Leakage-Aware Energy Synchronization for Wireless Sensor Networks

Ting Zhu, Ziguo Zhong, Yu Gu, Tian He and Zhi-Li Zhang
Department of Computer Science and Engineering, University of Minnesota
{tzhu, zhong, yugu, tianhe, zhzhang}@cs.umn.edu

ABSTRACT

To ensure sustainable operations of wireless sensor systems, environmental energy harvesting has been regarded as the right solution for long-term applications. In energy-dynamic environments, energy conservation is no longer considered necessarily beneficial, because energy storage units (e.g., batteries or capacitors) are limited in capacity and leakage-prone. In contrast to legacy energy conservation approaches, we aim at energy synchronization for wireless sensor devices. The starting point of this work is TwinStar, which uses ultra-capacitor as the only energy storage unit. To efficiently use the harvested energy, we design and implement leakage-aware feedback control techniques to match local and network-wide activity of sensor nodes with the dynamic energy supply from environments. We conduct system evaluation under three typical real-world settings — indoor, outdoor, and mobile backpack under a wide range of system settings. Results indicate our leakage-aware control can effectively utilize energy that could otherwise leak away. Nodes running leakage-aware control can enjoy 70% more energy than the ones running non-leakage-aware control and application performance (e.g., event detection) can be improved significantly.

Categories and Subject Descriptors

C.2.4 [**Computer Communications Networks**]: Distributed Systems

General Terms

Measurement, Design, Performance, Experimentation

Keywords

Energy, Ultra-capacitor, Leakage, Wireless Sensor Networks

1. INTRODUCTION

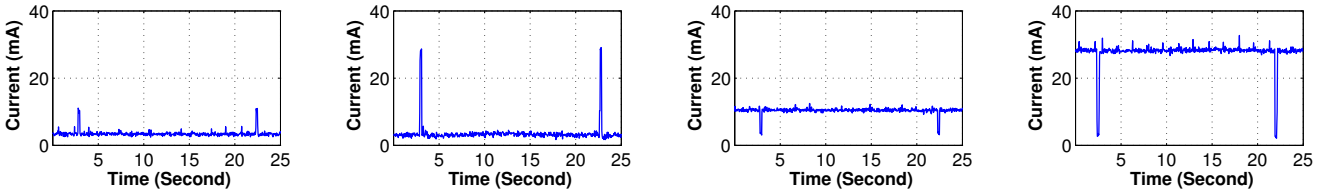
With the increasing need for cyber-physical interaction, wireless sensor networks (WSN) have evolved into a key technology for long-term applications such as military surveillance [12], habitat monitoring [22], and scientific exploration [9]. Due to the low-cost and small size requirements, sensor nodes used in these applications (e.g., mica series) are normally equipped with limited power sources (e.g., 2200 mAh at 3V for two AA batteries), and hazardous or inaccessible environments preclude manual battery replacement. Without renewable energy resources, a sensor node can sustain only a few hours at 100% duty cycle.

Clearly, existing energy conservation techniques can only partially alleviate the conflict between system lifetime and performance. To ensure sustainability, environmental energy harvesting [26] has been pursued by the research community as the right solution for long-term applications. To store harvested energy, researchers have proposed using rechargeable batteries [1], such as NiCAD, NiMH, or Li-ion, or a combination of capacitors and rechargeable batteries [30] as the energy storage buffer. Although these solutions have been proven effective in prolonging the lifetime of the sensor nodes, there is still room for further improvement, because (i) rechargeable batteries have limited recharge cycles due to cyclic memory and crystalline formation (e.g., a Li-ion battery has 500 cycles, NiMH 300 cycles) and (ii) sophisticated recharging circuits and electro-chemical conversion could reduce energy efficiency to as low as 6% [28].

Compared with rechargeable batteries, capacitors possess a set of advantages: they (i) have more than 1 million recharge cycles; (ii) have predicable remaining energy independent of discharge modes; (iii) are robust to temperature changes, shock, and vibration; and (iv) have high charging and discharging efficiency. Furthermore, recent advances in ultra-capacitor technology make it possible to use ultra-capacitors as the *only* energy storage device. For instances, a research group at MIT [21] has announced nanotube-based ultra-capacitors, which can provide energy storage densities comparable to those of batteries. In 2006, a U.S. patent [29] was issued for an electrical energy storage unit using an ultra-capacitor that has an energy/weight value of about 342W·h/kg, twice that of Li-ion batteries. The largest capacitance currently available on the market is 3,000F [19]. Powered by this kind of capacitor, sensor nodes can work for more than 527 days under a 1% duty cycle with a single initial charge of the capacitor. It is highly possible that in the near future we will witness a paradigm shift from a battery-based to an ultra-capacitor-based design for all kinds of embedded devices. Therefore, it is essential to explore this frontier in advance.

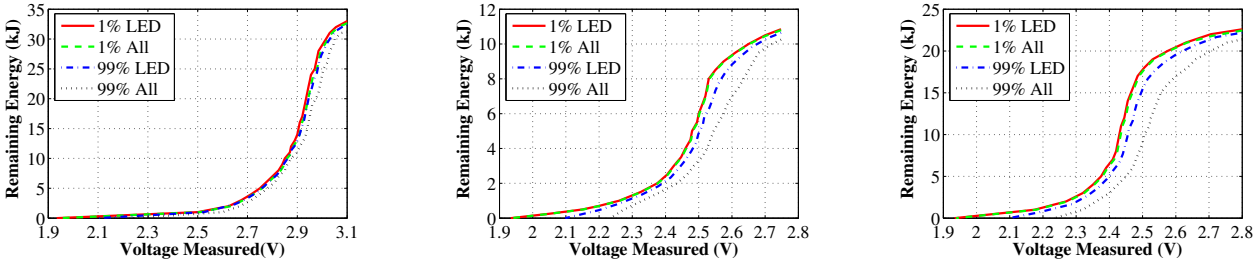
Although ultra-capacitor-based designs have many advantages, they impose a major challenge: the energy leakage of ultra-capacitors is high when they reach their full capacity. To our best knowledge, there has been little investigation

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
MobiSys'09, June 22–25, 2009, Kraków, Poland.
Copyright 2009 ACM 978-1-60558-566-6/09/06 ...$5.00.



(a) 1% Duty Cycle LED On (b) 1% Duty Cycle All On (c) 99% Duty Cycle LED On (d) 99% Duty Cycle ALL On

Figure 1: Workload Patterns



(a) Li-ion 3100 mAh (b) NiCAD 1100 mAh (c) NiMH 2500 mAh

Figure 2: Battery Discharge Characteristics Under Four Types of Workload

into designing systems that consider the leakage of ultra-capacitors. Therefore, this exploratory work would potentially offer an important step toward a wider acceptance of capacitor-based embedded devices.

Unlike previous research that focuses mainly on hardware design, the driving idea of this work is *energy synchronization* – a holistic approach towards energy equilibrium in WSN. A lightweight predictor is designed to estimate the life time of a sensor node based on the available environmental energy and the remaining energy inside the ultra-capacitor. Leakage-aware feedback control is then used to suggest an appropriate duty-cycle to the application layer, based on the gap between the predicted and targeted lifetimes. Instead of *conserving* as much energy as possible, the design objective of our work is to *consume* as much energy as possible while maintaining operational sustainability. Specifically, the major contributions of this work are as follows:

• To our knowledge, this is the first in-depth work to investigate the ultra-capacitor-based leakage-aware design. We empirically model the leakage of ultra-capacitors using offline and online methods to avoid environmental effect. The leakage model allows us to predict the lifetime and control the duty cycle of a node precisely to achieve an equilibrium between energy supply and demand.

• We design and implement both local and network-wide application-level adaptation techniques under diversified real-world environments including indoor, outdoor as well as mobile settings. The system evaluation on a network of TwinStar nodes indicates that our leakage-aware design successfully utilize energy that could otherwise leak away to improve application performance significantly.

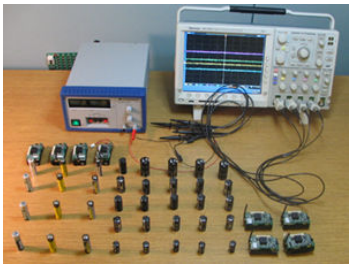


Figure 3: Lab Setting for Accurate Measurement

The rest of the paper is organized as follows: Section 2 describes the motivation behind a leakage-aware design for sensor networks. Section 3 gives an overview of the design architecture. Hardware design, modeling, control, and adaption are presented in Sections 4, 5, 6, and 7, respectively. System implementation and evaluation are detailed in Section 8 and 9. Related work is discussed in Section 10. Finally, Section 11 concludes the paper.

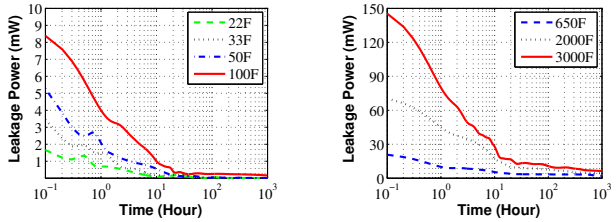
2. MOTIVATION

This section describes the motivation for our work by revealing the uncertainty in battery modeling and identifying challenging issues in ultra-capacitor-based design. In the empirical study, we used different types of batteries, ultra-capacitors, a DC Power-meter and a Tektronix DPO4054 Digital Phosphor oscilloscope as shown in Figure 3.

2.1 Uncertainty in Battery Modeling

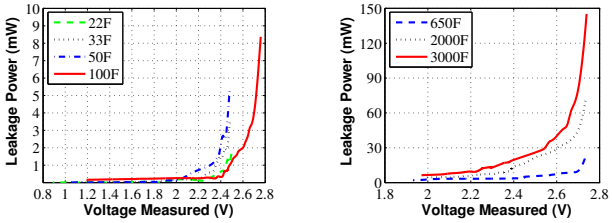
An essential requirement for energy management is estimating remaining energy accurately. In battery-based designs, remaining energy would be estimated by establishing battery models. Researchers [23] have demonstrated the possibility of estimating remaining energy with an ±3% error, assuming a fixed discharge rate, temperature, and battery type. However, it would fail to apply to sensor network settings in which the workload is driven by unpredictable events/workload, and changing environmental factors (e.g., temperature) affects electrochemical reaction significantly. In contrast, the remaining energy inside an ultra-capacitor can be precisely estimated according to its voltage.

Impact of Workload: To investigate the impact of changing workloads on the remaining energy estimation in battery-based designs, we conducted a series of experiments using four types of workload, as shown in Figure 1. These synthetic workloads are generated by choosing different sets of active components (LED only vs. all components) and workload (1% vs. 99%). By periodically measuring the voltages, we can get the battery discharging characteristics under these four workloads, as shown in Figure 2. Figure 2 confirms that even for batteries of the same type, remaining energy differs with different types of workloads. For example, Figure 2c shows that at the 2.5V, two NiMH batteries can supply 17.8kJ at the 1%-LED workload (Figure 1a), but only 9.8kJ at the 99%-All-On workload (Figure 1d). Clearly,



(a) Small Capacitance (b) Large Capacitance

Figure 4: Leakage Power vs. Time



(a) Small Capacitance (b) Large Capacitance

Figure 5: Leakage Power vs. Voltages

this big uncertainty would lead to ineffective energy control.

Impact of Battery Type: We also conducted a series of experiments to investigate the consistency of battery models over different battery types. In this study, three different types of the batteries are investigated: Li-ion, NiCAD and NiMH. Comparing Figures 2a, 2b, and 2c indicates that different types of the batteries have different discharge characteristics and different capacities. In other words, there is no single battery model that can be applied to different battery types. For example, when a sensor node runs under the 1%-LED workload, remaining energy under 2.7V is 3.8 kJ, 10.55 kJ, and 22.16 kJ for Li-ion, NiCAD, and NiMH batteries, respectively.

Figure 2 also shows that the discharge curves for different batteries exhibit deep slopes within a certain voltage range, in which a small error in voltage reading would be amplified into a large error in estimating remaining energy. For example, as shown in Figure 2b, an 0.1V error would be translated into a 200% energy difference between 2.4V and 2.5V readings.

2.2 Leakage in the Energy Storages

This section presents our empirical study on the leakage profile of batteries and ultra-capacitors. We conducted experiments over a period of 6 weeks, using different types of batteries and ultra-capacitors. Our empirical study concludes that (i) ultra-capacitors are a good substitute for batteries and (ii) energy leakage in ultra-capacitors should not be overlooked.

Leakage Profile Comparison: To simplify the comparison against ultra-capacitors, Li-ion batteries are chosen as representatives of the other batteries. We conducted leakage profile comparisons between Li-ion batteries and a 2,000F ultra-capacitor over a period of 2 months after they are fully charged. For Li-ion battery, the leakage rate is 8% per month. The 2,000F ultra-capacitor has a very high leakage rate (43.8%) during the first month, but low leakage rate (5.26%) during the second month. At the end of the second month, the ultra-capacitor still has 1.9V of remaining voltage. These results indicate that ultra-capacitors have leakage performance that is comparable to that of batteries when voltage is controlled under an appropriate level. However, ultra-capacitors suffer severe energy leakage when it is charged to the limit.

Leakage of Ultra-Capacitors: In this experiment, seven types of ultra-capacitors are used, ranging from 22F to 3000F. After an ultra-capacitor was fully charged, we isolated it and continuously monitored its remaining voltage over 1000 hours. Figure 4 reveals that (i) the leakage power reduces over time after the initial charge, and (ii) leakage is more severe for larger ultra-capacitors than for smaller ultra-capacitors. For example, at 2.7V, the leakage power of a 3,000F capacitor is about 17 times of that of a 100F.

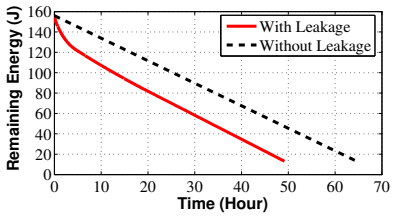


Figure 6: Remaining Energy Vs. Time

We also investigated the correlation between the voltage and leakage of ultra-capacitors. Figure 5 shows that when the voltage of ultra-capacitors approaches its limit, the value of leakage power increases significantly. For example, when the voltage of the 100F ultra-capacitor increases to 2.7V, its leakage power is 8.4 mW, which is equivalent to the power needed for powering a Telos node to work under more than 13% duty cycle.

Lifetime Difference due to Leakage: With the empirical data collected, we simulate the lifetime of a node considering leakage. Figure 6 compares the remaining energy inside the ideal non-leaking ultra-capacitor and the real ultra-capacitor with leakage. Here we use a fully charged 50F ultra-capacitor, to power a MicaZ node at 1% duty cycle. We can see that ideally, without leakage the MicaZ node's lifetime would be 65 hours, but because of the leakage, the MicaZ node's lifetime is actually only 49 hours, or 24% less than the lifetime we would otherwise expect.

3. OVERVIEW OF SYSTEM DESIGN

The empirical study in Section 2 indicates that energy leakage is a critical hurdle that hinders designers exploiting the benefits of ultra-capacitor over batteries. To address the hurdle, we propose a three-layer design as shown in Figure 7.

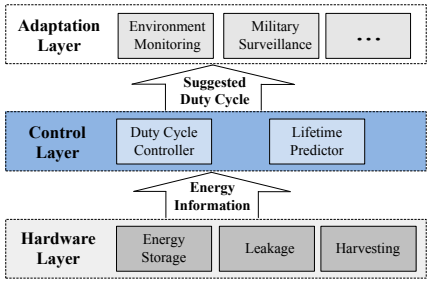


Figure 7: Overview of System Architecture

• The hardware layer uses an energy harvesting circuit (e.g., solar panel or wind generator) to harvest the environmental energy and uses an ultra-capacitor as the *only* energy storage unit to power sensor nodes. This layer also (i) monitors the remaining energy inside the ultra-capacitor, (ii) samples the energy harvesting rate from the energy harvesting circuit, and (iii) models the leakage profile of the capacitor. These three pieces of information are then fed into the control layer.

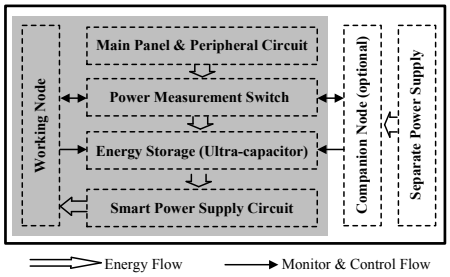


Figure 8: TwinStar Hardware Architecture

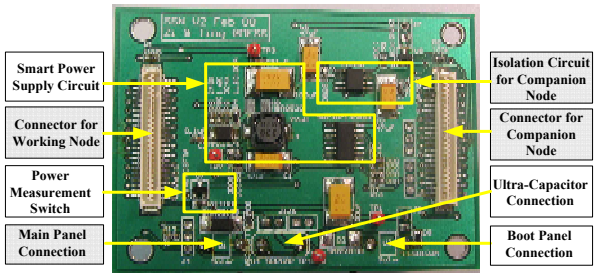


Figure 9: TwinStar Platform

• The control layer predicts the lifetime of a node based on the energy harvesting, leakage, and consumption rates. The difference between the predicted lifetime and user-specified lifetime is used as the control signal to the duty cycle controller, which *suggests* a certain percentage of duty cycle change to the application. The control layer deals with two conditions: energy oversupply and undersupply. In the case of oversupply, the controller prevents the ultra-capacitor from charging to its limit, hence reducing the energy leaked away. In the case of undersupply, the controller signals the running program to reduce the rate of energy consumption so as to ensure the minimal but critical activities of the sensor node at any time.

• In the adaptation layer, a running application changes its working schedules according to the suggested energy budget. Adaptation can be archived by changing per-component activities in communication and sensing as well as per-flow activities [15]. We note that designs for the adaption layer are highly diverse. Instead of using a case-by-case approach, we propose a generic adaptation technique by treating duty-cycle as first-class resource that can be allocated, scheduled and adjusted in Section 7.

4. HARDWARE DESIGN OF TWINSTAR

TwinStar is an add-on power board that harvests energy from environments and uses an ultra-capacitor as the only energy storage to overcome the intrinsic limitations of batteries (e.g., energy uncertainty, limited recharge cycles, low conversion efficiency, and environment unfriendliness). The architecture of TwinStar is shown in Figure 8. It consists of (i) the solar panels and peripheral circuit for energy harvesting; (ii) the power measurement switch; (iii) the ultra-capacitor-based energy storage; and (iv) the smart power supply circuit with a DC/DC converter for powering the working node attached to the TwinStar board. The corresponding printed circuit board is shown in Figure 9. Due to space constraints, we only explain a few unique features of the TwinStar design in the rest of the section.

4.1 Smart Power Supply Circuit

To accommodate fluctuating ambient energy, the voltage of a power supply shall be stabilized. We apply a high-efficiency switch-type DC/DC converter for providing a stable power supply for the working node. In the battery-based approaches, normally a DC/DC converter is powered directly by a battery, assuming enough energy is left in the battery [15, 16]. However, it would be problematic for the ultra-capacitor-based design, especially under extremely low ambient energy situations due to the *zero-energy bootstrap* problem: Initially, the voltage of the ultra-capacitor is around 0 and all the system components do not work including the DC/DC converter. When the voltage of the ultra-capacitor reaches a threshold level, the converter enters a warming up stage, during which the high-efficiency switch-type converter needs to excite the external inductor into oscillations with high transient current (at the level of 10mA). If the scavenged energy is not sufficient to support the converter to finish this stage, the voltage of the ultra-capacitor drops below the threshold level and the converter fails to boot up. Such failure repeats itself, wasting energy harvested from the environment.

To address this zero-energy boot-up challenge, a smart control circuit is designed to automatically boot-up and shut down the DC/DC converter. The basic idea is that during the charging state, the DC/DC converter is kept shut down by the smart control circuit until the ultra-capacitor accumulates sufficient energy and reaches a voltage much higher than the DC/DC converter's input voltage threshold (1.1V); during the discharging period, the DC/DC converter is turned off when the voltage of the ultra-capacitor approaches the minimum input voltage (0.7V). In this way, high boot-up current stage and energy waste can be avoided.

However, achieving this kind of smart control needs to overcome a hidden challenge: there is no appropriate power supply for the smart control circuit itself before enabling the DC/DC converter. In our design, we address all the above challenges by proposing a novel dual solar panel solution (a small boot solar panel and a big main panel) together with a Schmitt trigger-based control circuit. The design is illustrated in Figure 10.

As shown in Figure 10, a small boot solar panel is applied to power the Schmitt trigger module that is used to control the on/off status of the DC/DC converter. The small panel charges a small capacitor (C1, which is 47μF) and keeps the voltage of C1 higher than 2.5V so as to power the Schmitt trigger, which is an extremely low power device with power consumption less than 10μW. Therefore, even with a low environmental energy supply, the smart control circuit can still work.

The main panel is the one harvesting energy for the ultra-capacitor. Note that when the DC/DC converter starts to work, its output also feeds back to power the Schmitt trigger, avoiding shutting down the DC/DC converter when there is no environmental energy but the ultra-capacitor is still energy-rich.

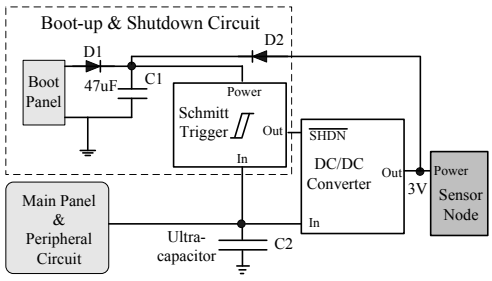


Figure 10: Boot-up and Shutdown Control Circuit

4.2 Ultra-Low-Power Measurement Switch

To enable the control middle layer to accurately predict the lifetime of the sensor node, the hardware provides power measurement capabilities. Keeping in mind that it is important to make the measurement circuit simple and energy efficient, we used P-channel MOSFET as the switches (K_1 and K_2 as shown in Figure 11), which are controlled by the sensor node.

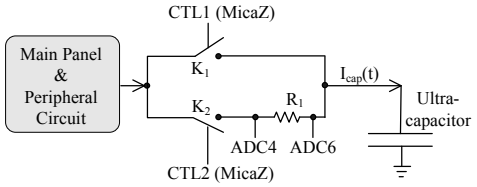


Figure 11: Power Measurement Circuit

Normally the switch K_1 is kept on to harvest environmental energy. When measuring the energy harvested into the ultra-capacitor, K_1 is switched off and K_2 is turned on by the control signals from the sensor node. In our implementation, we use MicaZ node and its $ADC4$ and $ADC6$ pins extract the voltage across the resister R_1. Since the resistance of the resistor R_1 is known, the current $I_{cap}(t)$ goes through R_1 can be calculated.

The power consumption of the P-MOSFET switch is extremely low, at the level of nW, and thus negligible. A very small resister R_1 is used for the voltage formation and used only during the measurement moment (at the level of μs), therefore having a negligible effect on the charging efficiency for the ultra-capacitor.

4.3 External Sensor Nodes

Our system supports two external sensor nodes: (i) a working node, (ii) an optional companion node. The working node is powered solely by the ultra-capacitor. It periodically samples the voltage of the ultra-capacitor. Based on the sampled value it calculates the remaining energy inside the ultra-capacitor. It also periodically controls the power measurement switch to measure the environmental power and calculate the harvested energy.

In order to analyze the system performance and support on-line debugging, the TwinStar platform also contains the interface for attaching an optional companion node (in Figure 9). It is powered and functions separately. Isolation is important here because this design prevents the companion node's interfering with the working node (in Figure 9).

The companion node has three capabilities: (i) periodically sampling the voltage of the ultra-capacitor, (ii) periodically measuring the energy charged into the ultra-capacitor, and (iii) logging data. Since the companion node is powered by a separate energy source, it does not have energy constraint during a short experiment period. The information captured in the first two functions can either be written into the companion node's flash memory or transmitted via radio, supporting either off-line or on-line analysis and debugging. All the measurements conducted by the companion node is protected by voltage follower circuits to avoid load effects and achieve isolation.

We note that the working node can run alone without the companion node. It only needs to monitor its energy status at a low frequency for control purpose. The main purpose of adding the companion node is to accurately monitor our system's performance at a high frequency for evaluation purpose. By using a companion node, the *observer affect* can be avoided, otherwise the working node has to monitor and record energy status by itself. These operations consume a lot of energy that is not part of workload, causing inaccuracy in performance evaluation.

4.4 Summary of Hardware Design

The TwinStar node has a battery-free hardware structure using a combination of an ultra-capacitor and solar panels. It has several features that match our requirements:

• ***Stability***: To the best of our knowledge, it is the first design that considers the zero-energy boot up challenge and uses a dual-panel structure comprising a boot panel and a harvesting panel. The boot panel ensures stable power harvesting under extremely low ambient energy situations. The harvesting panel is used as the main charging device for the ultra-capacitor. DC/DC converter provides stable output to power the working node.

• ***Visibility***: As an experimental platform, the TwinStar node provides monitoring/debug interfaces for attaching an additional companion node (as shown in Figure 9), which can help us understand the behaviors of the running working node and facilitate system debugging and performance evaluation. The separate power supply for the companion node ensures accurate characterizing of the energy profile of the working node. In the commercial release version, this companion node can be easily removed to reduce cost.

5. SYSTEM ENERGY MODELING

System modeling is an essential step for effective control. In this section, we introduce three models: the energy harvesting model (Section 5.1), the energy consumption model (Section 5.2) and the energy leakage model (Section 5.3). To achieve effective energy synchronization, we need to balance the interaction among these models into an equilibrium state.

5.1 Energy Harvesting Model

The energy harvesting model is built online using the measurement switch discussed in Section 4.2. The total energy E_E harvested during the time interval $[(k-1)T, kT]$ is:

$$\Delta E_E[kT] = \int_{(k-1)T}^{kT} I_{cap}(t) \cdot V_{cap}(t) dt \quad (1)$$

Here $V_{cap}(t)$ is the ultra-capacitor's voltage at time t. Both $I_{cap}(t)$ and $V_{cap}(t)$ can be measured by the ultra-low-power measurement switch shown in Figure 11. T is called the sampling *period* and k is called the *sampling instant*.

5.2 Energy Consumption Model

The energy consumption of the sensor node is determined by three parameters: average active mode current I_{active}, average sleep mode current I_{sleep} and duty cycle D (the percentage of active time). The energy consumption during the time interval $[(k-1)T, kT]$ can be represented as:

$$\Delta E_C[kT] = V_{node} \cdot (I_{active} \cdot D + I_{sleep} \cdot (1-D)) \cdot T \quad (2)$$

The average power consumption at the end of time kT can be represented as follows:

$$P_{C_{AVG}}(kT) = \frac{\Delta E_C[kT]}{T} \quad (3)$$

5.3 Energy Leakage Model

Based on our empirical study in Section 2, we formulate a leakage model to characterize the relationship between the remaining energy in the ultra-capacitor and the leakage power. To minimize the environmental effect, such as

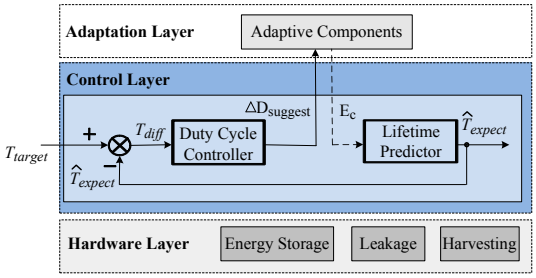


Figure 13: Control Layer Design Overview

temperature, the leakage model is normally obtained off-line and can be adjusted online after deployment.

To conduct off-line modeling, the harvesting circuit is shut down and the node executes a testing program at a constant duty cycle. At the end of every T seconds, the control layer periodically monitors the remaining voltage $V_{cap}(t)$ of the ultra-capacitor. Therefore, the difference of remaining energy $\Delta E_R[kT]$ during the time interval $[(k-1)T, kT]$ can be calculated as:

$$\begin{aligned}\Delta E_R[kT] &= \frac{C}{2}\left(V_{cap}((k-1)T)^2 - V_{cap}(kT)^2\right) \\ &= E_R\left((k-1)T\right) - E_R(kT) \end{aligned} \quad (4)$$

Without energy harvesting, $\Delta E_R[kT]$ is the sum of the energy consumed by the node and energy leaked away during time interval $[(k-1)T, kT]$, namely

$$\Delta E_R[kT] = \Delta E_C[kT] + \int_{(k-1)T}^{kT} P_L(t)dt$$

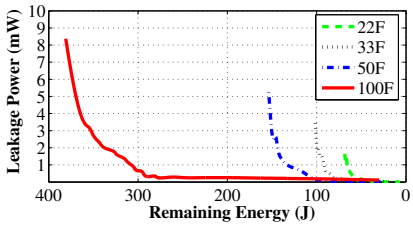
For small value of T, the above equation can be simplified as:

$$\Delta E_R[kT] = \Delta E_C[kT] + P_L(kT)\cdot T \quad (5)$$

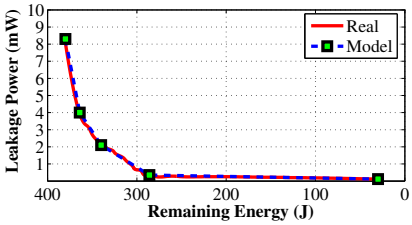
By combining Equations (4) and (5), at the end of any given time kT, the leakage power P_L can be calculated by the sensor node as follows:

$$P_L(kT) = \frac{E_R((k-1)T) - E_R(kT) - \Delta E_C[kT]}{T} \quad (6)$$

The sensor node builds the leakage model by storing the value of $P_L(kT)$ and the corresponding $E_R(kT)$ value in its memory. Figure 12(a) shows the empirical leakage data of the leakage model for diverse capacitors. To express the model mathematically and reduce the storage space for the



(a) Empirical Leakage Pattern



(b) Leakage Model Fitting

Figure 12: Leakage Model

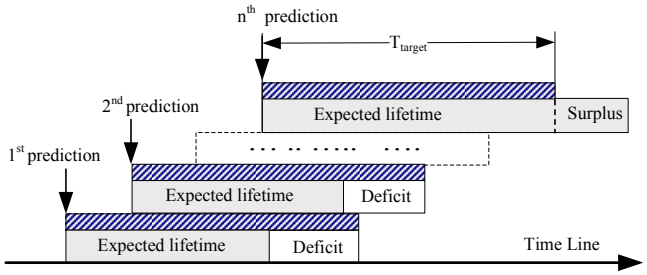


Figure 14: Sustainability

model, we use a piecewise linear approximation of the leakage curve. We define *turning points* as the points in the curve with considerable slope change, which are used to decide the start and end points of line segments. Therefore, the whole leakage model can be represented by a set of linear functions, as shown in Eqn.(7).

$$P_L(E_R) = \begin{cases} a_1 \cdot E_R + b_1; & E_{R_1} \le E_R < E_{R_2} \\ a_2 \cdot E_R + b_2; & E_{R_2} \le E_R < E_{R_3} \\ \vdots \quad ; & \vdots \\ a_n \cdot E_R + b_n; & E_{R_n} \le E_R \le E_{R_{n+1}} \end{cases} \quad (7)$$

Here E_{R_1}, E_{R_2},$\cdots$, $E_{R_{n+1}}$ are the remaining energy values corresponding to the turning points of the segments. a_1, a_2,$\cdots$, a_n and b_1, b_2,$\cdots$, b_n are the coefficients for each line segment. Figure 12(b) shows an example for the line-segment-based modeling of the leakage power. The square dots represent the turning points. The model we built accurately matches the empirical results.

6. CONTROL LAYER DESIGN

The design objective of the control layer is to *consume as much energy as possible while maintaining sustainability.* Sustainability T_{target} is defined as the duration a node must survive without ambient energy. T_{target} is a configurable parameter, set by users according to the environment in which sensors are deployed. For example, in energy-rich environments such as deserts, users can set a smaller T_{target} to consume energy aggressively. On the other hand, in energy-poor and unpredictable environments, a larger T_{target} is desired to ensure the aliveness of sensor nodes. Once T_{target} is chosen, it is used as the set point in the feedback control based design as shown in Figure 13.

As shown in Figure 14, to maintain sustainability, the lifetime predictor periodically predicts the expected lifetime $\widehat{T}_{expect}$ and compares $\widehat{T}_{expect}$ with the target lifetime T_{target}. In the case of energy deficit (i.e., $\widehat{T}_{expect} < T_{target}$), a node is expected to run out of energy after $\widehat{T}_{expect}$ seconds, if it maintains current level of activity. In the case of energy surplus (i.e., $\widehat{T}_{expect} > T_{target}$), a node can increase the activity accordingly for application performance improvement. As shown in Figure 13, the change of activity is achieved by using the difference between $\widehat{T}_{expect}$ and T_{target} as the input to the duty cycle controller, which suggests a certain percentage of duty cycle change to the adaptation layer. Unlike conventional control designs, the duty cycle is *suggested* by the control layer to the adaptation layer. The adaptation layer adjusts the duty cycle based on the application's requirement. This allows a more flexible design of the energy adaptation layer. The rationale behind this is that the short-term duty cycle available might not always be synchronized with the activity demanded by the applications.

For example, a node should not stop sending critical control messages disruptively simply because the control layer suggests to reduce duty cycle due to the brief drop in energy supply. Short-term mismatching is tolerable because the energy storage unit (e.g., ultra-capacitor) can serve as a buffer.

The control layer contains two major components: the lifetime predictor and the duty cycle controller, as shown in Figure 13. These are described in the following subsections.

6.1 Design of the Lifetime Predictor

To ensure that sensor nodes can survive dark periods during which no ambient energy is available (e.g., night), it is essential to predict lifetime conservatively based on (i) the remaining energy, (ii) the energy leakage rate, and (iii) the energy consumption rate. Since the available environmental energy changes dynamically, when do prediction, we conservatively assume zero energy from the environment in the future (note: current energy harvested can be measured and therefore is not assumed to be zero).

As discussed in Section 5.3, the leakage power changes along with the voltage of an ultra-capacitor. The higher the voltage, the larger leakage power an ultra-capacitor suffers. To precisely estimate the expected lifetime $\widehat{T}_{expect}$, we need to iteratively estimate the amount of energy leaked away.

As shown in Figure 15, a prediction is conducted at time t_0. The predictor initializes $\widehat{T}_{expect} = 0$, then goes through an iterative process. During each iteration, the predictor extends the value of the expected lifetime $\widehat{T}_{expect}$ by ΔT and predicts how much energy will be consumed during the ΔT period. The iterative prediction stops, when the expected remaining energy is below the minimum energy required to keep the sensor node alive. Suppose the number of total iterations is N, the expected lifetime $\widehat{T}_{expect} = N\Delta T$. More specifically, the prediction goes through the following steps.

Step 1: Measure Remaining Energy

At the time t_0, the remaining energy $\widehat{E}_R(t_0)$ can be obtained by sampling the voltage V_{cap} of the ultra-capacitor:

$$\widehat{E}_R(t_0) = \frac{1}{2}CV_{cap}^2 \text{ when } n = 0 \tag{8}$$

Step 2: Predict the Leakage Power at Time $t_0 + n\Delta T$

Based on the leakage model specified in Eqn. 7, the predicted leakage power at time $t_0 + n\Delta T$ is estimated as:

$$\begin{aligned} &\widehat{P}_L(t_0 + n\Delta T) = a_i \cdot \widehat{E}_R(t_0 + n\Delta T) + b_i \\ &where\ E_{R_i} \le \widehat{E}_R(t_0 + n\Delta T) < E_{R_{i+1}}, n \ge 0 \end{aligned} \tag{9}$$

Where ΔT is the time step used by the predictor to conduct the prediction operation.

Step 3: Predict Remaining Energy at $t_0 + (n+1)\Delta T$

Assuming no environmental energy, the predicted remaining energy at time $t_0 + (n+1)\Delta T$ is:

$$\begin{aligned} &\widehat{E}_R(t_0 + (n+1)\Delta T) = \\ &\widehat{E}_R(t_0 + n\Delta T) - \int_{t_0+n\Delta T}^{t_0+(n+1)\Delta T} \Big(P_C(t) + \widehat{P}_L(t)\Big)\, dt \end{aligned} \tag{10}$$

For simplicity, the above prediction is approximated as:

$$\begin{aligned} &\widehat{E}_R(t_0 + (n+1)\Delta T) \approx \\ &\widehat{E}_R(t_0 + n\Delta T) - \Big(P_C(t_0 + n\Delta T) + \widehat{P}_L(t_0 + n\Delta T)\Big)\, \Delta T \end{aligned} \tag{11}$$

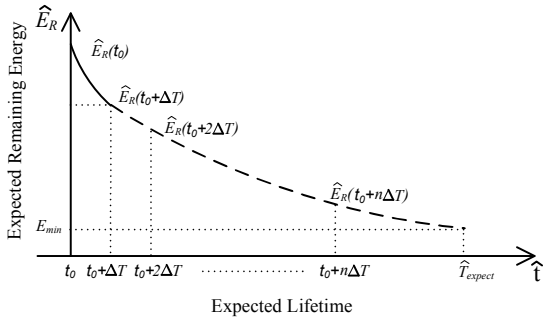


Figure 15: Iterative Prediction

Step 4: Test and Jump

As shown in Figure 15, the prediction process is iterative, in which the remaining energy $\widehat{E}_R(t_0 + (n+1)\Delta T)$ is predicted based on the prediction result of $\widehat{E}_R(t_0 + n\Delta T)$. The predicted lifetime of a node ends when there is no sufficient energy to sustain its basic operations. Namely, if $\widehat{E}_R(t_0 + (n+1)\Delta T) < E_{min}$, the lifetime predictor returns $n\Delta T$ as the expected lifetime $\widehat{T}_{expect}$ of the node. Otherwise, $n = n + 1$ and the lifetime predictor goes back to Step 2.

We note the prediction result is affected by power consumption pattern $P_C(t)$. With a constant energy consumption rate (i.e., $P_C(t) = c$), more energy is leaked during the high voltage stage. For a leakage-aware design, it is necessary to adjust the power consumption pattern $P_C(t)$ based on the value of the leakage power $P_L(t)$. In general, we shall increase power consumption when the $P_L(t)$ value is high and reduce power consumption when the $P_L(t)$ value is low. This type of adjustment is achieved in the controller design introduced in the following sections.

6.2 Reducing Prediction Overhead

Our prediction is based on the approximations that the predicted leakage power value $\widehat{P}_L(t)$ does not change during two consecutive iterations. In order to increase the accuracy of prediction, a smaller ΔT value is desired. However, a smaller ΔT value leads to more iterations, resulting in more energy consumption in computation. Clearly, it is difficult to reconcile the conflict between prediction accuracy and computation overhead with a single ΔT value. It should be noted that the leakage power value changes quickly when voltage is high but changes slowly when voltage is low, as shown earlier in Figure 12(a). Therefore, the prediction of the lifetime can be scheduled along the predicted timeline with increasing ΔT values (decreasing frequency). This *adaptive prediction* allows the node to estimate accurately in high leakage power stage and efficiently in low leakage power stage. In our implementation, using adaptive prediction, the maximum number of iterations needed is 305. Compared to the approach using a single small ΔT value, which needs 10,389 iterations, our adaptive predictor can reduce 10084 extra iterations, while maintaining a 99% accuracy.

6.3 Design of the Duty-Cycle Controller

In this section, we describe the design of the controller in the system. The controller indicates the suggested change of duty cycle $\Delta D_{suggest}$ to the adaptation layer based on the remaining energy inside the ultra-capacitor and the time difference $T_{diff} = \widehat{T}_{expect} - T_{target}$. Here we use a P-controller, considering the fact that the bigger the difference T_{diff}, the larger ΔD is needed to adjust.

The major issue for the P-controller design is to choose an appropriate control gain P, so that the sensor node can reduce the amount of energy leaked away and still meet the lifetime goal of T_{target}. Intuitively, if the sensor node increases its duty cycle at the ultra-capacitor's high leakage power stage, it can *capture* energy that would otherwise leaked away. To illustrate the design, consider two scenarios:

Energy Surplus: In the case of an energy surplus, $\widehat{T}_{expect}$ is larger than T_{target}, we get a positive T_{diff} value. To converge $\widehat{T}_{expect}$ to the set point T_{target}, a positive duty-cycle change that can increase the power consumption $P_C(t)$ needs to be suggested to the adaption layer. Clearly, it is desirable to increase more duty-cycle when the leakage power value $P_L(t)$ is high. Therefore, in the case of an energy surplus, we used the following adaptive P-controller:

$$\Delta D_{suggest} = G_+ \cdot P_L \cdot T_{diff} \tag{12}$$

where G_+ is a coefficient for control gain adjustment in the case of an energy surplus.

Energy Deficit: In the case of an energy deficit, $\widehat{T}_{expect}$ is smaller than T_{target}, and we get a negative T_{diff} value. To converge $\widehat{T}_{expect}$ to the set point T_{target}, we shall reduce the power consumption $P_C(t)$ by suggesting a negative duty cycle change to the adaption layer. It is desirable to keep a high duty cycle when the leakage power $P_L(t)$ is high. Therefore, in the case of an energy deficit, we would like to reduce the duty cycle slowly in the ultra-capacitor's high leakage power stage. The following adaptive P-controller is used:

$$\Delta D_{suggest} = \frac{G_-}{P_L} \cdot T_{diff} \tag{13}$$

where G_- is a coefficient for control gain adjustment in the case of an energy deficit.

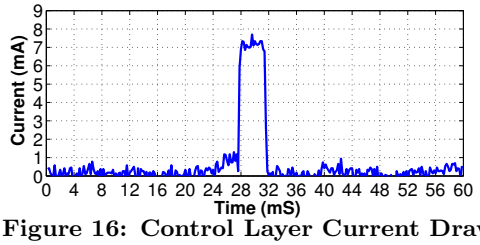


Figure 16: Control Layer Current Draw

6.4 Computation Overhead

The duty-cycle control is an online process, hence its overhead should be carefully considered. Clearly, the overhead of the control layer is determined by the overhead of each control operation and the frequency of control. In each control operation, a node needs to predict the lifetime and use the difference between the expected lifetime and targeted lifetime to control the duty cycle change. The active duration of each control operation is very small. For instance, Figure 16 shows the profile of current draw measured with an oscilloscope. Each control operation draws approximately $7.2mA$ for about $4ms$, which can be translated into $86.4\mu J$ per control operation. The frequency of control operation in our design is at the order of minutes (e.g., 5 minutes by default). Compared to the typical energy consumption of a sensor device (e.g., MicaZ), energy consumption of the control layer adds only 0.0013% overhead to the system.

7. ADAPTATION LAYER DESIGN

The high-level goal of the adaptation layer is to optimize the system performance by adjusting the system activities, such as sensing and communication, based on the application's requirement and the suggested duty-cycle from the control layer. Specifically, let the active instance of a sensor node be a tuple (t, d), where t is the start time of the active instance and d is the corresponding duration of this active instance, the working schedule of a node can be represented as $\Gamma = \{(t_1, d_1), (t_2, d_2), ...\}$. Essentially, Γ stores all active instances for a node. With the suggested duty-cycle from the control layer, a node can increase or decrease its duty cycle by either increasing/reducing the number of active instances or extending/shrinking the active duration. For sensor activities such as sensing or communication, such increase or decrease of duty cycle at a single node could significantly impact the whole system performance. To optimize the system performance and make the best use of available energy, the adaptation layer conducts two types of energy synchronization operations: (i) local energy synchronization and (ii) global energy synchronization.

Duty-cycle-based adaption is generic enough to be applied in many protocols. Due to space constraints, in the rest of this section, we can only present an exemplary adaptation layer design for the *event detection* application as a case study to demonstrate how the energy synchronization operations are conducted. Other designs can be accommodated in the future. For most of the sensing applications, the working schedules of sensor nodes are periodic. For example, a periodic working schedule $\{(1, 3), (101, 3), (201, 3), ...\}$ represents the sensor node is active for 3 active durations every 100 units of time, The *duty cycle* of a sensor node, therefore, is the ratio between the total number of active durations within a period and the time duration of the period. In the above example, the duty cycle is 3%.

7.1 Local Energy Synchronization

In local energy synchronization, an individual sensor node adjusts its own duty cycle based on the suggestion from the control layer. For the event detection application, an individual sensor node's active instances should be evenly scattered within a period to minimize the expected detection delay of a target within its sensing range. For example, assume the initial duty cycle is 1% and the duration of a period is 100 units of time, the adaptation layer can set the working schedule to be $\Gamma = \{(0, 1), (100, 1), ...\}$. The expected detection delay is $\frac{100}{2} = 50$. When the control layer suggests to change the duty cycle to 2%, to minimize the expected detection delay, the adaptation layer can set the working schedule to be $\Gamma = \{(0, 1), (50, 1), ...\}$. The expected detection delay is reduced to $\frac{50}{2} = 25$. In this example, we can see that the adaptation layer's local energy synchronization is important. By doing local energy synchronization, the sensor node makes best use of available energy and reduces the expected detection delay from 50 to 25.

7.2 Global Energy Synchronization

In global energy synchronization, multiple sensor nodes inside the network cooperatively adjust their duty cycles to optimize the system performance. In many applications, multiple sensors are deployed to monitor the same region to enhance detection fidelity. Under such scenarios, those sensor nodes should coordinate their working schedules with each others' available duty cycle budget. Specifically, the nodes that monitor the same region can divide the whole period into several sections. The number of sections is equal to the number of these nodes. Each node only increases or decreases the number of active instances within its own

sections. In this way, no energy is wasted for the nodes to exchange information to coordinate their schedules and avoid being active at the same time. For example, assuming nodes a and b are deployed to monitor a region and the initial duty cycle for node a and node b is 1% each. As shown in Figure 17(a), the initial working schedule for node a and b is $\Gamma_a = \{(0,1),(100,1),...\}$ and $\Gamma_b = \{(50,1),(150,1),...\}$, respectively. The expected detection delay in the common sensing region of node a and b is $\frac{50}{2} = 25$. When the control layer suggests node a to increase the duty cycle to 2%, without global energy synchronization, to minimize the detection delay the adaptation layer will add active instances in the middle of node a's initial active instances, as shown in Figure 17(b). These newly added active instances overlap with node b's initial active instances. Therefore, the expected detection delay in the common sensing region of node a and b remains the same. However, with global energy synchronization, node a's adaptation layer will evenly distribute the active instances within node a's sections (i.e., from 0 to 49), as shown in Figure 17(c). In this way, the expected detection delay in the overlapping region of node a and b reduces from 25 to 18.75.

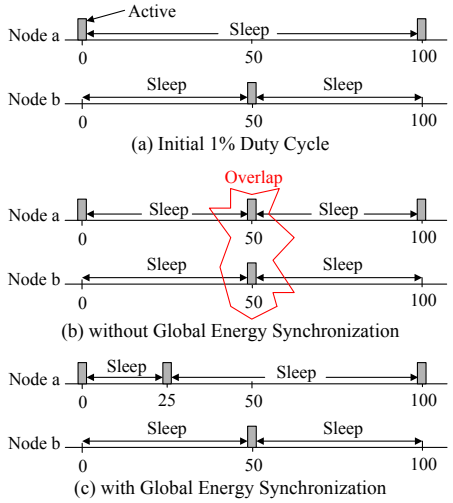


Figure 17: Global Energy Synchronization Example

The above example shows that in addition to local energy synchronization, global energy synchronization is important for the adaptation layer to efficiently utilize the available energy suggested from the control layer and optimize the network level system performance as a whole.

8. EVALUATION I: LOCAL ENERGY SYNCHRONIZATION

In this section, we evaluate system performance of a single TwinStar node under different types of environmental energy patterns. Since this work is the first one investigating leakage-aware control, the state-of-art (e.g, energy harvesting and conservation) is complementary, however, provides no appropriate baselines for comparison. Therefore, we compare our system working in Leakage-Aware (LA) mode with the same system but in a Non-Leakage-Aware (NLA) mode. The NLA mode has no information on the leakage rate of the ultra-capacitor. In other words, the system in the non-leakage-aware mode predicts the expected lifetime and controls the duty cycle of the working node, based on the current energy consumption and remaining energy inside the ultra-capacitor. Without considering the leakage profile, the feedback controller in the non-leakage mode uses a fixed P-controller for both the energy surplus and energy deficit scenarios.

We also compare the LA and NLA systems with an *oracle system* that assumes the knowledge of all the available environmental energy in the future. The oracle system runs offline, based on the empirical environmental energy collected under three different scenarios (described in Section 8.3).

8.1 Evaluation Metrics and Baseline

The key advantage of the leakage-aware design is efficiently using energy that would possibly leak away. We use two metrics to evaluate the performance of our system, (i) **Cumulated Active Time (CAT):** the cumulated active time of a sensor node. Under a given energy harvesting pattern, the larger the CAT value, the more work a sensor node can perform. (ii) **Leakage to Consumption Ratio (LCR):** the ratio between leakage power value and power consumption value at any given time. The smaller the LCR value, the less energy leaked away.

8.2 Implementation

We designed and implemented the adaptation layer and the control layer using TinyOS and NesC. The compiled image of a full MicaZ node implementation occupies 17,894 bytes of code memory and 368 bytes of data memory.

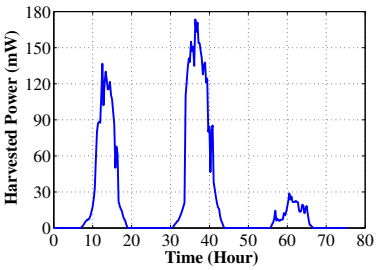
As shown in Figure 8 in Section 3, our hardware contains a custom circuit board to harvest the energy and two off-the-shelf sensor nodes (e.g., MicaZ nodes): (i) a working node, which is powered by the ultra-capacitor, and (ii) a companion node, which is powered by batteries. The companion node periodically wakes up to measure the energy harvesting rate and the remaining voltage inside the ultra-capacitor for evaluation purpose. Measurements are logged into the flash and are used to conduct off-line analysis of system performance. We note that the companion node has a separate power supply and acts as the observer for our system. The working node is the node that actually runs our system. It also periodically wakes up to sample the remaining voltage of the ultra-capacitor for the purpose of feedback control. The predictor and controller are invoked periodically and get the suggested duty cycle ΔD. The adaption layer adjusts the node's duty cycle based on the suggested ΔD by extending the length of active instances.

8.3 Different Deployment Scenarios

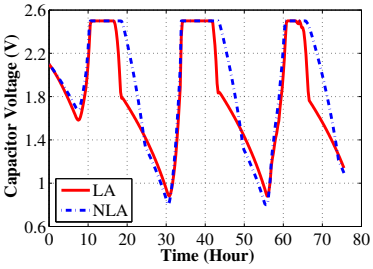
We ran our system under three different scenarios: outdoor, indoor, and mobile backpack, as shown in Figures 18, 19 and 20, respectively. These scenarios are carefully selected to represent a wide range of energy harvesting patterns: (i) periodical and vibrated energy for the outdoor environment, (ii) periodical and constant energy for the indoor environment, and (iii) dynamic and highly unpredictable energy in the mobile environment. For each scenario, we used two sets of hardware, one running in the leakage-aware mode, the other in the non-leakage-aware mode. We put these two sets of hardware near each other so that they harvested a similar amount of environmental energy. The working node woke up every 5 minutes in all three scenarios. The companion node woke up every 15 seconds for the outdoor and indoor experiments. For the backpack experiment, since the environmental energy changed very frequently, the companion node woke up every 1 second. In all the scenarios, the MicaZ nodes were initially working at a 1% duty cycle.

Outdoor

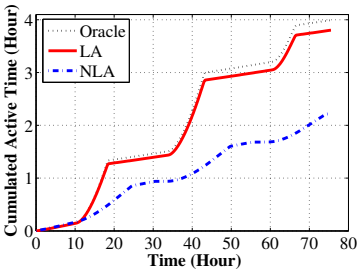
(a) Experimental Site



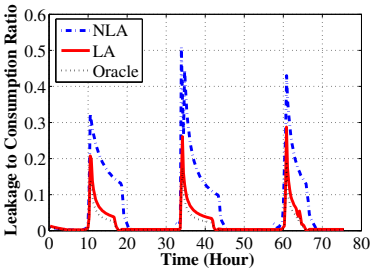
(b) Harvested Power



(c) Capacitor Voltage



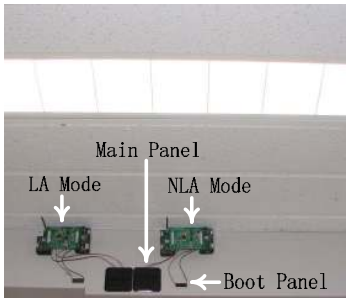
(d) Cumulated Active Time



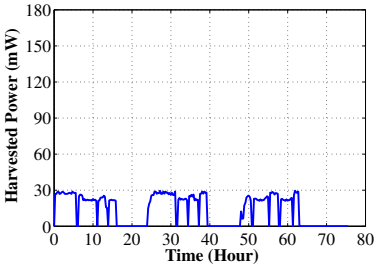
(e) Leakage to Consumption Ratio

Figure 18: Outdoor

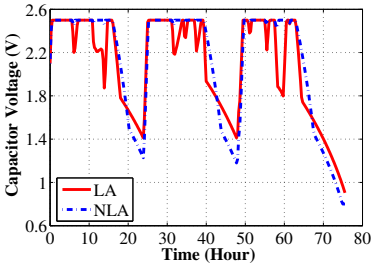
Indoor



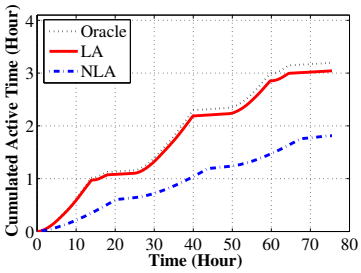
(a) Experimental Site



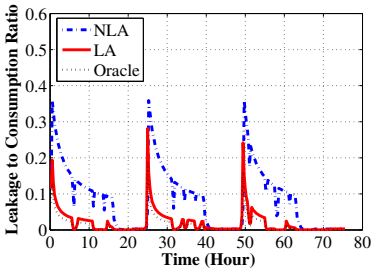
(b) Harvested Power



(c) Capacitor Voltage



(d) Cumulated Active Time

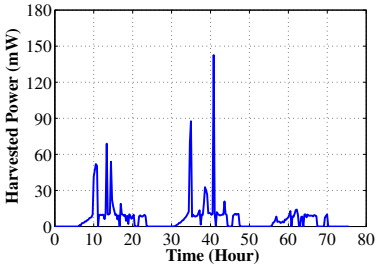


(e) Leakage to Consumption Ratio

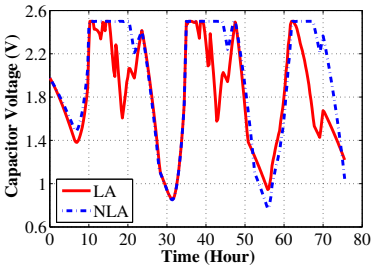
Figure 19: Indoor

Mobile Backpack

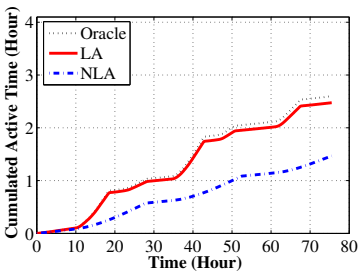
(a) Experimental Site



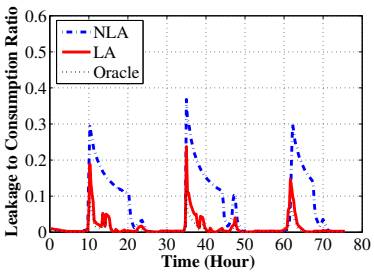
(b) Harvested Power



(c) Capacitor Voltage



(d) Cumulated Active Time



(e) Leakage to Consumption Ratio

Figure 20: Mobile Backpack

8.4 Outdoor Experiment

The outdoor scenario represents such potential applications as environment monitoring [22] and scientific exploration [9]. In the outdoor experiment, we deployed our system outside of a fifth-floor apartment, as shown in Figure 18(a). We put TwinStar nodes outside the window, facing south. The total time duration of this experiment was 76 hours.

Figure 18(b) shows the energy harvested by the system in the leakage-aware and non-leakage-aware modes over the 76 hours. We placed two systems close to each other to harvest similar amounts of energy from the environment. By inspecting Figure 18(c) closely, one can discover that the leakage-aware system consumes energy quickly when the voltage is high (i.e., leakage is high) and it slows down the consumption when the voltage is low. This reduces the amount of energy leaked away. As a result, at the end of the experiment, the capacitor's voltage for the leakage-aware system was 1.14V, higher than the capacitor's voltage for the non-leakage-aware system, which was 1.09V.

Figure 18(d) compares the cumulated active time among the leakage-aware, non-leakage-aware, and oracle systems. Clearly, the leakage-aware system allows nodes to consume more energy over time than the non-leakage-aware system. Since the oracle system have the knowledge of the available environmental energy in the future, it can more aggressively consume energy to reduce the energy leaked away. However, the curves of leakage-aware and oracle systems are very close to each other. Moreover, it's very difficult to predict the available environmental energy in the future. For example, in Figure 18(b), the available environmental energy on the third day is $5-6$ times less than on the previous two days. By inspecting Figure 18(d) carefully, one can observe that the leakage-aware system exhibits a certain stair effect. This is because of the changing duty cycles of the adaption layer according to the energy harvesting pattern, as shown in Figure 18(b). Obviously, the leakage-aware system is very effective. For example, after 70 hours, the leakage-aware system allowed a node to stay active 70% longer than did the non-leakage-aware system.

Figure 18(e) shows the Leakage to Consumption Ratio (LCR) values of the systems running in the oracle, leakage-aware, and non-leakage-aware modes over the 76 hours. By increasing the energy consumption at the high leakage power stage and reducing the energy consumption at the low leakage power stage, the oracle and leakage-aware systems always maintain the LCR value at very low level. The LCR value of the non-leakage aware system is more than twice larger than that of the leakage aware system.

8.5 Indoor Experiment

The indoor scenario represents such potential applications as facility management, structure monitoring. As shown in Figure 19(a), our system was deployed under the overhead light in our lab. The total time duration for this experiment was 72 hours. The light was turned on in the morning, when people came to the lab, and turned off in the middle of the day or during the night when no one was inside the lab.

Figure 19(b) shows the energy harvested by the systems working in leakage-aware and non-leakage-aware modes over the 72-hour period. The fluctuation of the energy level was due to the turned on or off of the neighboring overhead lights. These two systems harvested similar amounts of energy from the environment, but they had different amounts of cumulated active time, as shown in Figure 19(d). The cumulated active time for both leakage-aware and non-leakage-aware systems in the indoor scenario was smaller than that of the outdoor scenario because in the indoor scenario, the total available energy was smaller. For the same reason, the gap of cumulated active time between the leakage-aware system and oracle system was smaller than that of the outdoor scenario. Figure 19(e) shows again that compared with the non-leakage aware system, the leakage aware system has much smaller value of LCR during the high leakage power stage than the non-leakage-aware system.

8.6 Mobile Backpack Experiment

We also deployed our system on a backpack, as shown in Figure 20(a). It was carried by a graduate student every day for 3 days. During the night and in the early morning, the backpack was placed near a living room window to harvest environmental energy through the window. During the day, the graduate student carried it to attend outdoor and indoor activities. This scenario represents such potential applications as assisted living and human-centric sensing [8]. The total time duration of this experiment was 76 hours.

Figure 20(b) shows the energy harvested by the systems working in the leakage-aware and non-leakage-aware modes over the 76 hours. Compared with the outdoor and indoor cases, the energy harvested in the backpack experiments was very bursty, with the high peak corresponding to outdoor activity and the flat part corresponding to indoor activity. Figure 20(c) shows the capacitor's voltage measured over time. Clearly, the leakage-aware system was more responsive to the change in the energy harvesting rate, increasing duty cycle rapidly in the presence of bursty influxes of energy. Therefore the leakage-aware system stayed at high voltage values for a smaller amount of time, hence suffering less energy leakage. At the end of the experiment, the voltage of the capacitor for the leakage-aware system was 0.19V higher than that of the non-leakage-aware system. Figure 20(d) shows again that the leakage-aware system enjoys much more cumulated active time than the non-leakage-aware system. Figure 20(e) confirms the above results.

8.7 Summary

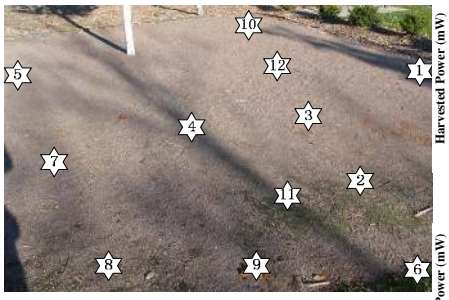
We have evaluated our system under different environmental energy patterns. In all these scenarios, our leakage-aware system outperforms the non-leakage-aware system. The key observations are: (i) the working node never runs out of power (note: minimum voltage to maintain liveness is 0.7V); (ii) the leakage-aware design is more responsive to the influx of energy, increasing duty cycle quickly; and (ii) the leakage-aware system maintains a high duty cycle when the voltage is high and a low duty cycle when the voltage is low. This type of control effectively reduces the amount of leakage.

9. EVALUATION II: GLOBAL ENERGY SYNCHRONIZATION

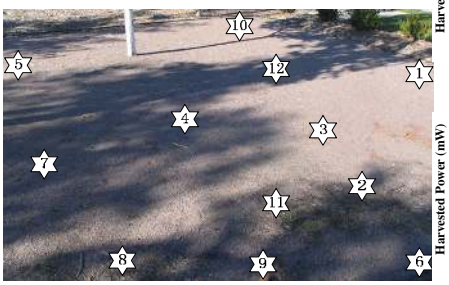
In this section, we evaluate the system performance of a network of TwinStar nodes under the global energy synchronization scheme. The event detection application is run on top of the adaptation layer.

9.1 Evaluation Metrics

In event detection application, the detection delay is a very important metric to evaluate system performance. For the events such as fire, a shorter detection delay significantly reduces the amount of damage. The metrics used to evalu-

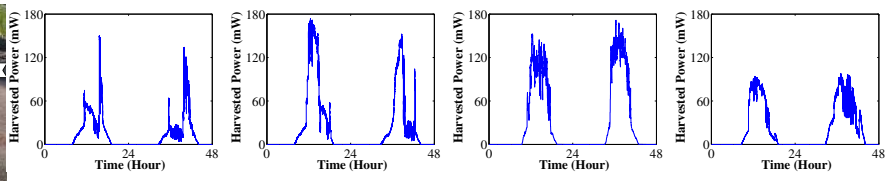


(a) Morning

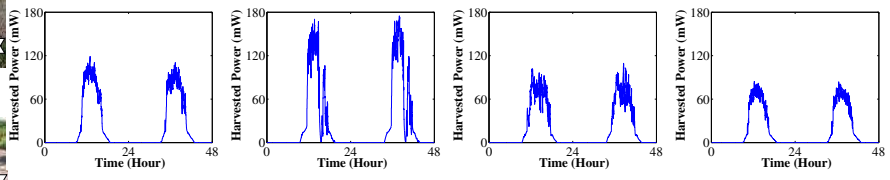


(b) Afternoon

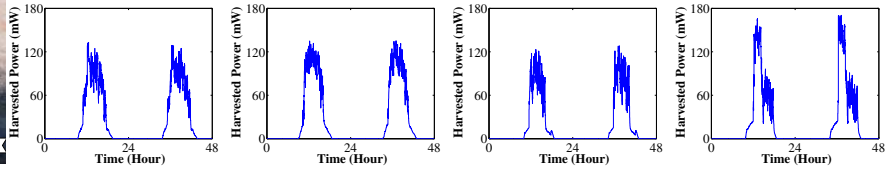
Figure 21: Experiment Sites



(a) Node 1 (b) Node 2 (c) Node 3 (d) Node 4



(e) Node 5 (f) Node 6 (g) Node 7 (h) Node 8



(i) Node 9 (j) Node 10 (k) Node 11 (l) Node 12

Figure 22: Distribution of Harvested Power

ate the performance of our system is the **Average Detection Delay:** the summation of the detection delay of all the events divided by the total number of detected events.

9.2 Implementation

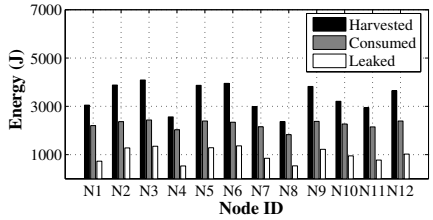
We deployed 12 TwinStar nodes in a 12 meters by 11 meters wooded area (shown in Figure 21) to continuously collect solar energy (shown in Figure 22) for two days. The collected energy pattern is used to power 12 TwinStar nodes to work under 3 different modes: Leakage-Aware (LA), Non-Leakage-Aware (NLA), and oracle, with 48 hours per mode. The sensing irregularity model [14] is used. During the 48 hours' experiment, 500 events are randomly emulated within the 12m × 11m area.

9.3 Distribution of Energy

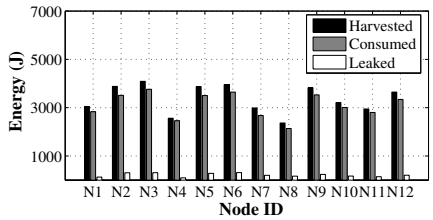
Figure 23 shows the energy distribution among these 12 nodes, after they run for 48 hours in the LA, NLA, and oracle modes, respectively. Figure 23(a) shows that when the nodes work in NLA mode, regardless of available environmental energy, all the nodes consume similar amount of energy. The extra harvested environmental energy is leaked away. As shown in Figure 23(b), when the nodes work in LA mode, the energy they consumed is synchronized with the amount of energy they harvested from the environment. The more energy they harvested, the more energy they consumed. By comparing Figure 23(b) with Figure 23(c), the nodes work in LA mode and oracle mode have similar performance.

9.4 Impact of Harvested Power Fluctuation

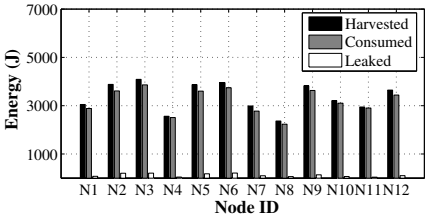
Since the available environmental energy varies every day, in this section, we evaluate the impact of harvested power fluctuation to the systems work in LA, NLA, and oracle mode, respectively. We use the energy pattern collected (shown in Figure 22) as a baseline, and generate an energy pattern in which the harvested power is varied from 20% to 200% of the baseline for each TwinStar node. The metric used is the average detection delay (ADD). As shown in Figure 24, when the harvested power increases, the systems work in all the 3 modes have smaller average detection delay, and the gap of the ADD among these 3 modes increases.



(a) Non-Leakage Aware Nodes



(b) Leakage Aware Nodes



(c) Oracle Nodes

Figure 23: Distribution of Energy

The reason is that with more energy harvested, more energy leaked away in NLA mode. When the harvested power increases to 200% of the baseline, the ADD of LA is 27% less than that of NLA, while the ADD of oracle is only 9% less than that of LA.

9.5 Impact of Energy Period

Different environments have different period of available environmental energy. For example, during the winter season, the solar panel may be covered with snow, which pre-

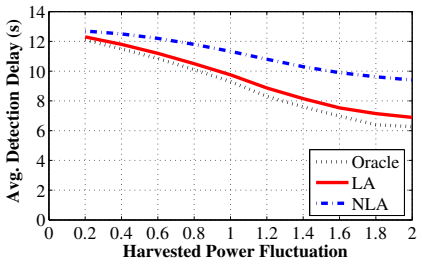


Figure 24: Impact of Harvested Power Fluctuation

vents TwinStar nodes from harvesting the environmental energy effectively. TwinStar node can only resume to harvest the energy after the snow on the solar panel melts. In this section, we evaluate the impact of energy period to the systems work in LA, NLA, and oracle mode, respectively. We use the energy pattern collected (shown in Figure 22) as a baseline, and vary the energy period from 12 hours to 48 hours. Average detection delay is used to evaluate the system performance. Figure 25 shows that when the energy period increases, the systems working in all the 3 modes have higher average detection delay, and the gap of the average detection delay among these 3 modes decreases. This is because when the energy period increases, the TwinStar nodes work more conservatively. When the energy period is 12 hours, the average detection delay of the network works in LA mode is 48% less than that works in NLA mode. The performance of LA and oracle is very close to each other.

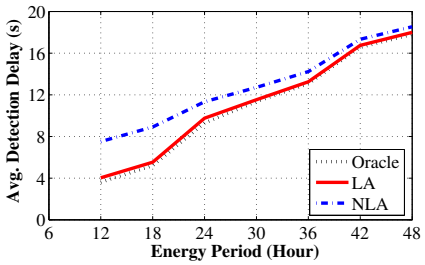


Figure 25: Impact of Environmental Energy Period

10. STATE OF THE ART

Energy harvesting is the conversion of ambient energy into usable electrical energy. Several technologies have been developed for extracting energy from the environment, including solar [30], wind [5], kinetic, and vibrational [26] energy. With these energy-harvesting technologies, researchers have designed various types of platforms to collect ambient energy from human activity or environments [20, 16, 5, 30]. Notable ones designed specially for sensor networks are Heliomote [16, 1], Prometheus [30], Trio [20], AmbiMax [5] and PUMA [4]. According to the type of energy storage used, these platforms can be separated into three categories: (i) rechargeable battery-based platforms [1], (ii) designs combining ultra-capacitors and rechargeable batteries [20, 30], and (iii) capacitor-based designs.

• In rechargeable battery designs, such as Heliomote [16, 1], the energy harvesting panel is directly connected to its battery. As the primary energy storage device, the rechargeable battery is charged and discharged frequently, leading to low system lifetimes due to the physical limitation on the number of recharge cycles.

• In designs that combine ultra-capacitors and rechargeable batteries such as Prometheus [30], the solar energy is first stored in the primary energy buffer, which is one or multiple ultra-capacitors. The rechargeable batteries are then used as the secondary energy buffer. This design inherits both the advantages and limitations of batteries and capacitors. For examples, it is difficult to predict remaining energy because of the inclusion of batteries, and the lifetime of energy storage subsystem is decided by the shelf time of batteries (in the order of a few years). Within this category, several similar systems have been built with a few enhanced features. For examples, AmbiMax [5] harvests energy from multiple ambient power sources (e.g., solar and wind generators), and PUMA [4] uses a power routing switch to route multiple power sources to multiple subsystems. The higher utilization of ambient power is achieved through a combination of MPPT and power defragmentation.

• To our best knowledge, previous works have intentionally avoided capacitor-only design, citing the leakage issue [30]. To alleviate leakage, small capacitors are normally used, which makes secondary energy storage (batteries) necessary. However, the development of battery capacities is very slow and still leakage-prone. In addition, charging efficiency for batteries is comparatively low [28]. For example, according to the Natural Resources Defense Council [28], common battery chargers provide an efficiency between 6% and 40%. Different from previous works, this work investigates the frontiers of capacitor-only design and studies not only hardware designs but also related software control techniques to reduce the impact of energy leakage. Energy conservation is an intensively studied area. Many solutions have been proposed at different layers, including high-efficiency hardware design [7], link layer design [32], topology management [24], node placement [6], network routing [17], sensing coverage [18], data aggregation [25], data placement [3, 27], operating system [11], and application-level energy-aware designs [12]. However, only a few works have focused on energy measurement [31] and energy adaptation [13, 10, 2]. Odyssey [13], and ECOsystem [10] demonstrate that application-level adaptations can successfully meet user-specified lifetimes for PDAs and laptops running Linux or other embedded OSes. The differences between these works and ours are that (i) they focus on battery modeling or application adaptations, while our work focuses on battery-free design and control, and (ii) they currently do not consider energy harvesting issues.

The most closely related work for energy adaption is Eon [15], which is the first programming language for energy-adaptive applications in wireless sensor networks. Using Eon, programmers can easily annotate flows in energy states. Eon's automatic energy management uses these annotations to change application behavior adaptively. Since Eon focuses on language and the adaption layer, it is highly complementary to the leakage-aware hardware and controller design in this work.

11. CONCLUSIONS

Slow development in battery technology and rapid advances in ultra-capacitor design have motivated us to investigate the possibility of using capacitors as the sole energy storage for wireless sensor nodes. In this work, we build TwinStar, an add-on power board that uses energy harvesting circuits (e.g., solar panels or wind generators) to harvest the energy from the environment and employs an ultra-capacitor as the only energy storage unit to power the sensor node. The Twinstar power board incorporates a smart control circuit to address the zero-energy boot-up

problem and an ultra-low-power measurement circuit to precisely measure the energy harvesting rate.

To promote the capacitor-only-design, this work takes a first step in reducing the impact of energy leakage on overall energy efficiency. With the leakage model built, we improve the accuracy in predicting the expected lifetime of an individual sensor node. To ensure long-term sustainability, we propose a feedback-based approach that suggests an appropriate duty-cycle change to the adaption layer, based on the gap between the predicted and targeted lifetimes. The controller increases activity when the leakage power is high. This type of leakage-aware control allows us to utilize the energy that would otherwise leak away.

We invested significant amount effort to evaluate our design in multiple real-world settings: (i) indoor, (ii) outdoor, and (iii) mobile backpack. We also studied the global synchronization with a network of TwinStar nodes. The results indicate the effectiveness of the leak-aware design compared with a design that ignores the leakage issue. In future work, we shall focus more on the design of the adaptation layer, so that a system can gracefully adjust activities for optimal user-level performance.

12. ACKNOWLEDGEMENTS

This work was supported in part by MSR Sensornet Research Platform Project, NSF grants CNS-0626614 and CNS-0845994. Zhi-Li Zhang is supported in part by CNS-0626812, CNS-0626808, and CRI 0709048. We would also like to thank our shepherd Dr. Richard Hankins and the reviewers for their valuable feedback.

13. REFERENCES

[1] A. Kansal, J. Hsu, S. Zahedi, and M. B. Srivastava, "Power management in energy harvesting sensor networks," *TECS*, vol. 6, no. 4, 2007.

[2] A. Lachenmann, P. J. Marrón, D. Minder, K. Rothermel, "Meeting lifetime goals with energy levels," in *SenSys '07*, 2007.

[3] S. Bhattacharya, H. Kim, S. Prabh, and T. Abdelzaher, "Energy-conserving data placement and asynchronous multicast in wireless sensor networks," in *MobiSys '03*, 2003.

[4] C. Park and P. Chou, "Power utility maximization for multi-Supply systems by a load-matching switch," in *ISLPED '04*, 2004.

[5] C. Park and P.H. Chou, "AmbiMax: Autonomous energy harvesting platform for multi-supply wireless sensor nodes," *SECON '06*, 2006.

[6] D. Ganesan, R. Cristescu, and B. Lozano, "Power-efficient sensor placement and transmission structure for data gathering under distortion constraints," in *IPSN '04*, 2004.

[7] D. Lymberopoulos and A. Savvides, "XYZ: a motion-enabled, power aware sensor node platform for distributed sensor network applications," in *IPSN '05*, 2005.

[8] S. B. Eisenman, E. Miluzzo, N. D. Lane, R. A. Peterson, G.-S. Ahn, and A. T. Campbell, "The bikenet mobile sensing system for cyclist experience mapping," in *SenSys '07*, 2007.

[9] G. Werner-Allen, K. Lorincz, J. Johnson, J. Lees, and M. Welsh, "Fidelity and yield in a volcano monitoring sensor network," in *OSDI '06*, 2006.

[10] H. Zeng, C.S. Ellis, and A.R. Lebeck, "Experiences in Managing Energy with ECOSystem," *Pervasive Computing, IEEE*, Jan.-March 2005.

[11] C.-C. Han, R. Kumar, R. Shea, E. Kohler, and M. Srivastava, "A dynamic operating system for sensor nodes," in *MobiSys '05*, 2005.

[12] T. He, S. Krishnamurthy, J. A. Stankovic, T. Abdelzaher, L. Luo, R. Stoleru, T. Yan, L. Gu, J. Hui, and B. Krogh, "Energy-efficient surveillance system using wireless sensor networks," in *MobiSys '04*, 2004.

[13] J. Flinn and M. Satyanarayanan, "Managing battery lifetime with energy-aware adaptation," *ACM Trans. Comput. Syst.*, vol. 22, no. 2, pp. 137–179, 2004.

[14] J. Hwang, T. He, and Y. Kim, "Exploring in-situ sensing irregularity in wireless sensor networks," in *SenSys '07*, 2007.

[15] J. Sorber, A. Kostadinov, M. Garber, M. Brennan, et al, "Eon: A language and runtime system for perpetual systems," in *SenSys '07*, 2007.

[16] K. Lin, J. Yu, J. Hsu, S. Zahedi, D. Lee, J. Friedman, A. Kansal, V. Raghunathan, and M. Srivastava, "Heliomote: enabling long-lived sensor networks through solar energy harvesting," in *SenSys '05*, 2005.

[17] K. Seada, M. Zuniga, A. Helmy, and B. Krishnamachari, "Energy efficient forwarding strategies for geographic routing," in *SenSys '04*, 2004.

[18] S. Kang, J. Lee, H. Jang, H. Lee, Y. Lee, S. Park, T. Park, and J. Song, "Seemon: scalable and energy-efficient context monitoring framework for sensor-rich mobile environments," in *MobiSys '08*, 2008.

[19] MAXWELL Technologies, available at http://www.maxwell.com/ultracapacitors/products/large-cell/bcap3000.asp.

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

[21] R. Signorelli, J. Schindall, and J. Kassakian, "Carbon Nanotube Enhanced Double Layer Capacitor," in *15th International Seminar on Double Layer Capacitors and Hybrid Energy Storage Devices*, 2005.

[22] R. Szewczyk, A. Mainwaring, J. Anderson, and D. Culler, "An analysis of a large scale habit monitoring application," in *SenSys '04*, 2004.

[23] R.Rao, S.Vrudhula, D.N.Rakhmatov., "Battery modeling for energy-aware system design," in *Computer, 36(12):77-87*, 2003.

[24] S. Lin, G. Zhou, L. Gu, T. He, and J. A. Stankovic, "ATPC: Adaptive transmission power control for wireless sensor networks," in *SenSys '06*, 2006.

[25] S. Madden, M. Franklin, J. Hellerstein, and W. Hong, "TAG: A tiny aggregation service for Ad-Hoc sensor networks," in *OSDI '02*, 2002.

[26] S. Meninger, J.O. Mur-Miranda, R. Amirtharajah, A. Chandrakasan, and J. Lang, "Vibration-to-electric energy conversion," in *ISLPED '99*, 1999.

[27] C. M. Sadler and M. Martonosi, "Dali: a communication-centric data abstraction layer for energy-constrained devices in mobile sensor networks," in *MobiSys '07*, 2007.

[28] Suzanne Foster, "The energy efficiency of common household battery charging systems: Results and implications," in *Natural Resources Defense Council*, http://www.efficientproducts.org/reports/bchargers/NRDC-Ecos_Battery_Charger_Efficiency.pdf.

[29] Weir, et al, "Electrical-energy-storage unit (EESU) utilizing ceramic and integrated-circuit technologies for replacement of electrochemical batteries," in *United States Patent 7033406*, 2006.

[30] X. Jiang, J. Polastre, D. Culler, "Perpetual environmentally powered sensor networks," in *IPSN '05*, 2005.

[31] X. Jiang, P. Dutta, D. Culler, and I. Stoica, "Micro power meter for energy monitoring of wireless sensor networks at scale," in *IPSN '07*, 2007.

[32] G. Zhou, T. He, S. Krishnamurthy, and J. A. Stankovic, "Impact of radio irregularity on wireless sensor networks," in *MobiSys '04*, 2004.