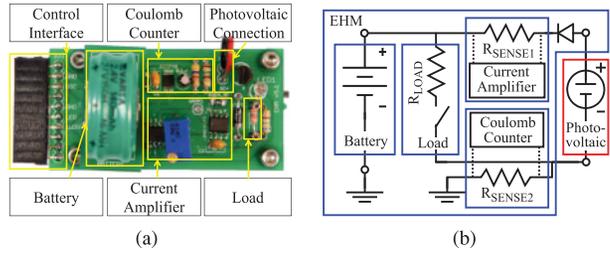| R_{LOAD}	220Ω
R_{SENSE1}	10Ω
R_{SENSE2}	10Ω
τ	20ms
Max. avg. current	5.12mA
Resolution, instant. current	625nA
Resolution, battery charge	<5mC

Fig. 10. (a) A photo of the Energy-Harvesting Module (EHM) and (b) its block diagram.

prototype's battery), under different irradiance (light intensity) levels. Using the EHM (described below), we measure the power produced by the cells. Correspondingly, using the software-controlled lights, we can estimate the energy incident on the solar cells. The *harvesting efficiency* is defined as the ratio between the two. Hence, we obtain the harvesting efficiency of the OPVs is 1%, while the harvesting efficiency of a-Si cells varies between 1% and 3% depending on the irradiance (see the measurement results in Figure 9(a)). Correspondingly, as shown in Figure 9(c), when we expose the two photovoltaics to irradiance levels based on a light energy trace recorded over a day in L-1 (Figure 9(b)), *the "curves" of the power generated by the two photovoltaics have different shapes*. We note that these effects are difficult to capture in simulations (e.g., Fan et al. [2008], Gorlatova et al. [2013], and Huang and Neely [2011]), which simply assume that the energy harvested by a solar cell is a linear function of the irradiance.

5.2. Energy-Harvesting Module (EHM)

The EHM stores the energy harvested by the photovoltaic in a rechargeable battery. The EHM is designed such that the photovoltaic or the battery can be easily added or removed; this feature is used to condition the batteries and to interchange OPVs with a-Si cells.[10] A block diagram and a photo of the EHM are shown in Figure 10. The EHM's interactions with the other prototype components are shown in Figure 4. The hardware parameters for the EHM are summarized in Table II.

For energy storage, the EHM uses a 2.4V NiMH battery with 150mAh (540C, 1.3kJ) capacity. The battery is intentionally oversized (for the envisioned EnHANTs the battery capacity will be on the order of 1–2C) to allow emulating a wide range of different battery capacities in software. The EHM monitors, in near real time, the battery level, $B(i)$, and the energy-harvesting rate, $e(i)$, and reports them to the Control Module. To track $B(i)$, the EHM's energy-monitoring circuitry uses a Maxim DS2740 *Coulomb counter*, which measures and integrates the bidirectional current across R_{SENSE2}, hence tracking the changes to the battery level in Coulombs. To track $e(i)$, the EHM uses a Maxim MAX4173 high-side current sense amplifier, which measures the instantaneous current across R_{SENSE1}. The Coulomb counter updates the battery level every 0.875s, and the battery level resolution is under 5mC.

The EHM does not supply energy to the other EnHANT prototype components. Rather, as shown schematically in Figure 4, the EHM implements controlled energy-spending functionality that corresponds to transceiver energy spending on transmitting

---

[10]The harvesting point of the photovoltaic is fixed to 3.1V, which is the minimum voltage required to charge the prototype's battery. No power point tracking techniques (i.e., MPPT) are used. With this simple design, we minimize inefficiencies and enable a seamless interchange between the two employed photovoltaics (a-Si and OPVs).

and receiving packets.[11] The Control Module signals to the EHM to activate a small load, which spends energy at a requested rate $s(i)$. Removing the constraint of running the prototype using harvested energy allows us to experiment with various hardware and protocol configurations. In the EHM, the energy is spent by discharging the battery through the load resistor R_{LOAD} (see Figure 10) for τ ms. Each load activation reduces the battery by $\Delta B = \tau \cdot V_{\text{BAT}}/(R_{\text{LOAD}} + R_{\text{SENSE1}}) = 208.68\mu\text{C}$. To verify the precision of the EHM's controllable energy spending, we compared the energy-spending rates calculated according to this formula, and the energy-spending rates we experimentally obtained, for a set of EHM load activation rates. For up to 12 load activations per second, the discrepancy was under 2.1%.

In a typical environment, an indoor energy-harvesting node would have access to 1–12 J/cm^2/day [Gorlatova et al. 2013]; therefore, a device with a 5×5cm solar cell of 1% efficiency will harvest 0.25–3 J/day (corresponding to an average harvested power of $2.89\mu\text{W}$–$34.7\mu\text{W}$). Thus, the currently used battery can be used to emulate energy storage sizes up to 432x incoming energy.

In this work, we only consider energy harvesting from light, as it is convenient to control, measure, and reproduce for experiments. However, we note that in practice, other sources may also be available in indoor environments (e.g., ambient RF [Liu et al. 2013] and kinetic [Gorlatova et al. 2015]). Although not directly applicable to other energy-harvesting sources, our EHM design can be used as a baseline for other energy-harvesting sources.

5.3. Node Energy Allocation Policies

Many different *energy allocation policies* for energy-harvesting nodes have been recently proposed [Fan et al. 2008; Kansal et al. 2007; Noh et al. 2009]. We evaluate a set of simple policies that aim to achieve *energy neutrality*—full, yet not excessive, spending of the energy harvested by the nodes. These policies, briefly described below, closely match node energy-spending rates, $s(i)$, to node energy-harvesting rates, $e(i)$. In the following sections, we use these policies as building blocks for wider-scale (i.e., link and network) energy-harvesting adaptive policies.

Exponential Policies (EX): The desired energy-spending rate $s'(i)$ is set to the *exponential average* of the energy-harvesting rate: $s'(i) \leftarrow \widehat{e}(i) = \alpha \cdot \widehat{e}(i-1) + (1-\alpha) \cdot e(i)$, $0 \leq \alpha \leq 1$. Similar policies were evaluated, via simulations, in Liu et al. [2010].

Energy Profile-Based Policies (EP-K): A node creates an *energy profile* $\{\overline{e}(1), \ldots, \overline{e}(K)\}$ by determining its *expected harvesting rate*, $\overline{e}(i)$, for different time intervals i. Energy profiles are used as inputs to many proposed policies [Kansal et al. 2007; Gorlatova et al. 2013; Fan et al. 2008; Noh et al. 2009; Chen et al. 2011; Liu et al. 2010]. In the EP-K policies (K corresponds to the number of time intervals), the node's desired energy-spending rates are set to the expected energy-harvesting rates: $s'(i) \leftarrow \overline{e}(i) \ \forall i \in K$. For example, the EP-1 policy (examined, via simulations, in Gorlatova et al. [2013] and Fan et al. [2008]), corresponds to a node spending energy at its *average expected harvesting rate* over the entire planning horizon.

We conducted extensive experiments with different EX and EP-K policies, providing nodes, equipped with either an a-Si solar cell or an OPV, dynamic energy trace-based light energy inputs. Examples of energy-spending rates, recorded in prototypes running these policies, are shown in Figure 11(a). For these experiments, we provided the prototype with the light input shown in Figure 11(c) corresponding to the light energy recorded over a day in L-3, compressed to 321s and scaled by 2.1x. The

---

[11]In the envisioned EnHANT, the transceiver power consumption is the main source of energy consumed, and hence, the energy consumption from computation is ignored.
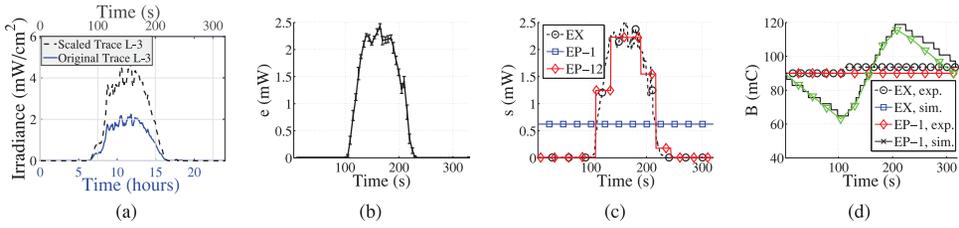
Fig. 11. Node energy allocation policies: (a) the day-long irradiance trace from L-3 [Gorlatova et al. 2011] compressed to 321s and scaled by 2.1x, (b) energy-harvesting rates from four experiment repetitions with error-bars, (c) energy-spending rates under the EX, EP-1, and EP-12 policies, and (d) battery levels, experimentally measured and simulated, under the EX and EP-1 policies.

prototype was equipped with an a-Si solar cell. The energy-spending rates are obtained for node energy-harvesting rates illustrated in Figure 11(b), where error bars represent variations in energy-harvesting rates in different experiments (the variability in total energy harvested was *under 1.9%*).

The experimental results and the results obtained via MATLAB simulations closely match. For example, for the policies shown in Figure 11(d), the largest mismatch between the experimentally recorded and simulated battery levels was 5mC (one battery resolution level of the EHM). This confirms the reliability and precision of the energy state monitoring and controlled energy-spending functionalities of the EHM.

The EX and EP-*K* policies effectively ensure energy neutrality. For example, in all our experiments with the EX and EP-*K* policies corresponding to node energy-harvesting rates shown in Figure 11(b), the node used 95%–96% of the harvested energy. To the best of our knowledge, achieving such a level of energy neutrality is not feasible with currently available COTS devices (e.g., Texas Instruments [2012]). However, as we will demonstrate in subsequent sections, while these simple policies are effective "in isolation," they do not take into account considerations important for *networked* nodes (e.g., that the nodes communicate, and thus may spend energy, in all time slots, including those with $s'(i) = 0$).

6. UWB-IR COMMUNICATIONS, MEDIUM ACCESS CONTROL, AND LINK LAYER

The EnHANT prototypes communicate using ultra-low-power Ultra-Wideband Impulse Radio (UWB-IR) transceivers. The UWB-IR techniques used in the prototypes are fundamentally different from traditional narrow-band techniques. In UWB-IR, information is transmitted using very short *pulses,* and most of the transceiver circuitry can be shut down between the pulses, resulting in significant power savings [Vigraham and Kinget 2014]. Yet, pulse-based communications pose many challenges. Clear Channel Assessment (CCA) functionality, for example, is not straightforward in UWB-IR. This is because, in UWB-IR there is no continuous carrier waveform to sense and the low transmit power complicates the discerning of a busy channel from narrowband interference [Zhang et al. 2009]. Additionally, in UWB-IR, the energy required to receive a bit is an order of magnitude higher than the energy required to transmit a bit [Crepaldi et al. 2011; Wentzloff et al. 2007].

6.1. UWB-IR Chipset and EnHANT Prototype Integration

The UWB-IR transceiver [Crepaldi et al. 2011] operates in a 500MHz band around a center frequency of 3.8GHz. The modulation scheme is Synchronized On-Off Keying (S-OOK), shown in Figure 12, which permits low-power timing acquisition. An S-OOK symbol consists of a synchronization pulse, which is always present, and a data pulse, which is only present for a "1" bit. These pulses are separated by a time interval
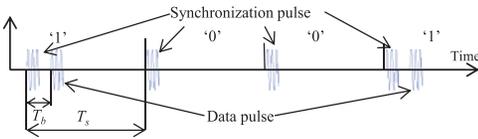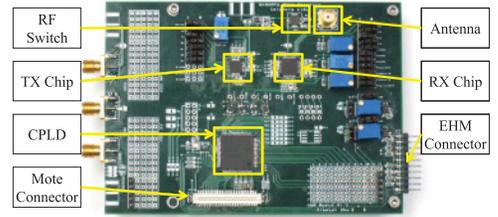
Fig. 12. S-OOK modulation and parameters.

Fig. 13. The UWB-IR Communication Module.

T_b. Adjacent symbols are separated by a time interval T_s. Targeting low data rate applications, we currently use $T_b = 1\mu s$, $T_s = 54\mu s$, for a transmission rate of 18.5kbps.

The UWB-IR Communication Module, shown in Figure 2 and represented schematically in Figure 4 (as part of the EnHANT prototype diagram), is implemented on a dedicated printed circuit board (see Figure 13). The board integrates the transceiver chipset and a 1.5cm×1.2cm omnidirectional UWB antenna [Low et al. 2005] (shown in Figure 2) with the EnHANT prototype. A wideband RF switch alternately connects the transmitter and receiver to the antenna port, enabling half duplex communication.

To integrate the UWB-IR transceiver with the software protocol stack, we developed a device driver under the TinyOS and the Fennec Fox framework [Szczodrak and Carloni 2011]. This framework provides a layered modular programming environment, enabling separate interchanging of software modules at different protocol stack levels, and thus allowing operating over a custom physical layer. The driver provides an abstract interface to the MAC layer, and functionalities such as transmit/receive mode switch, packet serialization, buffering, and Clear Channel Assessment (CCA).

Previous work on MAC for UWB-IR assumes that performing CCA is not possible. Yet, for the non–time-hopping UWB-IR communications used in our prototypes, we managed to implement CCA using a simple threshold-based energy detector.[12] The receiver chip listens for a pulse for $2 \cdot T_b$; if no pulse is demodulated, the channel is declared idle. Energy spikes from interfering systems [August et al. 2007] may cause CCA to falsely report a busy channel. However, for our purposes, these CCA "false alarms" do little to affect our application, which can wait until the noise spikes dissipate to transmit a packet. In future work, we can further reduce this interference by adding a dedicated UWB filter that rejects energy from known frequency bands (i.e., cellular, WiFi).

6.2. Single Link Experiments

Prior to performing MAC experiments, we extensively characterized the UWB-IR communication module's sensitivity to narrowband interference, transmission-rate limits, and packet-error rates under different conditions.

The UWB-IR frequency spectrum overlaps many frequency bands from narrowband communications systems including, for example, cellular and WiFi frequencies. Therefore, we tested the transceiver's vulnerability to a given narrowband interfering system by placing a smartphone at varying distances from the receiver and generating a 5-minute call at each distance. The packet error rates are shown in Table III. The results indicate that our transceiver is susceptible to narrowband interference at

---

[12]In *time hopping UWB-IR* [Win and Scholtz 2000], pulses are transmitted at different time slots according to a time hopping code. Due to the low required data rates of our applications, our UWB-IR transceiver foregoes the added complexity of time-hopping, and as a result, only one node accesses the channel at a time. On the other hand, this enables the implementation of the described CCA scheme.

Table III. The Packet Error Rates When a Cell Phone Placed
Calls at Varying Distances from the Receiver

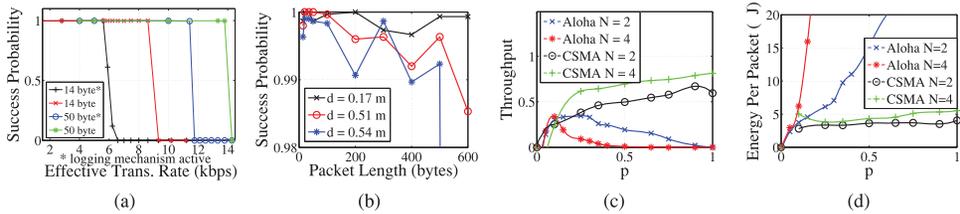| Cell Phone Distance	0m	0.25m	0.50m	0.75m
Packet Error (%)	26.34%	23.64%	3.44%	0.44%

Fig. 14.   (a) Packet success probability as a function of packet rate for 14- and 50-byte packets with logging enabled and disabled, (b) packet success probability as a function of packet length and distance between nodes, (c) average throughput as a function of p, the transmit probability for Aloha and CSMA (100% throughput is 18.5kbps), and (d) average energy spent per successful packet transmission, as a function of p.

distances less than 0.5m, which is sufficient for our experiments. Cellular interference is a representative example of narrowband systems in our testing environment, and therefore, we generalize these results and conclude that the UWB-IR transceiver and the CCA scheme operate uninhibited when interfering transmitters are kept at least a couple of meters away.

While the transceiver is capable of a transmission rate up to 18.5kbps, computational constraints limit the effective transmission rate when sending packetized data.[13] Note that the computational constraints are not fundamental but are a limitation of our current prototype implementation and microcontroller. We, therefore, conducted experiments to characterize the effective transmission rate limit for a single link. In a first set of experiments, 3,000 packets were sent over a link in one direction, for varying packet rates and a constant packet size.[14] In each experiment (i.e., for each packet size and rate), the receiver reported how many packets were correctly received. In a follow-up experiment, the receiver logged information on packet arrival time by sending data via Ethernet to the prototype monitoring system.[15] The results of these experiments are shown in Figure 14(a). For a fixed packet size of 14 bytes, a 99% packet success rate was achieved for effective transmission rates less than 6.2kbps and 8kbps with and without logging, respectively. After exceeding these rates, packets always fail. This is due to interrupt overhead incurred mainly from packet transmission and reception. Increasing the packet length to 50 bytes enables higher transmission rates. This is due to the lower computational load resulting from additional time between successive packet reception interrupts. In the following sections, by using a 50-byte packet with effective transmission rates not exceeding 5kbps, we minimize the packet errors caused by computational overloading of the system.

We characterized the effect of varying the packet length and distance between nodes on the packet error rate. The results, shown in Figure 14(b) indicate that the nodes achieve a very low packet error rate at short distances even for long packets. At the maximum packet length of 620 bytes, the effect of the added distances is observed.

---

[13]We convert to effective transmission rate = (packet length)*(packet rate).

[14]Unless otherwise stated, nodes were directly adjacent to each other at a distance of 0.17m.

[15]Sending information through the Ethernet port is imperative for many of our experimental results. Therefore, it is important to characterize the system performance with this constraint.

Therefore, for all experiments reported below, we minimize the packet error rates by setting the distance between nodes to 0.17m.

6.3. Medium Access Control (MAC) Evaluation

On top of the UWB-IR physical layer, we implemented two basic MAC protocols: p-persistent CSMA (enabled by our implementation of CCA) and the slotted Aloha.[16] To the best of our knowledge, while more sophisticated protocols have been theoretically analyzed, this is *the first attempt to implement and experimentally demonstrate MAC protocols over a UWB-IR physical layer*.

CCA requires listening to the channel, which is a relatively energetically expensive operation in UWB-IR. Thus, we conducted experiments to analyze the tradeoff between the added energy consumption of CSMA and the increased throughput.

For both MAC protocols, Figures 14(c) and 14(d) show experimentally obtained throughput and energy-spending values, correspondingly. The evaluations were conducted in networks of 2 and 4 "infinitely backlogged" (i.e., sending as often as possible) prototypes. Energy spending per successful packet, shown in Figure 14(d), was calculated based on transceiver energy consumption data provided in Crepaldi et al. [2011] (0.31mW transmitting, 3.9mW idle listening). Clearly, the Aloha throughput "curves," shown in Figure 14(c), bear strong resemblance to classical analytical Aloha results.

Expectedly, in terms of throughput, CSMA outperforms Aloha. Additionally, despite the added cost of CCA, the collision avoidance provided by CSMA reduces the average energetic cost per successful packet significantly as compared to Aloha. Expectedly, CSMA throughput increases as p increases. Yet, CSMA throughput is lower than the theoretical values, due to a delay between sensing a free channel and sending a packet, and due to delays imposed by the prototype microcontroller (whose resources are occupied by other processes, i.e., energy monitoring). In all subsequently reported experiments, the prototypes use the *1-persistent CSMA MAC*.

6.4. Energy Costs

To demonstrate energy adaptive networking, we must first quantify c_{tx} and c_{rx}, the expected energy spent (at the MAC and physical layers) to send and receive a bit, respectively. Using 1-persistent CSMA, c_{tx} and c_{rx} will increase as a function of channel utilization. Yet, with some approximations, we estimate c_{tx} and c_{rx} for a 50-byte packet using power consumption data from Crepaldi et al. [2011] as follows. Transmitting a packet consumes 6.7μJ. An additional 0.41μJ is consumed per CCA. If perfectly synchronized, the receiver would consume 31.3μJ per packet reception. However, the receiver must idly listen for the start of the packet, which will vary based on clock drift. The current prototypes rely on the microcontrollers of MICA2 motes (ATmega 128L AVR); their internal clock drift is rated at 3%. Assuming a packet transmission every 200ms, we set the guard time to $\pm$6ms, which results in 45.6μJ per guard time. Further, the receiver must listen for 1 guard time around every possible CCA slot until it hears the packet. We, therefore, limit the sender to performing 1 CCA, and if the channel is busy, drop the packet. This bounds the maximum sending and receiving per packet at 7.1μJ and 76.9μJ, respectively, and we therefore estimate $c_{\text{tx}} = 17.8$nJ/bit and $c_{\text{rx}} = 179.8$nJ/bit.

Due to our conservative estimations regarding the additional overheads, these c_{tx} and c_{rx} values are 1 or 2 orders of magnitude higher than the chipset energy consumption (1nJ/bit transmitting and 2–3nJ/bit receiving [Crepaldi et al. 2011]); however, even under the conservative assumptions, the energy consumption is lower than that of

---

[16]In Aloha, given a packet, a node transmits in a slot with probability p. In p-persistent CSMA, with probability p, a node senses the channel, and transmits if the channel is determined to be idle.
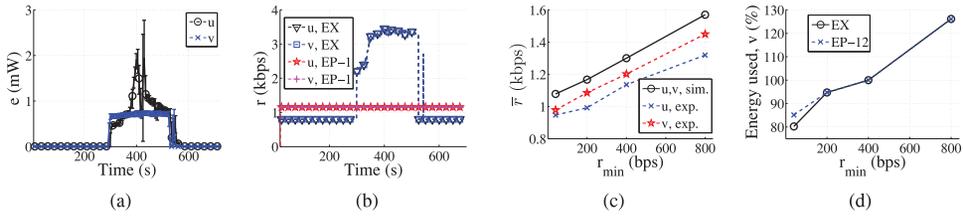
Fig. 15. Decoupled Rate Control (DRC) link policies: Using a day-long irradiance trace compressed to 700s and scaled by 20.0x from L-2, L-3, and the corresponding (a) energy-harvesting rates from 10 experiment repetitions with error bars, (b) data rates under the EX and EP-1 policies, (c) average data rates under the EP-12 policy, experimentally obtained and simulated, and (d) percentage of harvested energy used by v, under the EX and EP-12 policies.

Zigbee (296.4nJ/bit transmitting and 324nJ/bit receiving [Lee et al. 2007]), Bluetooth Low Energy (153nJ/bit [Smith 2011]), or the CC2500 radio from Texas Instruments [2012] (200nJ/bit). Aligning the overall energy consumption with that of the chipset is a subject for future work. Note that it is significantly more expensive to receive than it is to transmit, with the majority of the energy consumed when searching for the start of the packet. This can be improved by using a clock with higher accuracy, using clock synchronization techniques (including techniques based on properties of *indoor lighting* [Li et al. 2012]), or through receiver initiated MAC protocols such as Dutta et al. [2010].

6.5. Energy-Harvesting-Aware Data Rate Adaptation Policies

Communicating energy-harvesting nodes need to jointly adapt their link layer behavior. Hence, we experimentally evaluated energy-harvesting-aware link data rate adaptation policies—the *Decoupled Rate Control* (DRC) policies, proposed in Gorlatova et al. [2013]. These polices aim to achieve *energy neutrality* (see Section 5.3). Under the DRC policy, prototypes u and v determine, independently, their desired energy-spending rates $s'(i)$, using the EX or EP-K policies described in Section 5.3. The prototypes exchange these values and calculate their data rates as $\{r_u(i), r_v(i)\} \leftarrow max\{min[s'_u(i), s'_v(i)]/(c_{tx} + c_{rx}), r_{min}\}$, where r_{min} is the minimum rate required to maintain a link.

We extensively evaluated the DRC policies with different light energy inputs. Figure 15(b), for example, shows the data rates assigned by the DRC policy, in combination with the EX and EP-1 policies, corresponding to energy-harvesting rates shown in Figure 15(a) (where error bars represent variations in energy-harvesting rates in different experiments). For these experiments, we provided the prototypes with light inputs corresponding to the light energy recorded over a day in L-1 and L-2 (nearby locations in the same office), compressed to 700s and scaled by 20.0x. The prototypes were equipped with OPVs. In 10 experiments with these light inputs, the variability in the total energy harvested was under 4.7% for node u, and under 6.1% for node v.

The experimentally obtained link data rates are slightly lower than the data rates obtained in simulations. For example, Figure 15(c) shows experiments with the DRC policy in combination with EP-12 (each data point in Figures 15(c) and 15(d) corresponds to a complete EnHANT testbed experiment that lasts 700s). For node energy-harvesting rates shown in Figure 15(a), nodes u and v achieved, correspondingly, 90%–92% and 84%–87% of the data rates expected based on the simulations. These discrepancies are attributed to packet errors, which are not captured by the simulations, and are the result of MAC layer collisions, system computational overloads, and noiseinduced bit errors.
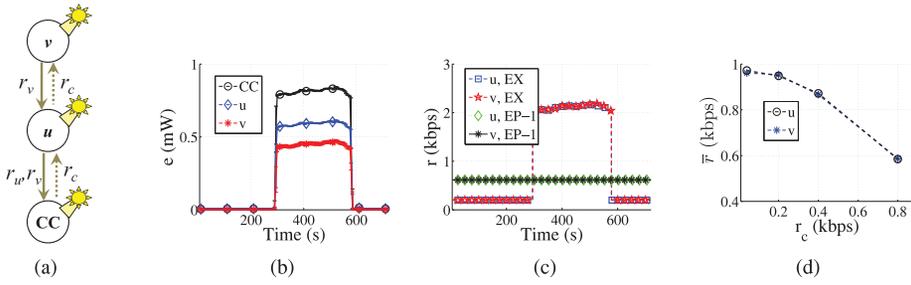
Fig. 16. FLEX flow control policies for (a) a *3-node line* network: Using a day-long irradiance trace compressed to 720s and scaled by 60.4x from L-2, and the corresponding (b) energy-harvesting rates from 8 experiment repetitions with error bars, (c) data rates under the EX and EP-1 node energy allocation policies, and (d) average data rates, under the EX policy, for different values of r_c.
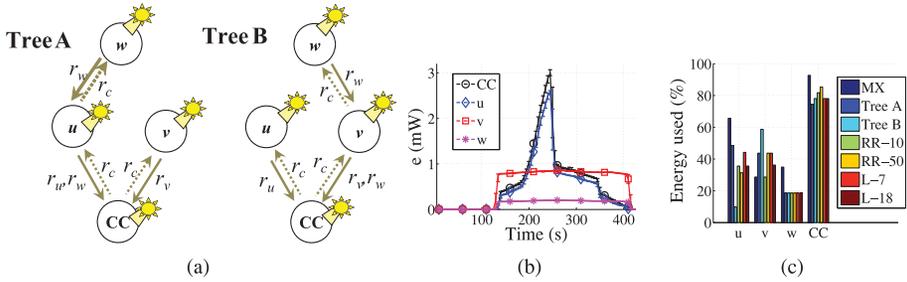

Fig. 17. Topology adaptation policies for (a) a 4-node *"diamond"* network: Using a day-long irradiance trace compressed to 400s and scaled by 20.0x from L-1, L-2, and the corresponding (b) energy-harvesting rates from eight experiment repetitions with error bars, and (c) percentage of harvested energy used by the nodes.

In combination with the EP-1 policy, the DRC policy achieves energy neutrality. Combined with other policies, it may, due to its *per-time slot* calculations, underspend the overall energy harvested. Moreover, it may also *overspend energy* if nodes need to communicate (e.g., to maintain synchronization, or to exchange basic energy awareness information) in all time slots, including those in which $\min\{s_u'(i), s_v'(i)\} = 0$. Both underspending and overspending of energy can be observed in Figure 15(d), which shows the energy spent by nodes running the DRC policy in combination with the EX and EP-12 policies, for energy-harvesting rates shown in Figure 15(a).

7. ENERGY-HARVESTING ADAPTIVE FLOW CONTROL AND TREE SELECTION

In this section, we evaluate, using a small multihop network of prototypes, policies for networks of energy-harvesting nodes. To the best of our knowledge, the EnHANT testbed network is the first UWB-IR-based multihop network. We note that most algorithms proposed for networks of energy-harvesting nodes [Liu et al. 2010; Chen et al. 2011; Huang and Neely 2011] are too complex for implementation in ultra-low-power indoor environments (recall that indoors the amount of light energy available is a thousand times lower than outdoors), as they require multiple local [Chen et al. 2011] or global [Liu et al. 2010] iterations, or complex calculations [Huang and Neely 2011]. We thus focus on evaluating simple policies and heuristics.

Our evaluations focus on data collection scenarios (e.g., ID collection), corresponding to the envisioned object tracking applications. Figures 16(a) and 17(a) show the considered network topologies: a *3-node line* network and a *diamond* network. In these topologies, prototypes u, v, and w generate messages, and send them, via multihop collection trees, to a prototype that serves as a *Collection Coordinator* (CC). In Section 7.1,

---

ALGORITHM 1: FLEX policy running on the CC, for the 3-node line multihop network topology.

Input: $s'_u(i), s'_v(i), s'_{cc}(i)$;
if $s'_u(i) > s'_v(i) \cdot (2 + c_{rx}/c_{tx})$ **then** $\widehat{s}_v(i) \leftarrow s'_v(i)$; $\widehat{s}_u(i) \leftarrow s'_u(i) - s'_v(i) \cdot (1 + c_{rx}/c_{tx})$;
else, $\{\widehat{s}_u(i), \widehat{s}_v(i)\} \leftarrow s'_u(i)/(2 + c_{rx}/c_{tx})$;
if $(s'_u(i)/c_{tx}) > (s'_{cc}(i)/c_{rx})$ **then**
$\quad R_T \leftarrow s'_{cc}(i)/c_{rx}$; $R_{curr} \leftarrow [\widehat{s}_u(i) + \widehat{s}_v(i)]/c_{tx}$;
$\quad$ **if** $R_{curr} > R_T$ **then** $\widehat{s}_u(i) \leftarrow \widehat{s}_u(i) \cdot R_T/R_{curr}$; $\widehat{s}_v(i) \leftarrow \widehat{s}_v(i) \cdot R_T/R_{curr}$;
Return: $r_u(i) \leftarrow \max[\widehat{s}_u(i)/c_{tx}, r_{min}]$; $r_v(i) \leftarrow \max[\widehat{s}_v(i)/c_{tx}, r_{min}]$;

---

we evaluate a set of flow control policies that assign data rates to network nodes. In Section 7.2, we briefly evaluate heuristics for energy-harvesting adaptive collection tree selection.

7.1. Flow Control

Our experiments with networks of energy-harvesting nodes strongly indicated the need for flow control policies. When nodes set their data rates without considering other nodes (i.e., by setting $r(i) \leftarrow s'(i)/c_{tx}$), *networked* nodes overspend energy dramatically. This is particularly pronounced in UWB-IR-based networks, where, due to the ratio between energy spending on receiving and transmitting (see Section 6.3), a node spends approximately 10 times more energy on receiving and forwarding a packet for another node than on transmitting its own packet.

We evaluated flow control policies, to which we refer to as FLEX, that are based on the DLEX node data rate assignment algorithm proposed in Fan et al. [2008]. FLEX, running on the CC, assigns fair (*lexicographically maximal* [Fan et al. 2008]) data rates to the network nodes.[17] Under FLEX, the prototypes independently determine their desired energy-spending rates, $s'(i)$, using the EX or EP-*K* node energy allocation policies, and send them to the CC. The CC allocates data rates such that the total energy-spending rates of the nodes do not exceed $s'(i)$, and the assigned data rates are fair. The FLEX policy for the 3-node line network topology (Figure 16(a)) is shown in Algorithm 1. The algorithm first computes lexicographically maximal data rates that the forwarding node u can support, then checks whether the CC can support these rates, and, if necessary, scales the rates proportionally. To maintain network connectivity, nodes communicate at a rate of at least r_{min}.

We conducted extensive experiments with the FLEX policy using a variety of light inputs. Figure 16(c), for example, demonstrates data rates assigned by FLEX, in combination with the EX and EP-1 policies, for a network with node energy-harvesting rates shown in Figure 16(b) (where error bars represent variations in energy-harvesting rates in different experiments). We provided the prototypes with light inputs corresponding to the light energy recorded over a day in L-2, compressed to 720s (12min) and scaled by 60.4x. The prototypes were equipped with OPVs. In eight experiments, the variability in the total energy harvested was under 4.1% for the CC and under 1% for nodes u and v. Despite providing nodes nearly identical light inputs, the harvesting rates, shown in Figure 16(b), *differed by more than 1.8x*. This is due to different efficiencies of the OPVs integrated with different prototypes.

The FLEX policy, in combination with the EP-1, ensures energy neutrality. In combination with EX and with EP-*K* for $K \neq 1$, similarly to the DRC policy examined in the previous section, FLEX may *underspend or overspend* the energy of the nodes. For

---

[17]The DLEX data rate allocation algorithm developed in Fan et al. [2008] is implicitly tied to a particular, EP-based, node energy allocation policy. In FLEX, we combine the data rate allocation algorithm of Fan et al. [2008] with different node policies.

Table IV. Data Rates $\bar{r}$ under Different Topology Adaptation Policies

Policy	Tree A	Tree B	RR-10	RR-50	L-7	L-18
$\bar{r}$, % of MX	82.4	95.1	92.6	97.8	88.2	92.8

example, in the evaluation scenarios shown in Figure 16, combined with EP-1, FLEX spends 96.4% of the energy harvested by node u; combined with EX, it spends 125%. To achieve energy neutrality, FLEX needs to take into account energy spending on *control messages*, transmitted by the CC at a fixed rate r_c. For example, prior to calculating node data rates according to Algorithm 1, nodes' $s'(i)$ values need to be reduced as $s'_{cc}(i) \leftarrow s'_{cc}(i) - r_c \cdot c_{\text{tx}}$, $s'_u(i) \leftarrow s'_u(i) - r_c \cdot [c_{\text{rx}} + c_{\text{tx}}]$, $s'_v(i) \leftarrow s'_v(i) - r_c \cdot c_{\text{rx}}$. We note that r_c *values directly affect nodes' data rates*, as can be seen, for example, in Figure 16(c), which corresponds to energy-harvesting rates shown in Figure 16(a).[18] In future work, we will develop flow control policies that jointly determine, and adapt to environmental energy dynamics, node energy allocations, control message rates, and data rates.

7.2. Collection Tree Selection

We evaluated collection tree adaptation policies in a "diamond" network topology shown in Figure 17(a), where nodes u and v send messages to the CC directly, while node w sends its messages via a forwarder, u (*tree A*) or v (*tree B)*. A collection tree is chosen by the CC based on one of the heuristics outlined below. Once a tree is selected, node data rates are assigned using the FLEX flow control policies, in combination with the EX policies. We compare the performance of these heuristics to the performance in networks where the collection trees are fixed and to an EP-1-based *MX policy* that calculates the best collection tree offline.

Round Robin (RR-*T*): Trees A and B alternate every T seconds. We evaluate the RR-*T* policy for $T = 10$s (RR$_{10}$) and $T = 50$s (RR$_{50}$).

Battery Level-based (L-*B*): The collection tree is changed if the battery level of the forwarder in the current tree is B mC lower than the battery level of the other forwarder. We evaluate the L-*B* policy for $B = 7$ mC (L_7) and $B = 18$ mC (L_{18}). We note that tree selection based on node battery levels is commonly used in (non–energy-harvesting) sensor networks (e.g., Wu et al. [2010]).

The evaluation results presented below correspond to the node energy-harvesting rates shown in Figure 17(b) (error bars represent variations in harvesting rates in different experiments). We provided the prototypes with light inputs corresponding to the light energy recorded over a day in L-1 and L-2 (nearby locations in the same office [Gorlatova et al. 2011]), compressed to 400s (6.6min) and scaled by 20.0x. The prototypes were equipped with OPVs. In eight experiments, the variability in total energy harvested was under 3% for all nodes.

Table IV shows the average data rates obtained under different policies as a percentage of data rates obtained under the MX policy. Note that using Trees A and B results in different data rates (due to the difference in the energy harvested by u and v). Figure 17(c) shows the percentage of harvested energy used by the nodes.

All collection tree adaptation policies resulted in data rate improvements over the fixed Tree A. This emphasizes the need for topology adaptation policies for networks of energy-harvesting nodes. The use of the L-*B* policies *did not result in data rate improvements* over the fixed Tree B. Due to FLEX and EX not taking battery levels into account, L-*B* only changes energy use patterns in nodes u and v (as can be seen in Figure 17(c)). Developing and implementing policies for joint data rate determination and topology adaptation for EnHANTs is a subject for future work.

---

[18]Each data point in Figure 16(d) corresponds to a complete 700s (11.6min) testbed experiment.
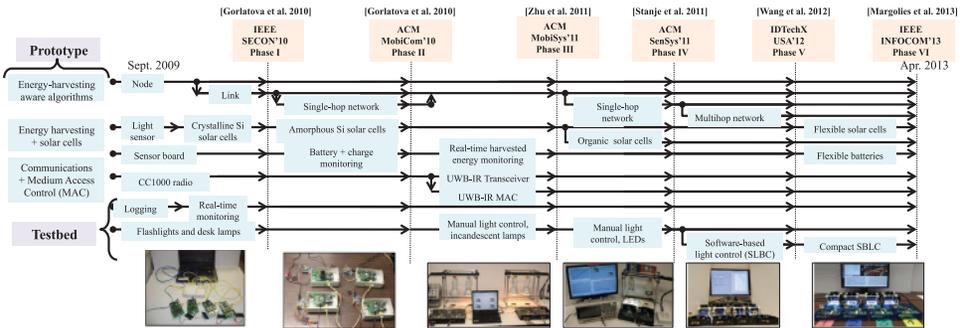
Fig. 18. A timeline of the prototype and the testbed development stages.

8. LESSONS LEARNED

In this section, we first summarize the design and development process for the different prototype components and the testbed. Then, we list several lessons that were learned throughout the process.[19] We separate the discussion to lessons learned in the design and development of the prototypes and lessons learned in the design and development of the testbed.

8.1. Evolutionary System Development

The prototypes and the testbed were developed and integrated in a number of phases over a 4-year period, as shown schematically in Figure 18.[20] We initially used off-the-shelf components; throughout the different phases, nearly all the off-the-shelf components were replaced with custom-designed ones. At the end of each phase, we integrated the phase's prototype and testbed components, and demonstrated a fully functional system. This approach necessitates continuous interaction between the designers and developers of different components and algorithms, and ensures that the integration challenges are addressed quickly. We summarize the functionality development steps below.

—**Energy harvesting** allows EnHANTs to self-power by obtaining energy from ambient sources. We have fabricated *flexible solar cells* that efficiently harvest indoor light, and integrated them with the prototypes. Initially, we designed the prototypes to sense, but not harvest, available environmental energy (Phase I). Next, we integrated rigid commercial solar cells and implemented real-time energy-harvesting state monitoring (Phases II and III). Finally, we integrated the custom-designed organic solar cells (Phase V) and commercial flexible solar cells (Phase VI).

—**Ultra-Wideband Impulse-Radio (UWB-IR) wireless communications** spend significantly less energy than other low-power wireless technologies [Crepaldi et al. 2011]. Early-phase prototypes communicated with each other via standard (non-UWB) commercial sensor network mote transceivers [Moog Crossbow 2012]. Prior to integration of the custom UWB-IR communication modules in Phase III, we substantially modified the mote operating system (which did not support custom transceivers). The integration additionally required the implementation of a custom medium-access control module, since the UWB-IR transceiver characteristics differ greatly from the properties of the conventional transceivers.

---

[19]While this article focuses on the technical lessons learned, we also learned several organizational lessons, which were summarized in Gorlatova et al. [2013] and Margolies et al. [2014].

[20]As described in Section 3, all experiments were conducted using the Phase V version of the prototype and testbed from Figure 18.

—**Energy-harvesting adaptive algorithms** were first designed and developed for simple single-node scenarios and were later implemented for network scenarios. Following the integration of the UWB-IR transceivers in Phase III, we reimplemented the algorithms to take the UWB-IR characteristics into account. In Phase VI, we introduced an adaptive multihop network.

—**Testbed functionality** first consisted of a data logger with a simple visualization interface, which we replaced with a custom-designed real-time monitoring and control system. We additionally developed several prototype light energy control systems, from relatively simple manual setups (Phases III and IV) to a software-based system that exposes the prototypes to real-world trace-based light energy conditions (Phase V).

At the end of each phase, a fully functional prototype testbed was demonstrated, as indicated in Figure 18.

8.2. Designing and Prototyping EnHANTs: Bridging the Theoretical and Experimental Gap

While designing, developing, and evaluating energy-harvesting adaptive networking algorithms, we aimed to bridge theory and experiments. However, the majority of the practical problems we ran into are considered out of the scope of theoretical studies. Hence, we list a few *gaps* between our experience and the common theoretical assumptions.

—Exact energy-spending mechanisms and detailed energy storage state tracking are very difficult to implement, and their implementation places substantial loads on the prototype (i.e., battery sampling overhead). This observation is in contrast to the assumptions made by many theoretical energy-harvesting adaptive algorithms which commonly rely on the availability of such mechanisms [Dutta et al. 2010; Wu et al. 2010; Dutta et al. 2006].

—In many theoretical studies on energy-harvesting adaptive networking, the hardware parameters are taken for granted and *are not part of the solution space*. Therefore, many of the parameters of the energy-harvesting adaptive algorithms that we evaluated corresponded to open theoretical problems. Specifically, to the best of our knowledge, *there are no guidelines* on issues such as determining the optimal battery size for energy-harvesting nodes, optimal energy storage quantization level and update frequency, and network adaptation recalculation interval. Additionally, we found that accounting for a minimum link rate r_{min} (Section 6.5) and a network control traffic rate r_c (Section 7.2) complicated maintaining energy neutrality in the network's nodes. This is a phenomenon that is often ignored.

—Theoretical studies on MAC for UWB-IR (i.e., Di Benedetto et al. [2006]) typically assume the existence of complex physical layer implementations and, therefore, cannot be evaluated in a testbed. On the other hand, even basic theoretical MAC layer algorithms (e.g., Aloha and CSMA) are difficult to implement using experimental hardware. The lesson learned is that MAC layer evaluation using experimental hardware can provide grounding assumptions for theoretical studies.

—Policies optimized for an individual layer of the network (e.g., node, link) may not perform well when combined together and can be significantly affected by the choice of physical components. For example, we observed that using different types of solar cells results in different dynamics in energy harvesting (see Figure 9(c)). Similarly, different transceivers have *very different levels of physical layer reliability* (i.e., bit error rates). Hence, when we switched from CC1000 radios to the UWB-IR transceivers in Phase III (see Figure 18), we had to redesign the higher layer algorithms.

8.3. Testbed Design and Development

8.3.1. Debugging Interfaces. In system design with many components, debugging interfaces are critical. Observing critical system parameters while not interfering can be a significant challenge. In our testbed, we have many components that ideally require separate debug interfaces, such as the following:

—The UWB-IR Communication Module (see Figure 13) contains chipset debugging signals which are observed through added ports using an oscilloscope. Additionally, the module contains jumper connections and potentiometers to modify physical layer parameters.
—The prototypes are placed on MIB600 programming boards to enable an Ethernet control channel. However, sending data over the control channel incurs a computational load on the CM, and hence, it must be used sparingly.
—The EHM was designed and verified separately and incorporated with the rest of the prototype without additional debug interfaces. Namely, in the testbed, we could observe B and s values reported by the EHM but had no means of verifying them. That made designing and verifying energy-harvesting adaptive algorithms, which rely on EHM-reported values and control EHM operations, very difficult. Other similar systems (e.g., Texas Instruments [2012]) should consider having debug interfaces added for energy monitoring.

As a consequence of debugging interfaces, the prototype is much larger than envisioned. Although reducing the size is not one of our current goals, it is certainly a challenge to do it while maintaining the debugging capabilities.

8.3.2. Facilitating Repeatable Experiments in an Energy-Harvesting Testbed. To the best of our knowledge, the EnHANTs testbed is the only energy-harvesting testbed that is able to support repeatable experiments. This capability requires generation of physical phenomena and is necessary, since energy generated by other means (i.e., the environment) has uncontrollable order-of-magnitude variations. Below, we outline some of the lessons learned from implementing this functionality.

—The software-based light control system exposes individual prototypes to repeatable light energy conditions (see Section 3.2). Providing *controllable and repeatable* trace-based light energy conditions proved to be very challenging process, involving a painstaking calibration that utilized NIST-traceable photodiodes. Moreover, the energy spending of the prototype, which is performed via activation of a load resistor, also had to be carefully tuned to account for variations in battery voltage and the internal capacitance of the components.
—Lights are *point energy sources,* and thus small changes in photovoltaic cell position changes the energy-harvesting rates dramatically. Hence, we learned the importance of precise and reproducible photovoltaic cell placement. Therefore, the photovoltaics were carefully placed inside custom-designed dark box enclosures, shown in Figure 7.
—As light and heat are inherently related when placing lamps close to the nodes, the solar cells are affected by the heat. However, it is impractical to place lamps far from the nodes. We observed many effects related to heat in solar cells; our experiments demonstrated a loss of up to 7.6% of the harvested energy as solar cells were heating up. In one of our early designs, the intense heat generated by incandescent lamps affected not only the performance of solar cells, but also *the clock frequency drift of the UWB-IR Communication Module*. Hence, we learned the importance of minimizing heat generation and utilized cool white LEDs that were positioned external to the prototype.

9. ONGOING AND FUTURE WORK

In our ongoing work, we have been focusing on developing hardware, software, and algorithms that will support the realization of the long-term EnHANTs vision. Namely, we have been improving the design of the UWB-IR communication module, integrating flexible batteries, and implementing an adaptive multihop network (e.g., Margolies et al. [2013]). We now list some of the directions for future work.

—**Testbed and Control and Monitoring System:** From the prototypes design and testbed experiments, we learned that energy harvesting forces a complete rethinking of the protocol stack as well as the node architecture. However, while our small testbed provides a proof of concept, it does not allow for large-scale networks experimentation. Therefore, in our future work, we will enhance the existing testbed. Namely, we will design and deploy a large testbed composed of the next generations of prototypes. The testbed will allow for evaluating experiments with uncompressed and unscaled light traces as well as for tracking the detailed energy consumption from the various components. Furthermore, we will also have an option to evaluate nodes in *live environments*, where they will harvest energy from ambient light sources instead of the software controlled lights.

   The prototypes will be modified for streamlined deployment in a large testbed. Non-tag functionality will be moved from the prototype CM to a backbone network with a control and monitoring system. The CM in the current prototype deals with both tag-essential functionalities (e.g., MAC, routing, energy management) and non—tag-essential tasks such as logging and event transmission for the purpose of monitoring the experiments. The monitoring tasks unnecessarily stress the CM in terms of power, memory, and performance requirements. The separation will enable further optimizations for both the prototypes and the backbone network control and monitoring system.

—**Microcontroller:** As we are pushing the limits of performance and energy efficiency via a range of custom hardware and software, the legacy MICA2 hardware platform (that has been used for the CM) has been found to be a bottleneck for further research explorations. Therefore, we will integrate a KMote [2014] as a dedicated EnHANTs CM board for the tag-essential tasks. The KMote uses the 16b Texas Instruments MSP430 [TI 2014] which is a state-of-the-art low-power microcontroller. It offers substantial power savings and performance improvements over the 8b Atmel AVR [Atmel 2014] used in MICA2 motes. Through preliminary testing, we have already verified the compatibility of the KMote with the EHM and UWB-M. To allow for a smooth transition of our existing software, drivers, and algorithms, and to enable relatively quick deployment, we plan to initially use the same operating system as in the current testbed TinyOS version 2.1.2 [TinyOS 2014], which also supports the MSP430 platform. In parallel to the development process, we will evaluate other low-power operating systems like Contiki [2014] or Lorien [2014] as potential upgrades.

—**Application Optimization:** The design of the lower layers is usually driven by application requirements. Therefore, we plan to develop tracking and monitoring applications and to evaluate their performance along with various combinations of lower-layer algorithms (e.g., harvesting-aware MAC and forwarding) and hardware configurations (e.g., UWB transceiver settings, computational architectures, and energy storage components). We will, for example, develop and evaluate the algorithms for specific applications including detecting misplaced objects, searching for objects, or obtaining and maintaining an inventory list of objects.

—**Alternate Energy Sources:** The light control system can be used to provide the EHM with traces that were collected from other energy sources. In particular, since

kinetic (movement) energy availability is different from light energy, the developed algorithms should be different [Gorlatova et al. 2015]. The ability to provide the nodes energy traces and emulate a network of kinetic harvesting nodes without the need to fabricate (and move around) kinetic harvesters will be an important step towards the use of this technology for networking applications.

—**Multitag Synchronization in UWB-IR:** Maintaining synchronization in impulse-radio requires novel UWB-IR receiver architectural approaches. When nodes are unsynchronized, the high cost of non–duty-cycled reception while searching for the packet start results in significant energy waste. Synchronization can then be maintained through the use of a lightweight overhead connection. This consumes some energy, however, and when to keep or drop synchronization is an open area of research to be investigated. One possible approach is to use immediate hardware initiated acknowledgments and leverage the energy savings of the existing synchronization. We aim to substantially reduce synchronization energy costs through the development of a cross-layer protocol, optimized to application requirements.

—**Energy-Harvesting-Aware MAC**: Similarly to algorithms in the higher layers (e.g., flow control), the MAC protocols will have to adapt to harvesting and battery states of the node and neighbors. For example, in some cases, it may be beneficial to create a high-power mode and spend more energy than what is typically spent by a tag (e.g., when the EHM reports the battery is fully charged and the tag is harvesting energy). Other examples include energy-harvesting adaptive sleep scheduling. However, implementing sleep scheduling with UWB-IR is nontrivial (due to bit synchronization issues discussed above). Due to the paradigm shift in the receive and transmit cost, we will consider strategies such as receiver-initiated wake-up [Dutta et al. 2010] that shift the energy demand of sending a packet from the receiver to the sender.

—**Form Factor:** We will design and integrate with the prototypes hardware components tailored for the future EnHANT form factor. The current prototypes (Figure 2) are much larger than the envisioned EnHANT (Figure 1). The current prototypes include a number of debug interfaces and test points to aid development. After the completion of the prototype development, the core electronic components, including the microcontroller, UWB-IR transceiver, and energy management circuits, will be combined into a single custom Integrated Circuit (IC). This IC will then be mounted on a flexible Printed Circuit Board (PCB) that includes an integrated antenna, thin-film solar cell, and thin-film battery.

A physical mock-up of an envisioned EnHANT is shown in Figure 1(b). This mock-up consists of a thin-film battery ($2.5 \times 5.8cm^2$, 2.2mAh, 31.6J) [Infinite Power Solutions 2014] and thin-film solar cell ($3.65 \times 6.4cm^2$) [PowerFilm Solar 2014] attached to a PEN plastic substrate an anisotropic conductive paste and contact points that allow the solar cell and battery to be used in the existing testbed. This simple mock-up demonstrates that it is possible to build a functional tag with the desired form factor. Based on an efficiency of the solar cell at 1% and an irradiance level of $1J/cm^2$/day [Gorlatova et al. 2011, 2013], this tag could harvest 0.233J/day and could maintain a continuous data rate of 2.7kbps using the UWB-IR transceiver. The mock-up does not include an antenna, but previous research has demonstrated designs for UWB antennas that are less than $3 \times 3cm^2$ [Chen et al. 2007].

10. CONCLUSIONS

This article describes the design and development of the Energy-Harvesting Active Networked Tag (EnHANT) prototypes and testbed, as well as experiments with energy harvesting adaptive policies. To the best of our knowledge, the prototypes and the testbed are *the first of their kind*. The design, development, and evaluation of the EnHANTs prototypes, testbed, and algorithms brings about various *lessons learned*

that often cannot be modeled. These lessons learned include, for example, the difficulty in implementing precise energy-spending and tracking mechanisms, that small changes in the photovoltaic type or position can significantly change the energy-harvesting rates, and that policies optimized for an individual layer of the network (e.g., node, link) may not perform well when combined together and can be significantly affected by the choice of physical components. We note that these contributions are applicable to various networks of energy-harvesting devices, not just the EnHANTs.

ACKNOWLEDGMENTS

REFERENCES

Atmel 2014. Atmel AVR 8- and 32-bit Microcontrollers. Retrieved from www.atmel.com/products/microcontrollers/avr/default.aspx.

L. Atzori, A. Iera, and G. Morabito. 2010. The internet of things: A survey. *Computer Networks* 54, 15 (2010), 2787–2805.

Audio-technica. 2012. SpectraPulse Ultra Wideband (UWB) Wireless Microphone System. Retrieved from http://www.audio-technica.com/cms/wls_systems/e7dd603ca6147a97/index.html.

N. J. August, H.-J. Lee, and D. S. Ha. 2004. Pulse sense: A method to detect a busy medium in pulse-based ultra wideband (UWB) networks. In *Ultra Wideband Systems*.

N. J. August, H.-J. Lee, and D. S. Ha. 2007. Enabling distributed medium access control for impulse-based ultrawideband radios. *IEEE Trans. Veh. Technol.* 56, 3 (May 2007), 1064–1075.

S. Bandyopadhyay and A. P. Chandrakasan. 2012. Platform architecture for solar, thermal, and vibration energy combining with MPPT and single inductor. *IEEE J. Solid-State Circuits* 47, 9 (Sept 2012), 2199–2215.

B. Buchli, D. Aschwanden, and J. Beutel. 2013. Battery state-of-charge approximation for energy harvesting embedded systems. In *Wireless Sensor Networks*. Springer, 179–196.

A. Castagnetti, A. Pegatoquet, C. Belleudy, and M. Auguin. 2012. An efficient state of charge prediction model for solar harvesting WSN platforms. In *IWSSIP'12*.

S. Chen, P. Sinha, N. B. Shroff, and C. Joo. 2011. Finite-horizon energy allocation and routing scheme in rechargeable sensor networks. In *Proc. IEEE INFOCOM'11*.

Z. N. Chen, T. See, and X. Qing. 2007. Small printed ultrawideband antenna with reduced ground plane effect. *IEEE Trans. Antennas Propag.* 55, 2 (2007), 383–388.

Contiki. 2014. Contiki. Retrieved from http://www.contiki-os.org/.

M. Crepaldi, C. Li, K. Dronson, J. Fernandes, and P. Kinget. 2011. An interference robust S-OOK IR-UWB transceiver chipset. *IEEE J. Solid-State Circuits* 46, 10 (Oct. 2011), 2284–2299.

F. Cuomo, C. Martello, A. Baiocchi, and F. Capriotti. 2002. Radio resource sharing for ad hoc networking with UWB. *IEEE J. Sel. Areas Commun.* 20, 9 (Dec. 2002), 1722–1732.

D. C. Daly, P. P. Mercier, M. Bhardwaj, A. L. Stone, Z. N. Aldworth, T. L. Daniel, J. Voldman, J. G. Hildebrand, and A. P. Chandrakasan. 2010. A pulsed UWB receiver SoC for insect motion control. *IEEE J. Solid-State Circuits* 45, 1 (Jan. 2010), 153–166.

N. Dang, E. Bozorgzadeh, and N. Venkatasubramanian. 2011. QuARES: Quality-aware data collection in energy harvesting sensor networks. In *IEEE IGCC'11*.

A. Dementyev, J. Gummeson, D. Thrasher, A. Parks, D. Ganesan, J. R. Smith, and A. P. Sample. 2013. Wirelessly powered bistable display tags. In *Proc. ACM UbiComp'13*.

M. G. Di Benedetto, L. De Nardis, G. Giancola, and D. Domenicali. 2006. The aloha access (UWB)^2 protocol revisited for IEEE 802.15.4a. *ST J. Res.* 4 (Mar. 2006), 131–141.

P. Dutta. 2014. MicroMote (M^3) Project. Retrieved March 17, 2015 from http://www.eecs.umich.edu/eecs/about/articles/2015/Worlds-Smallest-Computer-Michigan-Micro-Mote.html.

P. Dutta, S. Dawson-Haggerty, Y. Chen, C.-J. Liang, and A. Terzis. 2010. Design and evaluation of a versatile and efficient receiver-initiated link layer for low-power wireless. In *Proc. ACM SenSys'10*.

P. Dutta, J. Hui, J. Jeong, S. Kim, C. Sharp, J. Taneja, G. Tolle, K. Whitehouse, and D. Culler. 2006. Trio: Enabling sustainable and scalable outdoor wireless sensor network deployments *(IEEE IPSN'06)*.

Enocean Alliance. 2014. Homepage Retrieved from http://www.enocean-alliance.org/home/.

K.-W. Fan, Z. Zheng, and P. Sinha. 2008. Steady and fair rate allocation for rechargeable sensors in perpetual sensor networks. In *Proc. ACM SenSys'08*.

M. Gorlatova, P. Kinget, I. Kymissis, D. Rubenstein, X. Wang, and G. Zussman. 2010a. Energy-harvesting active networked tags (EnHANTs) for ubiquitous object networking. *IEEE Wireless Commun.* 17, 6 (Dec. 2010), 18–25.

M. Gorlatova, R. Margolies, J. Sarik, G. Stanje, J. Zhu, B. Vigraham, M. Szczodrak, L. Carloni, P. Kinget, I. Kymissis, and G. Zussman. 2013. Prototyping energy harvesting active networked tags (EnHANTs). In *Proc. IEEE INFOCOM'13 Miniconference*.

M. Gorlatova, Z. Noorbhaiwala, A. Skolnik, J. Sarik, M. Szczodrak, J. Chen, M. Zapas, L. Carloni, P. Kinget, I. Kymissis, D. Rubenstein, and G. Zussman. 2010b. Demo: Prototyping energy harvesting active networked tags. *Presented in ACM MobiCom'10*.

M. Gorlatova, J. Sarik, G. Grebla, M. Cong, I. Kymissis, and G. Zussman. 2015. Movers and shakers: Kinetic energy harvesting for the internet of things. *IEEE J. Sel. Areas Commun., Special Issue on Wireless Communications Powered by Energy Harvesting and Wireless Energy Transfer* 33, 8 (Aug. 2015), 1624–1639.

M. Gorlatova, J. Sarik, P. Kinget, I. Kymissis, and G. Zussman. 2013. Project-based learning within a large-scale interdisciplinary research effort. In *Proc. ACM ITiCSE'13*.

M. Gorlatova, T. Sharma, D. Shrestha, E. Xu, J. Chen, A. Skolnik, D. Piao, P. Kinget, I. Kymissis, D. Rubenstein, and G. Zussman. 2010c. Demo: Prototyping energy harvesting active networked tags (EnHANTs) with MICA2 motes. In *Proc. IEEE SECON'10*.

M. Gorlatova, A. Wallwater, and G. Zussman. 2013. Networking low-power energy harvesting devices: Measurements and algorithms. *IEEE Trans. Mobile Comput.* 12, 9 (2013), 1853–1865.

M. Gorlatova, M. Zapas, E. Xu, M. Bahlke, I. Kymissis, and G. Zussman. 2011. CRAWDAD data set columbia/enhants (v. 2011-04-07). (Apr. 2011).

J. Gummeson, S. Clark, K. Fu, and D. Ganesan. 2010. On the limits of effective micro-energy harvesting on mobile CRFID sensors. In *Proc. ACM MobiSys'10*.

L. Huang and M. J. Neely. 2011. Utility optimal scheduling in energy harvesting networks. In *Proc. ACM MobiHoc'11*.

IEEE. 2007. IEEE Std. 802.15.4a-2007.

IMEC. 2013. Ultralow Power Wireless Communication. Retrieved from http://www2.imec.be/be_en/research/wireless-communication/ultralow-power-wireless-communic.html.

Infinite Power Solutions. 2014. Thinergy Micro-Energy Cells. Retrieved from http://www.infinitepowersolutions.com/products/.

IPS. 2014. Infinite Power Solutions Eval Kits. Retrieved from http://www.infinitepowersolutions.com/products/evalkits.html.

X. Jiang, J. Polastre, and D. Culler. 2005. Perpetual environmentally powered sensor networks. In *Proc. IEEE IPSN'05*.

A. Johansen, T. Sorensen, and P. Bonnet. 2011. Service and experiment: Towards a perpetual sensor network testbed without backchannel. In *Proc. IEEE MASS'11*.

A. Kansal, J. Hsu, S. Zahedi, and M. B Srivastava. 2007. Power management in energy harvesting sensor networks. *ACM Trans. Embedded Comput. Syst.* 6, 4 (2007).

E. Karapistoli, D. G. Stratogiannis, G. I. Tsiropoulos, and F. Pavlidou. 2012. MAC protocols for ultra-wideband ad hoc and sensor networking: A survey. In *Proc. Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*.

KMote. 2014. KMote-B. Retrieved from http://www.tinyosmall.com/product_p/100-101.htm.

J. Y. Le Boudec, R. Merz, B. Radunovic, and J. Widmer. 2004. DCC-MAC: A decentralized MAC protocol for 802.15.4a-like UWB mobile ad-hoc networks based on dynamic channel coding. In *Proc. BROADNETS*.

J.-S. Lee, Y.-W. Su, and C.-C. Shen. 2007. A comparative study of wireless protocols: Bluetooth, UWB, ZigBee, and Wi-Fi. In *Proc. IEEE IECON'07*.

Z. Li, W. Chen, C. Li, M. Li, X. Li, and Y. Liu. 2012. FLIGHT: Clock calibration using fluorescent lighting. In *Proc. ACM MobiCom'12*.

R.-S. Liu, P. Sinha, and C. Koksal. 2010. Joint energy management and resource allocation in rechargeable sensor networks. In *Proc. IEEE INFOCOM'10*.

V. Liu, A. Parks, V. Talla, S. Gollakota, D. Wetherall, and J. Smith. 2013. Ambient backscatter: Wireless communication out of thin air. In *Proc. ACM SIGCOMM'13*.

Lorien. 2014. Lorien. Retrieved from http://lorien.xrna.net.

Z. N. Low, J. H. Cheong, and C. L. Law. 2005. Low-cost PCB antenna for UWB applications. *Antennas Wireless Propag. Lett.* 4 (Dec. 2005), 237–239.

R. Margolies, M. Gorlatova, J. Sarik, P. Kinget, I. Kymissis, and G. Zussman. 2014. Project-based learning within a large-scale interdisciplinary research effort. *CoRR* abs/1410.6935 (2014).

R. Margolies, L. Pena, K. Kim, Y. Kim, M. Wang, M. Gorlatova, J. Sarik, J. Zhu, P. Kinget, I. Kymissis, and G. Zussman. 2013. Demo: An adaptive testbed of energy harvesting active networked tags (EnHANTs) prototypes. In *Proc. IEEE INFOCOM'13*.

N. Michelusi, K. Stamatiou, and M. Zorzi. 2012. On optimal transmission policies for energy harvesting devices. In *Proc. Information Theory and Application (ITA) Workshop*.

B. Miscopein and J.-M. Gorce. 2009. Energy efficient TH-UWB preamble detection. In *Proc. IEEE ICUWB'09*.

Moog Crossbow. 2012. Crossbow MICA2 Mote. Retrieved from http://bit.ly/1Sc9bHB.

D. Noh, L. Wang, Y. Yang, H. Le, and T. Abdelzaher. 2009. Minimum variance energy allocation for a solar-powered sensor system. In *Proc. DCOSS'09*.

J. A. Paradiso and T. Starner. 2005. Energy scavenging for mobile and wireless electronics. *IEEE Pervasive Comput.* 4, 1 (2005), 18–27.

C. Park, K. Lahiri, and A. Raghunathan. 2005. Battery discharge characteristics of wireless sensor nodes: An experimental analysis. In *Proc. IEEE SECON'05*.

P. Peumans, A. Yakimov, and S. R. Forrest. 2003. Small molecular weight organic thin-film photodetectors and solar cells. *J. Appl. Phys.* 93 (2003).

V. Pop, H. J. Bergveld, P. H. L. Notten, and P. Regtien. 2005. State-of-the-art of battery state-of-charge determination. *Measure.t Sci. Technol.* 16, 12 (2005), R93.

Powercast. 2014. Powercast Development Kits. Retrieved from http://www.powercastco.com/products/development-kits/.

PowerFilm Solar. 2014. SP3-37 Solar Cell. Retrieved from http://www.powerfilmsolar.com/products/oem/sp3-37.

PreparingPlexcore. 2014. Preparing the Plexcore PV 1000 Ink System. Retrieved from http://www.sigmaaldrich.com/technical-documents/protocols/materials-sci ence/preparing-the-plexcore.html.

V. Raghunathan, A. Kansal, J. Hsu, J. Friedman, and M. Srivastava. 2005. Design considerations for solar energy harvesting wireless embedded systems. In *Proc. IEEE IPSN'05*.

R. Rao, S. Vrudhula, and D. N. Rakhmatov. 2003. Battery modeling for energy aware system design. *Computer* 36, 12 (2003), 77–87.

P. Smith. 2011. Comparing Low-Power Wireless Technologies. http://www.digikey.com/en/articles/techzone/2011/aug/comparing-low-power-wir eless-technologies. (2011).

S. Solda, M. Caruso, A. Bevilacqua, A. Gerosa, D. Vogrig, and A. Neviani. 2011. A 5Mb/s UWB-IR transceiver front-end for wireless sensor networks in 0.13 CMOS. *IEEE J. Solid-State Circuits* 46, 7 (July 2011), 1636–1647.

G. Stanje, P. Miller, J. Zhu, A. Smith, O. Winn, R. Margolies, M. Gorlatova, J. Sarik, M. Szczodrak, B. Vigraham, L. Carloni, P. Kinget, I. Kymissis, and G. Zussman. 2011. Demo: Organic solar cell-equipped energy harvesting active networked tag (EnHANT) prototypes. In *Proc. ACM SenSys'11*. Best Student Demo Award.

M. Szczodrak and L. Carloni. 2011. Demo: A complete framework for programming event-driven, self-reconfigurable low power wireless networks. In *Proc. ACM SenSys'11*.

M. Szczodrak, O. Gnawali, and L. P. Carloni. 2013. Dynamic reconfiguration of wireless sensor networks to support heterogeneous applications. In *Proc. IEEE DCOSS'13*.

J. Taneja, J. Jeong, and D. Culler. 2008. Design, modeling, and capacity planning for micro-solar power sensor networks. In *Proc. IEEE IPSN'08*.

Texas Instruments. 2012. MSP430 Solar Energy Harvesting Development Tool. Retrieved from focus.ti.com/docs/toolsw/folders/print/ez430-rf2500-seh.html.

P. E. Thoppay, C. Dehollain, and M. J. Declercq. 2008. A 7.5mA 500MHz UWB receiver based on super-regenerative principle. In *Proc. ESSCIRC'08*.

TI. 2014. Texas Instruments MSP430 Microcontroller. Retrieved from www.ti.com/msp430.

Time Domain. 2013. PulsON 400 RCM. Retrieved from http://www.timedomain.com/p400.php.

TinyOS. 2014. TinyOS. Retrieved from http://www.tinyos.net/.

K. Tutuncuoglu, B. Varan, and A. Yener. 2013. Optimum transmission policies for energy harvesting two-way relay channels. In *Proc. IEEE ICC'13*.

B. Vigraham and P. R. Kinget. 2014. A self-duty-cycled and synchronized UWB pulse-radio receiver SoC with automatic threshold-recovery based demodulation. *IEEE J. Solid-State Circuits* 49, 3 (March 2014), 581–594.

J. Wang and D. Katabi. 2013. Dude, where's my card?: RFID positioning that works with multipath and non-line of sight. In *Proc. ACM SIGCOMM'13*.

D. D. Wentzloff, F. S. Lee, D. C. Daly, M. Bhardwaj, P. P. Mercier, and A. P. Chandrakasan. 2007. Energy efficient pulsed-UWB CMOS circuits and systems. In *Proc. IEEE ICUWB'07*.

M. Z. Win and R. A. Scholtz. 2000. Ultra-wide bandwidth time-hopping spread-spectrum impulse radio for wireless multiple-access communications. *IEEE Trans. Commun.* 48, 4 (Apr. 2000), 679–689.

Y. Wu, Z. Mao, S. Fahmy, and N. B. Shroff. 2010. Constructing maximum-lifetime data gathering forests in sensor networks. *IEEE/ACM Trans. Netw.* 18, 5 (2010), 1571–1584.

B. Yang, F. Guo, Y. Yuan, Z. Xiao, Y. Lu, Q. Dong, and J. Huang. 2013. Solution-processed fullerene-based organic schottky junction devices for large-open-circuit-voltage organic solar cells. *Adv. Mater.* 25, 4 (2013), 572–577.

Y. Yang, L. Wang, D. K. Noh, H. K. Le, and T. F. Abdelzaher. 2009. Solarstore: Enhancing data reliability in solar-powered storage-centric sensor networks. In *Proc. ACM MobiSys'09*.

L. Yerva, A. Bansal, B. Campbell, T. Schmid, and P. Dutta. 2012. Grafting energy-harvesting leaves onto the sensornet tree. In *Proc. IEEE IPSN'12*.

Zebra Technologies. 2013. Dart Ultra-Wide Band UWB. Retrieved from http://www.zebra.com/us/en/solutions/technology-need/uwb-solutions.html.

J. Zhang, P. V. Orlik, Z. Sahinoglu, A. F. Molisch, and P. Kinney. 2009. UWB systems for wireless sensor networks. *Proc. IEEE* 97, 2 (Feb. 2009), 313–331.

P. Zhang and D. Ganesan. 2014. Enabling bit-by-bit backscatter communication in severe energy harvesting environments. In *Proc. USENIX NSDI'14*.

J. Zhu, G. Stanje, R. Margolies, M. Gorlatova, J. Sarik, Z. Noorbhaiwala, P. Miller, M. Szczodrak, B. Vigraham, L. Carloni, P. Kinget, I. Kymissis, and G. Zussman. 2011. Demo: Prototyping UWB-enabled EnHANTs. In *Proc. ACM MobiSys'11*.

T. Zhu, Z. Zhong, Y. Gu, T. He, and Z.-L. Zhang. 2009. Leakage-aware energy synchronization for wireless sensor networks. In *Proc. ACM MobiSys'09*.