eShare: A Capacitor-Driven Energy Storage and Sharing Network for Long-Term Operation

Ting Zhu[*], Yu Gu[†], Tian He, and Zhi-Li Zhang
Department of Computer Science and Engineering, University of Minnesota, Twin Cities
{tzhu, yugu, tianhe, zhzhang}@cs.umn.edu

Abstract

The ability to move energy around makes it feasible to build distributed energy storage systems that can robustly extend the lifetime of networked sensor systems. *eShare* supports the concept of *energy sharing* among multiple embedded sensor devices by providing designs for *energy routers* (i.e., energy storage and routing devices) and related energy access and network protocols. In a nutshell, energy routers exchange energy sharing control information using their data network while sharing energy freely among connected embedded sensor devices using their energy network. To improve sharing efficiency subject to energy leakage, we develop an effective energy charging and discharging mechanism using an array of ultra-capacitors as the main component of an energy router. We extensively evaluate our system under six real-world settings. Results indicate our charging and discharging control can effectively minimize the energy leaked away. Moreover, the energy sharing protocol can quantitatively share 113J energy with 96.82% accuracy in less than 2 seconds.

Categories and Subject Descriptors

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

General Terms

Measurement, Design, Performance, Experimentation

Keywords

Energy, Capacitor, Networks, Green, Embedded System

1 Introduction

With increasing network connectivity available, data generated by embedded sensor devices have been effectively shared in many collaborative applications, such as sensor data collection [19, 30], environmental monitoring [3, 1], and wearable computing [21]. *Data sharing* has been a research topic for many years, with hundreds of related projects. In contrast, the concept of *energy sharing* among multiple embedded sensor devices has captured very little attention. Until now, energy has been predominately harvested and consumed locally in a single embedded sensor device [11, 23, 42].

In this work, we attempt to lay a foundation for energy sharing by providing a hardware design for *energy routers* (i.e., energy storage and routing devices) and related energy access and networking protocols. Inspired by the data network architecture, our objective is to route energy efficiently and quantitatively among embedded sensor devices. With energy sharing, we can (i) build distributed energy storage devices to power the embedded sensor devices, (ii) improve the efficiency of energy storage devices, which are subject to leakage, and (iii) balance energy usage to avoid the early depletion of individual energy storage devices. Because wireless energy transfer [12] is not yet a mature technology, we do not expect that our work can be applied in environments where wiring is prohibitive or inconvenient. However, we argue that this energy sharing technique can be beneficial for a handful of applications, such as wearable computing [21], green building [22], and infrastructure monitoring [3], where power wiring is feasible.

Energy sharing efficiency is the key research issue we address in this work. It is known that sophisticated recharging circuits and electro-chemical conversion in battery-based systems can reduce energy efficiency to as low as 6% [37]. To avoid such a limitation, we design and implement energy routers using ultra-capacitors alone, which have several advantages over batteries for energy sharing purposes. For instance, they (i) have high charge efficiency (i.e., more than 90%); (ii) have more than 1 million recharge cycles, which translates to a lifetime of more than 10 years; and (iii) can be charged very quickly. Furthermore, recent advances in ultra-capacitor technology make it possible to use ultra-capacitors as the *only* energy storage unit. For instance, research groups at MIT [35] and the University of Maryland [33] have announced nanotube-based ultra-capacitors, which can provide energy storage densities comparable to those of batteries. In 2006, a U.S. patent [44] 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

[*]Ting Zhu is now a CI Fellow at UMass Amherst
[†]Yu Gu is now an assistant professor at SUTD

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. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is premitted. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
SenSys'10, November 3–5, 2010, Zurich, Switzerland.
Copyright 2010 ACM 978-1-4503-0344-6 ...$10.00

of Li-ion batteries. The largest capacitance available on the market increased rapidly from 3,000F in 2008 to 5,000F [32] in 2009. Powered by a 5,000F capacitor, a MICAz mote can work for more than 138 days under a 5% duty cycle with a single initial charge of the capacitor.

On the other hand, it is a challenging task to build efficient ultra-capacitor-based energy storage and routing devices (i.e., energy routers) because the leakage power grows rapidly with the physical size and the remaining energy residing within a single large ultra-capacitor. An efficient approach is to build an energy storage and routing device using an ultra-capacitor array, which calls for advanced designs for controlling energy charging and discharging among multiple ultra-capacitors.

It should be noted that energy sharing requires the collaboration of a data network with an energy network that connects all the energy storage units of the embedded sensor devices, as energy routers need to exchange control information using a data network when transferring energy back and forth using an energy network. Such a combination makes our work on energy sharing unique and new. More specifically, our major contributions are as follows:

• We have designed and implemented the first ultra-capacitor-based energy router for sharing energy among embedded sensor devices. Our prototype can quantitatively control the amount of energy transferred from one device to another device.

• To the best of our knowledge, this is the first in-depth work to investigate effective energy charging and discharging procedures using an array of capacitors to minimize leakage and maximize sharable energy. Our hardware and control layer design is simple and general and can be easily extended to support n capacitors.

• This work is the first to design an *energy access protocol* to share energy among neighboring devices and an *energy network protocol* to optimally distribute energy among networked devices. These designs can efficiently utilize the limited energy and improve the performance of multiple energy storages as a whole.

• We evaluated our system extensively in multiple real-world testbeds and simulations. The results indicate that our system can efficiently share energy among multiple embedded devices and minimize energy leakage.

The paper is organized as follows: Section 2 discusses the need for energy sharing and its challenging issues. Section 3 gives an overview of the system architecture. Hardware, control, and energy sharing layers are presented in Sections 4, 5, and 6, respectively. System implementation and evaluation are detailed in Sections 7 and 8. Related work is discussed in Section 9. Finally, Section 10 concludes the paper.

2 Motivation

Energy sharing is a new style of energy management that allows energy to *efficiently* and *quantitatively* flow back and forth among multiple energy storage systems. It can be applied in many applications, including wearable computing [21], electrical vehicles [20], smart microgrids [47], and infrastructure monitoring [3]. To generalize major application requirements and illustrate the benefit of energy sharing features, we study two representative scenarios, as follows:

Scenario 1: Greenhouse Applications

ClimateMinder's GrowFlex technology [9] comprises a battery or solar-powered sensor network that enables farmers to optimize environmental conditions in their greenhouses. To measure the environmental parameters (e.g., soil moisture, leaf wetness, and ambient temperature) and to control the irrigation and vents, embedded sensor devices are deployed at different altitudes, which provide different energy harvesting power. Current solutions use back-up batteries [9] to provide a lifetime of 6-8 months.

In this scenario, if we connect the energy storage of the embedded sensor devices with each other and enable energy sharing among these devices, the energy harvested by the devices that are directly exposed to sunlight can be utilized to power the devices that are in shadow. In this way, the lifetime of the whole network can be prolonged. Given the dynamic changes in the environmental energy, each energy sharing operation among these devices should be finished within a short duration of time. We note that the greenhouse infrastructure provides a very good platform to connect the devices by using wires. Moreover, distributed energy sharing protocols are needed in large-scale greenhouse applications.

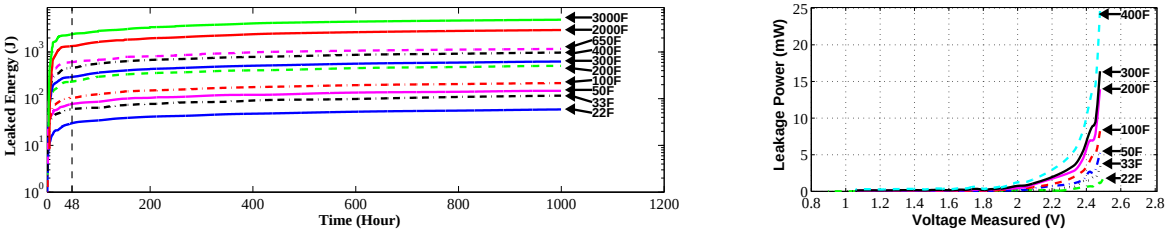
Scenario 2: Wearable Computing Applications

In wearable computing applications, Dr. Abowd's team has empirically studied the expected harvesting power from six body locations. Their results [21] indicate that devices at different locations have significantly different energy harvesting power (e.g., $115 \pm 106\mu W$ *on the wrist and* $1.01 \pm 0.46mW$ *on the arm). Moreover, the harvesting power varies over time.*

In this scenario, there is temporal and spatial diversity in the harvesting power of each device. If we enable energy sharing among these devices, the energy generated from the device on the arm can be utilized to power the device on the wrist. Moreover, the temporal variations in the harvesting power also requires the device on the wrist to share energy with the device on the arm during the period when only the device on the wrist can harvest energy. Given the low harvesting power of individual devices, highly efficient and quantitatively controllable energy sharing is needed to maintain the aliveness of all the devices. Moreover, these devices can be connected using wires integrated in clothing. In such a design, distributed energy storages are desired to ensure that devices can use their local energy storage units to power themselves after wiring failures.

2.1 Capacitors vs. Battery-Driven Storage

The two scenarios discussed above illustrate the benefit of energy sharing features and also indicate the requirements of energy sharing: it must be (i) fast, (ii) highly efficient, and (iii) quantitatively controllable. Theoretically, the energy sharing concept can be applied to battery-based systems, but the performance of these systems is significantly affected by the limitations of batteries: (i) low charge efficiency (as low as 6% [37]) due to electro-chemical conversion, (ii) limited charge current, and (iii) inaccurate remaining energy prediction, which results in inaccurate control of energy sharing. In contrast, capacitors do not have these limitations.



(a) Leaked Energy vs. Time (b) Leakage Power vs. Voltages

Figure 1. Leakage Property of Ultra-capacitor

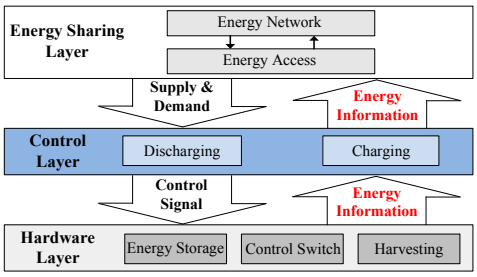


Figure 2. Overview of System Architecture

However, ultra-capacitors also have a major limitation, namely, leakage, which introduces a design challenge for a capacitor-driven energy storage system. To investigate the leakage property of the ultra-capacitor, we conducted experiments over a period of 6 weeks using ten types of ultra-capacitors, ranging from 22F to 3,000F. After an ultra-capacitor was fully charged, we isolated it and continuously monitored its remaining voltage over 1,000 hours.

Figure 1(a) reveals that the ultra-capacitors have a very large amount of energy (e.g., 22.9% of total energy for 3,000F) leaked away during the first 48 hours, but little energy (e.g., 21.98% for 3,000F) leaked away during the following 952 hours (i.e., 39.7 days). At the end of the 1,000 hours, the 3,000F capacitor still has 2.003V of remaining voltage. These results indicate that ultra-capacitors have a leakage performance that is comparable to that of batteries (e.g., 15% per month for an NiMH battery [38]) when voltage is controlled under an appropriate level. However, ultra-capacitors suffer severe energy leakage when they are charged to the limit. We also investigated the correlation between the voltage and leakage of ultra-capacitors. Figure 1(b) 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 400F ultra-capacitor increases to 2.5V (shown in Figure 1(b)), its leakage power is 24.65 mW, which is equivalent to the power needed for powering a MICAz mote to work under more than a 30% duty cycle.

3 System Overview

The eShare system uses energy routers to share energy among multiple networked embedded sensor devices. Since the main storage component of an energy router is an array of ultra-capacitors, we need to address the energy leakage issue as revealed by the empirical study in Section 2. To provide energy sharing with leakage control, we propose a three-layer design, as shown in Figure 2.

• The hardware layer (Section 4) uses (i) multiple ultra-capacitors to form a capacitor array as the *only* energy storage unit to power the system and (ii) a series of control switches to control different types of connections between the ultra-capacitors (i.e., in parallel, series, or hybrid). To achieve fast and efficient energy sharing, we use N-MOSFETs as switches, which support a large amount of current, low energy consumption, and extremely short switching time (at the level of *ns*). This layer also provides an energy harvesting circuit (e.g., solar panel or wind generator) to harvest the environmental energy. Moreover, this layer (i) monitors the remaining energy inside ultra-capacitors and (ii) samples the harvesting power from the energy harvesting circuit. These two pieces of information are then fed into the control layer.

• The control layer (Section 5) serves as a bridge between the hardware layer and energy sharing layer. It plays two important roles: (i) it calculates the energy leakage rate online based on the remaining energy inside each ultra-capacitor and then forwards the leakage information (together with the remaining and harvesting energy) to the energy sharing layer; and (ii) based on the leakage model and the energy supply and demand request from the energy sharing layer, it controls the working state (i.e., discharging or charging) of the energy storage system by sending the control signal to the hardware layer to control the on/off state of the control switches. Since the control layer has all of the energy information (including leakage information), it can conduct fine-grained control (including energy sharing).

• The energy sharing layer (Section 6) consists of an *energy access protocol* and an *energy network protocol*. Similar to the role of data-link and network protocols in a data network, an energy access protocol controls energy exchange between a pair of neighboring nodes and an energy network protocol decides the most efficient routes for energy distribution, respectively. By using the energy information forwarded from the control layer, the energy sharing layer calculates the energy gap between the actually available energy and the expected energy requirement. To close this gap, the energy sharing layer sends quantitative energy supply and demand requests via the data network and then exchanges energy via an energy network.

4 Hardware Layer

The design goal of the hardware layer is to build a *distributed energy storage* system with minimum leakage. Although a centralized energy storage system can be used to power all nodes if they are connected together, it has a few

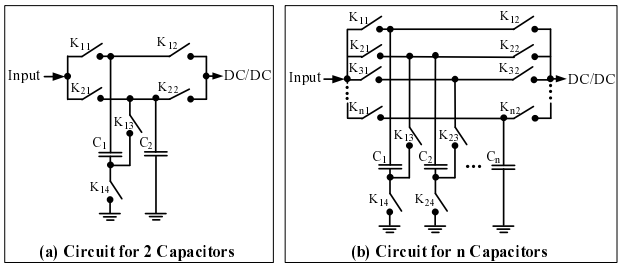


Figure 3. Capacitor Array Circuit

Connection Types	K_{11}	K_{12}	K_{13}	K_{14}	K_{21}	K_{22}
Discharge C_1, Disconnect C_2	0	1	0	1	0	0
Discharge C_1, Charge C_2	0	1	0	1	1	0
Charge & Discharge C_1 simultaneously	1	1	0	1	0	0
Discharge C_2, Disconnect C_1	0	0	0	1	0	1
Discharge C_2, Charge C_1	1	0	0	1	0	1
Charge & Discharge C_2 simultaneously	0	0	0	1	1	1
Discharge C_1, C_2 in parallel	0	1	0	1	0	1
Charge & Discharge C_1, C_2 in parallel	1	1	0	1	1	1
Discharge C_1, C_2 in series	0	1	1	0	0	0
Charge & Discharge C_1, C_2 in series	1	1	1	0	1	0

Table 1. Different Types of Connections

limitations. For instance, in WSNs, energy is normally harvested and consumed in a distributed manner. Due to the energy loss in wires, it would be wasteful to store energy in a central storage and then redistribute the energy to individual devices. Moreover, a distributed energy storage is more robust than a centralized one when the system encounters failures such as wire failure.

4.1 Energy Storage Design: Capacitor Array

A straightforward choice for energy storage is to use a single large ultra-capacitor. However, this suffers from a slow boot-up time (to reach operational voltage), high remaining energy (detailed in Section 5.3), and inflexibility in fine-grained control (due to the limited granularity of the A/D converter inside sensor devices). For example, MICAz uses a 10-bit A/D converter. When being charged or discharged with the same amount of energy, a single large capacitor's voltage value does not change as significantly as a small capacitor, which causes inaccurate charging and discharging control (detailed in Sections 5.2 and 5.3). In our design, we build the energy router using an array of ultra-capacitors that have different levels of capacitances. This capacitor array is designed to satisfy the following three requirements:

• ***Generality:*** The circuit should be easily extended to n capacitors. Moreover, the capacitors can be connected in different types of connections (e.g., series, parallel, or hybrid) by using this circuit.

• ***Simplicity:*** It should expose a minimum number of interfaces for control purposes.

• ***Stability:*** It needs to have a stable and simultaneous charging and discharging capability. In other words, during the charging state, the circuit still needs to output energy to power the embedded sensor device. Moreover, it is important to stably and smoothly change the connection type (e.g., from parallel to series) of the capacitors during the charging/discharging state.

The circuit we designed successfully achieves the above three design goals. Figure 3(a) shows the circuit for two

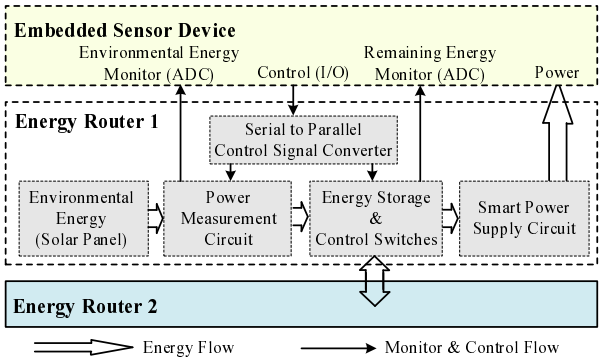


Figure 4. Energy Router Hardware Architecture

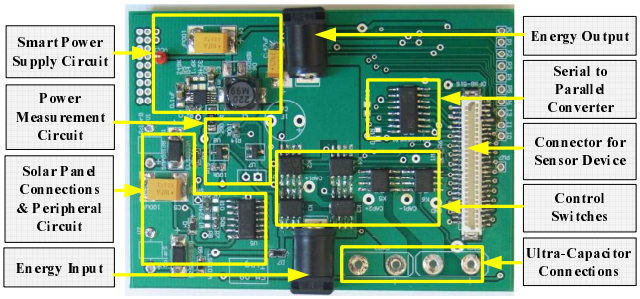


Figure 5. Energy Router Platform

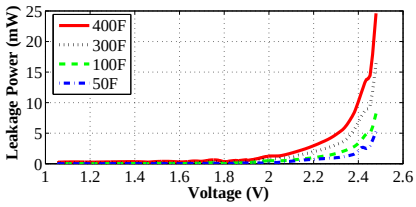
capacitors, which can be easily extended to support n capacitors (shown in Figure 3(b)). Moreover, this circuit can achieve different combinations of connections (shown in Table 1). Here we use 0/1 to indicate the on/off state of the switches. All of these switches can be controlled by an embedded sensor device through the serial to the parallel control signal converter (shown in Figure 4), which introduces a minimum number of control interfaces to the sensor device. The output of the circuit is connected to a step-up DC/DC converter. Since a DC/DC converter normally has a very wide input voltage range (e.g., $0.7V \sim 5V$ for MAX1676), our circuit can still provide a constant voltage from the output of the DC/DC converter when the capacitors are changing their connections (e.g., from parallel to series).

4.2 Hardware Platform

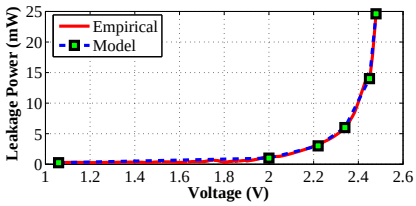
The energy router is an add-on power board that supports multiple ultra-capacitors to form an array as the only energy storage unit so as to overcome the intrinsic limitations of batteries (e.g., energy uncertainty, limited recharge cycles, low conversion efficiency, and environment unfriendliness). The architecture of the energy router is shown in Figure 4. It consists of (i) solar panels and a peripheral circuit for energy harvesting; (ii) a harvesting power measurement circuit; (iii) an ultra-capacitor-based energy storage and control switches, which can directly share energy with the other energy storage and supply devices; (iv) a serial to parallel control signal converter; and (v) a smart power supply circuit with a DC/DC converter for powering the embedded sensor device that is attached to the energy router board. The corresponding printed circuit board is shown in Figure 5.

5 Control Layer

By interacting with the hardware layer, the control layer is responsible for two basic functions: energy charging and



(a) Empirical Leakage Pattern



(b) Leakage Model Fitting for 400F

Figure 6. Leakage Model

energy discharging, which are the building blocks for energy sharing within a network. To improve the charge and discharging efficiency, we need to balance the energy stored within an array of capacitors according to their energy leakage model so that total leakage can be minimized during the charge, discharge, and storage stages.

5.1 Energy Leakage Model

Based on our empirical study in Section 2.1, we formulate a leakage model to characterize the relationship between the voltage in the ultra-capacitor and the leakage power. The embedded sensor device builds the leakage model by storing the value of the leakage power P_L and the corresponding capacitor's voltage value V_c in its memory. Figure 6(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 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.(1).

$$P_L(V_c) = \begin{cases} a_1 \cdot V_c + b_1; & V_1 \le V_c < V_2 \\ a_2 \cdot V_c + b_2; & V_2 \le V_c < V_3 \\ \vdots \quad ; & \vdots \\ a_n \cdot V_c + b_n; & V_n \le V_c \le V_{n+1} \end{cases} \quad (1)$$

Here $V_1, V_2, \cdots, V_{n+1}$ 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 6(b) shows an example for the line-segment-based modeling of the leakage power for a 400F capacitor. The square dots represent the turning points. The model we built accurately matches the empirical results.

5.2 Charging

As shown in Section 5.1, energy leakage is mainly affected by the remaining energy within an ultra-capacitor and its leakage model. Energy storage efficiency is reduced if we charge an array of capacitors in a sequential manner (i.e., one by one), because energy leakage increases rapidly when

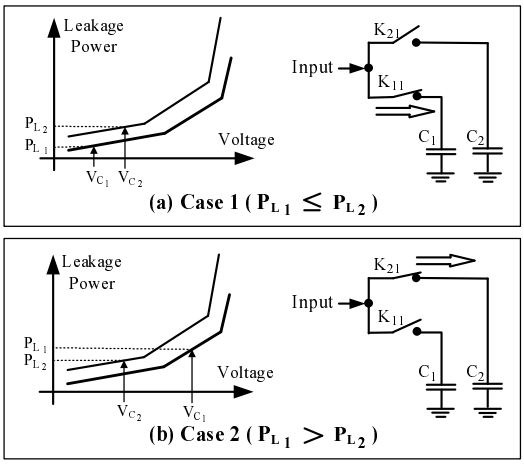


Figure 7. Charging Control Example

a capacitor approaches its capacity. The design goal of the charging control is to balance the energy stored in an array of capacitors so that energy can be stored for a long period with minimal energy leakage.

Basic Alternative Charging Control: As an example, Figure 7 shows the charging control of two capacitors (C_1 and C_2), which belong to a single capacitor array. Initially, switches K_{11} and K_{21} are turned off. When the charging control is conducted, the control layer measures the voltage values (V_{C_1} and V_{C_2}) of these two capacitors and calculates the corresponding leakage powers (P_{L_1} and P_{L_2}) based on the leakage model. If P_{L_1} is less than or equal to P_{L_2} (shown in Figure 7(a)), the control layer decides to charge C_1 by turning on K_{11}. In the case that P_{L_1} is greater than P_{L_2} (shown in Figure 7(b)), C_2 is charged. The charging control transfers energy between C_1 and C_2 alternatively with time interval T until there is no external energy available or all the capacitors are fully charged.

Adaptive Charging Control: As an all-purpose energy storage system, it is important that the system can be charged by different energy sources (e.g., environmental energy or DC power supply). However, these diverse energy sources introduce a challenge to the design of the charging control because their charge current varies over orders of magnitude (ranging from less than $1mA$ for environmental energy to more than $10A$ for DC power supply). If the charging control is conducted with a fixed short time interval (in favor of the large charge current), unnecessary voltage measurements and comparisons are conducted when the capacitor is charged with a small current and its voltage increases slowly, whereas if the time interval is large (in favor of the small charge current), more energy is leaked away when the charge current is large.

To minimize the energy leaked away and reduce the control overhead, we propose an adaptive charging control, which conducts the charging control operation with an adaptive time interval based on the charge current. Algorithm 1 illustrates the control algorithm for n capacitors. Initially, the controller turns off all the switches and calculates the leakage power of all the capacitors based on the measured voltage values of these capacitors (Line 1 to 3). Then the controller turns on the switch that connects to the capacitor

Algorithm 1: Adaptive Charging Control

```
1  for i ← 1 to n do
2      K_i1 ← 0 ;
3      Calculate P_Li based on measured V_ci ;
4  m ← GetMinCapId(), K_m1 ← 1 ;
5  Set timer with interval T ;
6  while timer fired do
7      for i ← 1 to n do
8          Measure V_ci and update P_Li ;
9      if m = GetMinCapId() then
10         T ← 2T ;
11     else
12         K_m1 ← 0, m ← GetMinCapId();
13         K_m1 ← 1 ;
14     if Capacitors Full/No External Energy then
15         Set timer with interval T ;
```

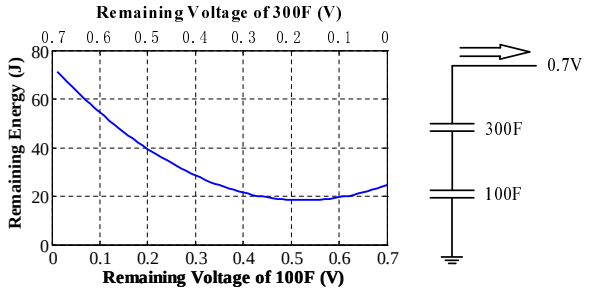


Figure 8. Different Voltage Combinations

with the smallest leakage power and sets the timer with the smallest time interval T (Line 4 and 5). Since we allow simultaneous charging and discharging, when the timer fires, the controller needs to recalculate the leakage power of all the capacitors based on the newly measured voltage values of those capacitors (Lines 7 to 8). If the capacitor with the smallest leakage power is the same as the one in the previous cycle (indicating the current time interval T is too short), the time interval is doubled (Line 9 and 10). Otherwise, the controller turns off the switch that connects to the previous capacitor and turns on the switch that connects to the capacitor that currently has the smallest leakage power (Line 11 to 13). The above control procedure repeats every time interval T until all of the capacitors are fully charged or there is no external energy available (Line 14 to 15).

5.3 Discharging

The ability to move energy around depends on how much energy can be discharged. For a given energy budget, the less energy inside the capacitor array at the end of discharging process, the more energy that can be shared. This section describes how to smartly discharge the capacitor array to share a maximum amount of energy.

5.3.1 Voltage Values for Minimal Remaining Energy

Since the output of the capacitor array serves as the input of the DC/DC converter that has the minimum input voltage (e.g., 0.7V for MAX1676), to minimize the remaining energy inside the capacitor array, the capacitors are connected in series so that the sum of their voltages is greater than the minimum input voltage of the DC/DC converter. However, for a fixed summation, different combinations of the capacitors' voltage values result in different remaining energy levels inside the capacitor array. As an example, Figure 8 shows that a 100F and a 300F capacitor are connected in a series to provide the same sum of 0.7V. However, the remaining energy is dramatically different for different combinations of the voltage values of the 100F and 300F capacitors. To minimize the remaining energy inside the series of capacitors, we need to determine the voltage value of each capacitor. The problem can be formulated as:

$$\begin{aligned} \text{minimize} \quad & V^T C V \\ \text{subject to} \quad & V \succeq 0 \\ & \sum_{i=1}^{n} V_{c_i} = V_{min} \end{aligned} \tag{2}$$

Here, we define $C = \frac{1}{2} diag(c_1, c_2, \cdots, c_n)$, and $V^T = [V_{c_1}, V_{c_2}, \cdots, V_{c_n}]$. V_{min} is the minimum input voltage of DC/DC converter. This is a quadratic programming problem and the analytic solution is as follows:

$$\begin{cases} V_{c_1} = V_{min} \cdot c_{total} \cdot \frac{1}{c_1}; \\ V_{c_2} = V_{min} \cdot c_{total} \cdot \frac{1}{c_2}; \\ \quad \vdots \quad ; \\ V_{c_n} = V_{min} \cdot c_{total} \cdot \frac{1}{c_n}; \end{cases} \tag{3}$$

Here c_{total} is the total capacitance of all the capacitors in series and $c_{total} = \frac{1}{\frac{1}{c_1} + \frac{1}{c_2} + \cdots + \frac{1}{c_n}}$.

5.3.2 Discharge Procedure

From the above, we get the voltage values of a series of capacitors with the minimum remaining energy. However, initially all of the capacitors have voltage values larger than the minimum input voltage of the DC/DC converter. In order to minimize the energy leaked away, only the capacitor with the highest leakage power is selected as the input of the DC/DC converter; all the other capacitors are originally disconnected from each other. To achieve the minimum remaining energy, we need to decide when to connect all of these capacitors in a series. If we connect them in a series too early (or late), their voltage values may be different, which causes the remaining energy not to be the minimum. By investigating Eqn. 3 carefully, one can find that the voltage values of each capacitor share a common factor $V_{min} \cdot c_{total}$. Moreover, these voltage values are reverse proportional to their capacitance. Our further experiments reveal that if the capacitors are discharged in a series with their initial voltage values reverse proportional to their capacitance, their voltage value will maintain this relation until the end of the discharge. Furthermore, the capacitor with the largest capacitance has the smallest voltage value. For example, in Figure 8, if the initial voltage values of the 100F and 300F capacitors are 2.1V and 0.7V, respectively, after they are discharged in series, their voltage value will maintain the relation in which the voltage value of 100F is 3 times more than the voltage value of 300F.

Guided by the above observation, we can calculate the minimum voltage values of these capacitors before they are connected in a series:

$$\begin{cases} V_{1_{min}} = V_{min} \cdot \frac{c_i}{c_1}; \\ V_{2_{min}} = V_{min} \cdot \frac{c_i}{c_2}; \\ \vdots \quad ; \\ V_{i_{min}} = V_{min} \quad ; \\ \vdots \quad ; \\ V_{n_{min}} = V_{min} \cdot \frac{c_i}{c_n}; \end{cases} \tag{4}$$

Here we assume that capacitor c_i has the largest capacitance, which results in the smallest voltage value. Since all of the parameters in Eqn. 4 are determined before conducting the discharging control, we can use an indexed lookup table, which is calculated offline, to reduce run-time computational overhead of Eqn. 4. In summary, the discharging control algorithm is straightforward and can be described as follows: Initially, all of the capacitors are adaptively discharged based on which capacitor has the highest leakage power. If the voltage value of the capacitor reaches the minimum voltage value calculated from Eqn. 4, this capacitor is excluded from the adaptive discharging. When all of the capacitors have voltage values less than or equal to the voltage in Eqn. 4, these capacitors are connected in a series and conduct discharging until their total voltage value is less than the minimum input voltage of the DC/DC converter.

We note that there is a trade-off between minimizing the leaked energy and minimizing the remaining energy. Here minimizing the remaining energy has higher priority than minimizing the leaked energy. This is because when the capacitors start to be connected in a series, their leakage power is very low. We only optimize the energy utilization efficiency on the capacitor side without considering the DC/DC converter's efficiency. The reason is that the change of a DC/DC converter's efficiency is very small and can be ignored when the input voltage value changes. For example, when the input voltage changes from 1.2V to 2.4V, MAX1676 (the DC/DC converter used in our experiment) only increases its efficiency by 2%, which is negligible.

6 Energy Access and Network Protocol

Similar to data-link and network protocols used for data sharing, energy sharing shall be controlled by energy access and network protocols. The former controls energy exchange between a pair of neighboring devices, and the latter decides the most efficient routes for energy distribution.

6.1 Energy Access Protocol

Without losing generality, we assume that supplier *i* wants to share some energy with receiver *j* (shown in Figure 9). To achieve the design goal of being fast and highly efficient, instead of going through the DC/DC converter, the energy is directly shared between the capacitor arrays inside supplier *i* and receiver *j*. This is because the DC/DC converter's output current is limited, which significantly increases the time duration of the energy sharing. One may think that lower current will result in lower energy loss. However, our empirical study reveals that the DC/DC converter consumes very large amount of energy, especially when its input voltage is low. Therefore, using a DC/DC converter is less energy efficient than directly connecting two capacitor arrays. As shown in Figure 9, the output of *i* is directly connected to the input of

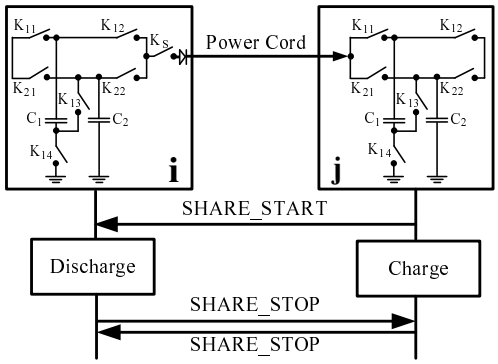


Figure 9. Energy Access Protocol

j through power cords. For high efficiency purposes, we use power cords instead of a wireless charging interface.

The energy access protocol is a receiver-initiated protocol. After receiving the energy sharing control from the network layer, the receiver *j* sends out the *SHARE_START* packet to the supplier *i* and prepares for charging. Upon receiving the *SHARE_START* packet, *i* turns on the switch K_S and conducts the discharging control. During the discharging control, the supplier *i* periodically monitors the amount of energy transferred by sampling the voltage of its capacitors and calculating its own remaining energy until the expected amount of energy is transferred. Then the supplier turns off the switch K_S and sends out the *SHARE_STOP* packet to receiver *j* informing *j* of the end of energy sharing.

On the receiver *j* side, the charging control is conducted. *j* also periodically monitors the energy received by sampling the voltage of its capacitors and calculating its remaining energy. When the expected energy is received, *j* turns off the charging switches (i.e., K_{11} and K_{21} in Figure 9) and sends out the *SHARE_STOP* packet to *i* telling *i* that the energy sharing is terminated. Since *j* can accurately monitor the increased remaining energy inside its capacitor, it can terminate the energy sharing process with high accuracy. In our energy access protocol, both the supplier and the receiver can decide when to terminate the energy sharing process. Therefore, it is robust to the failure of supplier or receiver.

6.2 Energy Network Protocol

In this section, we introduce our energy network protocol, which is aimed at efficiently utilizing the energy inside the whole network and therefore prolonging the network lifetime. Here we define the network lifetime as the period of time during which all the embedded sensor devices inside the network have sufficient energy to execute assigned tasks.

6.2.1 Finding the Minimum Energy Loss Path

In order to share energy within the whole network, essentially we need to efficiently *route* the energy stored in one device to another device through one or multiple hops. However, when sharing energy between a pair of devices, a certain amount of energy will be lost due to factors such as energy loss within the power cord, diodes, and switches. Therefore, in order to achieve the goal of efficiently sharing energy within the whole network, similar to the data packet routing in data communication networks, we need to find the minimum energy loss path between any pair of sensor devices in the network. Specifically, given a set of energy harvesting

devices and the interconnection of power cords among those energy harvesting devices, we can derive a graph that is similar to traditional communication network graphs. In such a graph, the vertices are the energy harvesting devices and the edges are the power cords. The weight of each edge is the efficiency of energy transfer between the pair of sensor devices that is incident to this edge. Figure 10 shows an example of such an energy sharing network. In Figure 10, the value on each edge denotes the energy transfer efficiency along this edge. For example, if we assume the energy transfer efficiency (e_{ab}) between device a and b is 70% and the total amount of energy transferred between device a and device b is $100J$, then the energy loss during the sharing procedure is $100 * (1 - 0.7) = 30J$.

By applying any distributed shortest path algorithms [8] on such an energy sharing network, we can easily obtain the minimum energy loss path between each device and any other devices in the network. Specifically, each device i in the network maintains a metric called *energy sharing efficiency* for any other device j in the network (ESE_{ij}). In Figure 10, we also show the energy sharing efficiency metric values for all devices towards device a. For example, the minimum energy sharing efficiency between device a and device d is $e_{ac} \times e_{cd} = 0.81$ by going through path $a \rightarrow c \rightarrow d$.

6.2.2 Energy Optimal Sharing Among Devices

In an energy harvesting network, due to unbalanced energy harvesting rates [42] and varying workloads at individual devices, it is essential to share the energy within the whole network so as to improve network efficiency and to reduce unnecessary energy wastage due to energy leakage. In this section, we introduce optimal solutions for routing energy within the network with minimum energy loss. In the following sections, we discuss scenarios for accumulating and distributing energy at a device, respectively.

Accumulating Energy at a Device: If a specific device i in the network experiences a low energy harvesting rate or high workload, it may lack energy to complete all of its tasks. Under such conditions, this device would request that other devices share their energy with it and accumulate enough energy to support its operations. The main challenge here, therefore, is to ensure minimum energy loss due to the energy transfer for energy accumulation at this device i.

In a loss-prone energy sharing network, the less energy transferred in the network, the less energy loss would be incurred during the energy transfer process. Consequently, depending on the current energy harvesting and consumption rate, device i first calculates the minimum amount of energy that has to be accumulated from other devices. Then, based on the knowledge of the energy sharing efficiency from all other devices to device i, device i would sequentially request energy sharing from devices with the maximal energy sharing efficiencies until it has accumulated enough energy. The detailed procedure at the energy requester device i is shown in Protocol 2. When device j receives an energy sharing request from device i with energy accumulation budget E, it calculates the amount of energy it can afford to share with device i (E_j), and then transfers $Min(E, \frac{E_j}{ESE_{ij}})$ energy through the minimum energy loss path to device i.

Protocol 2: Energy Accumulation Protocol

input : Energy Accumulation Budget E
input : Energy Sharing Efficiency (ESE) for all other devices in the network with n devices

1 Sort ESE in non-increasing order ;
2 $m \leftarrow 1$;
3 **while** $E > 0$ *OR* $m \leq n-1$ **do**
4 Send a data message to the device that is corresponding to sorted ESE_m, request energy sharing of E ;
5 Receive E_m from the requested device ;
6 $E \leftarrow E - E_m$;
7 $m \leftarrow m + 1$;

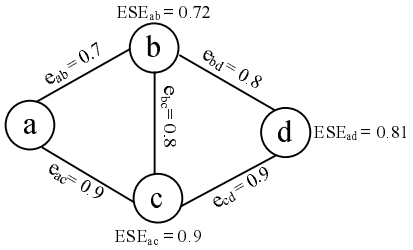
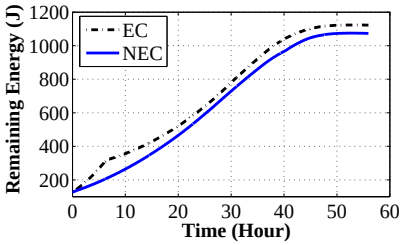


Figure 10. Energy Sharing Network

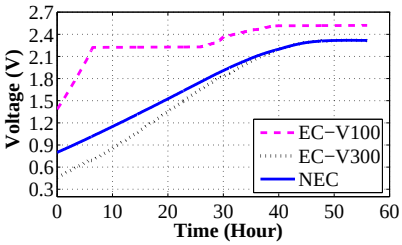
Proof of optimality: To prove the optimality of the above energy accumulation protocol, it is sufficient to show both the greedy choice and the optimal substructure properties. For the greedy choice property, since at each iteration we request energy from the device with the largest energy sharing efficiency to device i, we minimize the energy loss of energy transfer for the remaining energy needed at device i. This property can be proved by simple contradiction. Assume the remaining needed energy can be optimally transferred from n devices, then the total energy loss $(1-ESE_{1i}) \cdot E_1 + (1-ESE_{2i}) \cdot E_2 + \cdots + (1-ESE_{ni}) \cdot E_n$ is minimal, where E_j is the energy transferred from device j and $j = 1, 2, \cdots, n$. If there exists a $ESE_{n+1,i}$ which is larger than ESE_{ki}, $k = 1, 2, \cdots, n$, then accumulating energy from device k can result in more energy loss than accumulating energy from device $n+1$ first and then from device k. This can be formulated as $(1-ESE_{ki}) \cdot E_k > (1-ESE_{n+1,i}) \cdot E_{n+1} + (1-ESE_{ki}) \cdot (\frac{ESE_{ki} \cdot E_k - ESE_{n+1,i} \cdot E_{n+1}}{ESE_{ki}})$, which contradicts the optimality claim. Consequently, the greedy choice property holds. The optimal substructure property is straightforward since after each iteration, we reduce the problem to accumulating $E - E_k$ amount of energy, where E and E_k are the energy budget and the energy accumulated from device k at the previous iteration, respectively. By combining the above two properties, we prove that our proposed energy accumulation protocol is optimal in terms of minimum energy loss during the energy sharing process.

Case Study: To further illustrate the above energy accumulation process, we provide a simple walkthrough for the energy sharing network shown in Figure 10. Assume device a needs to accumulate $100J$ from the network. First it will request energy sharing with device c, which has the best energy sharing efficiency of 90% to device a. If device c decides it can share $80J$ of energy with device a, then device a would receive $80 * 0.9 = 72J$ from device c. Since $72J < 100J$, device a would request that device d share $28J$ energy with it.

(a) Indoor Harvesting Experiment



(b) Remaining Energy vs. Time



(c) Capacitor Voltage vs. Time

Figure 11. Indoor Energy Harvesting

Assuming device d can share up to $50J$ energy with device a, it then will transfer $\frac{28}{0.81} = 34.6J$ energy to device a, so as to meet the energy accumulation budget at device a by considering energy loss during the energy sharing process.

Distributing Energy at a Device: If a device i can harvest the excess amount of energy from the surrounding environment and has a low workload, then it would accumulate a large amount of energy at its ultra-capacitors, which would consequently lead to a large energy leakage. In order to conserve such otherwise leaked energy, device i should try to distribute its excessive energy throughout the network.

Specifically, depending on the current energy harvesting rate, energy consumption rate, and energy level at the local capacitor array, device i decides the amount of energy to distribute to other devices in the network. Then, similar to the energy accumulation process introduced in the previous section, device i attempts to distribute its energy to the devices with the best energy sharing efficiencies one by one until it has successfully distributed all its extra energy. Again, this greedy energy distribution process is optimal in terms of energy loss during energy sharing. The proof is also similar to that for energy accumulation in the previous section.

In summary, on top of the pairwise energy sharing primitive that is provided by the control layer, at the network-wide energy sharing layer we provide optimal solutions for controlling energy flow within the network for both accumulating and distributing energy at individual devices. Such network-wide energy sharing enables dynamic, efficient distribution of energy for improved system performance and energy efficiency in energy-dynamic energy harvesting systems with varying workloads. Our energy sharing design is compatible with all the other energy and activity prediction algorithms in the literature. For sensor network applications, sensor devices normally work under certain duty cycles with a specific time period. Therefore, the activities of sensor devices can be predicted based on these devices' previous activities. Since the environmental energy changes dynamically, instead of predicting the future available energy, the devices make energy sharing decisions based on their current remaining energy.

7 Evaluation of Effective Control

In this section, we evaluate the effectiveness of the control layer design under different types of real-world settings.

7.1 Evaluation Baselines and Metrics

In our experiment, we set the total capacitance to be 400F. To reduce the boot-up time and leakage, we use a combination of a 100F and a 300F capacitor, which are controlled by our efficient charging and discharging control algorithms. We call this approach the *efficient control* (*EC*), which is compared with the following baseline:

• **No Efficient Control (NEC):** This approach uses the same array of capacitors but provides no efficient control by simply connecting the capacitors in parallel to power the system.

The key advantage of our system design is that it efficiently minimizes the energy leaked away inside the energy storage system. Since it is hard to measure how much energy is leaked away, the metric used to evaluate the system is the *remaining energy*, the energy inside the capacitors that can be calculated based on the measured voltage values of the capacitors. With the same amount of energy charged both into the system and consumed, the larger the value of the remaining energy, the less the amount of energy leaked away.

7.2 Implementation

As shown in Figure 5 in Section 4, our hardware contains a custom circuit board that harvests energy to power the sensor device. As a proof of concept, we used a MICAz node as a sensor device that is powered by our energy storage system. We designed and implemented the control layer using TinyOS and NesC. The MICAz node normally works in sleeping mode, waking up only to conduct the charging and discharging control. Besides the control operations, it also measures the remaining energy inside the capacitor array and logs that data into the flash for the purpose of off-line analysis of the system's performance.

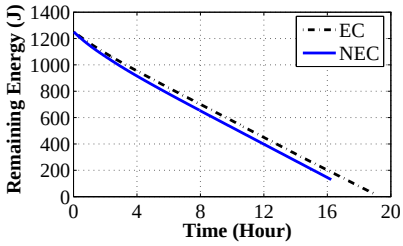
7.3 Different Experiment Scenarios

We carefully selected the experiment scenarios to represent a wide range of system working environments: (i) indoor energy harvesting, for investigating the charging control effectiveness; (ii) typical energy consumption pattern of an embedded mobile device, for studying the discharging control effectiveness; and (iii) outdoor energy harvesting, for exploring the hybrid charging and discharging control under a periodical and dynamic environment. For each scenario, we used two sets of hardware, one with efficient control (EC), the second one with no efficient control (NEC).

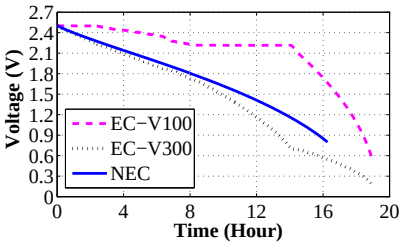
7.3.1 Indoor Energy Harvesting Experiment

The indoor energy harvesting experiment supports such potential applications as building automation and facility management. As shown in Figure 11(a), two sets of hardware were deployed near each other under a ceiling light so that they harvested a similar amount of environmental energy. The total time duration for this experiment was 56 hours. The light was turned on all that time.

(a) Mobile Phone Experiment



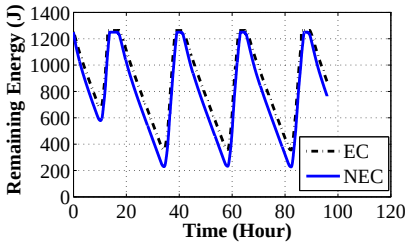
(b) Remaining Energy vs. Time



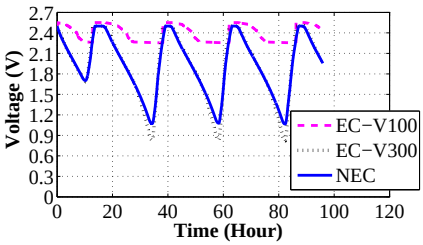
(c) Capacitor Voltage vs. Time

Figure 12. Mobile Phone Discharging

(a) Outdoor Harvesting Experiment



(b) Remaining Energy vs. Time



(c) Capacitor Voltage vs. Time

Figure 13. Outdoor Energy Harvesting

Figure 11(b) shows the remaining energy over time inside these two systems. With efficient control, the EC system always has greater remaining energy than the NEC system. At the end of the experiment, the remaining energy inside the EC system is 48.7J more than the remaining energy inside the NEC system. By showing the voltage values of the capacitors over time, Figure 11(c) reveals that the efficient charging control always selects the capacitor with the lowest leakage power to store the energy, which reduces the energy leaked away and results in the large voltage gap between the 100F capacitor and the 300F capacitor in the EC system.

7.3.2 Mobile Phone Discharging Experiment

The mobile phone discharging experiment supports such potential applications as content sharing [14], human-centric participatory sensing [28], and localization [2] using mobile phones. In this experiment, the mobile phone (Motorola V557) worked in the standby mode and was powered by one of these two systems at a time (shown in Figure 12(a)). By discharging the capacitor with the highest leakage power first (shown in Figure 12(c)), the EC system always maintains more remaining energy than the NEC system (shown in Figure 12(b)). Moreover, with effective switching from the parallel connection to the serial connection during the discharging, the EC system further prolongs the service time of the system. The total service time of the EC system is 19 hours, which is 17.3% more than the service time of the NEC system.

7.3.3 Outdoor Energy Harvesting Experiment

The outdoor energy harvesting experiment supports such potential applications as environmental monitoring [1, 3] and habitat monitoring [34]. In this experiment, we deployed our systems inside a weatherproof enclosure that was set facing south and at a 40° angle to the ground (shown in Figure 13(a)). The total time duration of the experiment was 96 hours. Figure 14 shows the energy harvested by these systems. We placed the solar panels of these systems close

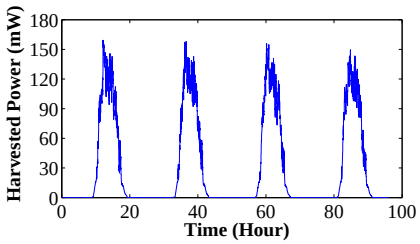


Figure 14. Harvested Power Over 4 Days

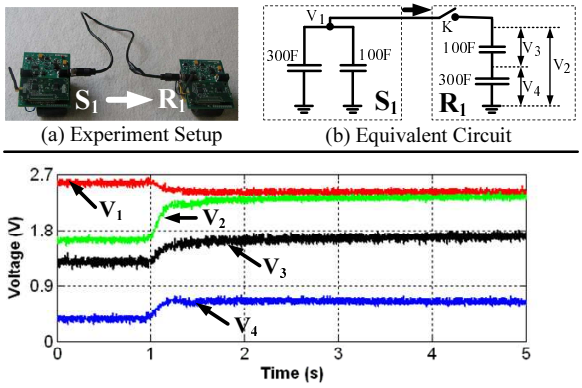
to each other to harvest similar amounts of energy from the environment. The MICAz nodes of these systems ran at a 10% duty cycle to consume the same amount of energy all the time. As shown in Figure 13(b), the EC system can efficiently maintain the remaining energy at a higher level than the NEC systems. At the end of the experiment, the remaining energy inside the EC system is 872.8J, which is 14.4% more than the remaining energy inside the NEC system. Figure 13(c) shows that our capacitor array circuit can smoothly conduct the charging and discharging controls.

7.3.4 Summary

We have evaluated our system under different energy patterns. Since the capacitor array in our evaluation contains only 2 capacitors, the gains shown in Figure 11, 12, and 13 are relatively modest. When the number of capacitors in a capacitor array increases, a better performance is expected. Moreover, the gains are not trivial for a small system such as a sensor device. For example, in Figure 11, the EC system has 48.7J more remaining energy than the NEC system. This amount of energy can power a MICAz mote to work under a 1% duty cycle for more than 16 hours.

8 Evaluation of Energy Sharing

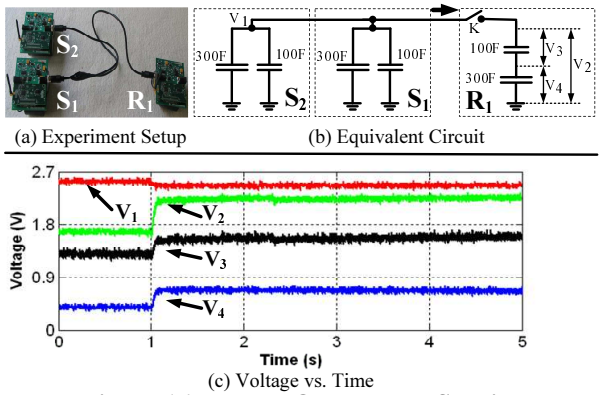
In Section 7, we have demonstrated the effectiveness of our hardware and control layer design. In this section, we evaluate the energy sharing layer design, which includes an energy access protocol and an energy network protocol.



(a) Experiment Setup (b) Equivalent Circuit

(c) Voltage vs. Time

Figure 15. One-to-One Energy Sharing



(a) Experiment Setup (b) Equivalent Circuit

(c) Voltage vs. Time

Figure 16. Two-to-One Energy Sharing

8.1 Evaluation of Energy Access Protocol

Our hardware design and the energy access protocol not only support energy sharing from one sensor device to another sensor device (one-to-one), but also support energy sharing from multiple devices to a single device (many-to-one). In this section, we evaluate and compare the energy sharing in these two scenarios.

8.1.1 One-to-One

As shown in Figure 15(a), the output of the supplier S_1 is connected to the input of the receptor R_1. The capacitors in S_1 are connected in parallel and their initial voltages are 2.5V each while the capacitors in R_1 are connected in series and their total voltage value is 1.6V (i.e., 0.4V for 300F and 1.2V for 100F). The equivalent circuit is shown in Figure 15(b). Figure 15(c) shows the voltage values of the capacitors during the energy sharing process. The energy sharing process starts at time 1 second and ends at time 3.1 seconds, for a total time duration of 2.1 seconds. At the end of the sharing process, the voltage value of the capacitors in S_1 is 2.37V each, and the voltage values of the 300F and 100F capacitors in R_1 are 0.64V and 1.71V, respectively. From the voltage values, we can calculate that the total energy lost in S_1 is 126.62J and the total energy received in R_1 is 113J, which can be used to power the MICAz mote to work in a 1% duty cycle for more than 38 hours.

8.1.2 Many-to-One

In this experiment, we evaluated the quantitative energy sharing among two suppliers (S_1 and S_2) and one receptor R_1 (shown in Figure 16(a)). For comparison purposes, the receptor R_1 was required to get the same amount of energy as that gained in the one-to-one experiment (i.e., 113J). As shown in Figure 16(b), the capacitors inside S_1 and S_2 were connected in parallel and the capacitors inside R_1 were connected in series. Initially the voltage values of the capacitors were pre-charged to the same values as those used in the one-to-one experiment (shown in Figure 16(c)).

R_1 started the energy sharing process at time 1 second and periodically sampled the total voltage value of the 100F and 300F capacitors with a time interval of 5ms until the voltage value reached the pre-defined value (2.35V in this case). Then R_1 turned off the switch K (shown in Figure 16(b)) and broadcasted the *SHARE_STOP* packet to S_1 and S_2 indicating the end of the energy sharing (i.e., at time 2.3 second in Figure 16(c)). Due to the granularity of the A/D converter, the actual total voltage value of 100F and 300F capacitors was slightly higher (i.e., 2.378V) than the pre-defined value (i.e., 2.35V) at the end of the experiment. However, our system still achieved very high accuracy. Compared with the pre-defined energy gain of 113J, the actual energy gain was 116.59J, which results in a transfer accuracy of 96.82%.

Performance	One-to-One	Two-to-One
Energy Supplied (J)	126.62	138.04
Energy Received (J)	113	116.59
Efficiency (%)	89.24	84.46
Time (second)	2.1	1.3

Table 2. Energy Sharing Scenario Comparison

8.1.3 Comparison

This section compares the performance of the energy sharing in the one-to-one and many-to-one scenarios. Comparing Figure 15(c) with Figure 16(c) shows that the many-to-one configuration shares energy faster than the one-to-one configuration. If we define the *energy sharing efficiency* as the ratio of the energy received to the energy supplied, then the two-to-one configuration achieves a lower energy sharing efficiency (i.e., 84.46%) than the one-to-one configuration (i.e., 89.24% shown in Table 2). One of the reasons is that the two-to-one configuration has more devices than the one-to-one configuration during the energy sharing process, which causes more energy loss.

8.2 Evaluation of Energy Network Protocol

Energy network protocols support a lot of applications, such as oil pipeline monitoring and climate monitoring. In this section, we use the application of climate monitoring and control in greenhouses as an example and evaluate the system performance under different settings. Since this work is the first one investigating energy sharing applications, the state-of-the-art research (e.g., energy conservation) is complementary, but provides no appropriate baseline for comparison. Therefore, we compare the system that enables network-wide global energy sharing (GES) with the same system working under the following two modes:

- **No energy sharing (NES):** The devices cannot share energy between each other within the network.
- **Local energy sharing (LES):** Each device shares energy only with its directly connected neighboring devices. For example, in Figure 10, device *a* shares energy only with de-

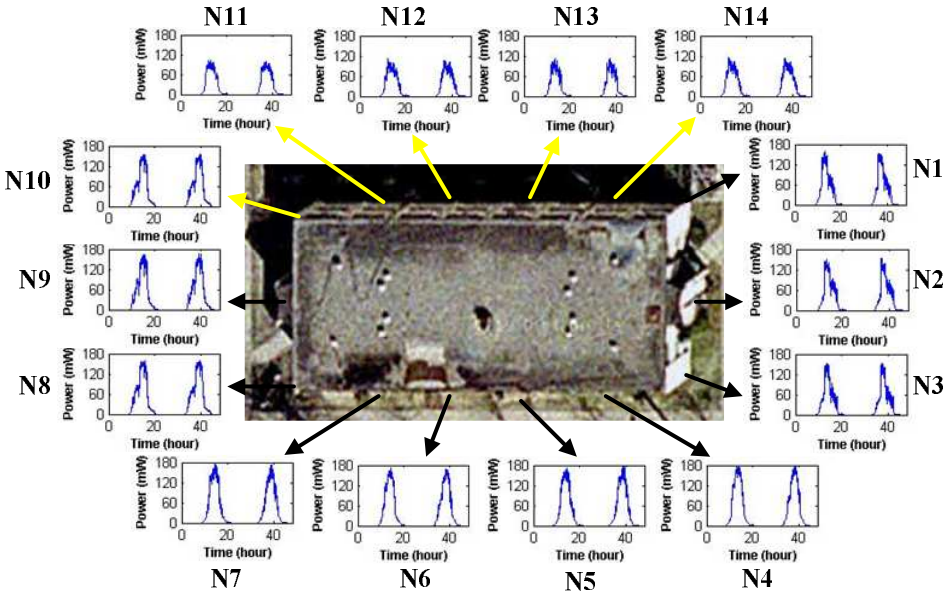
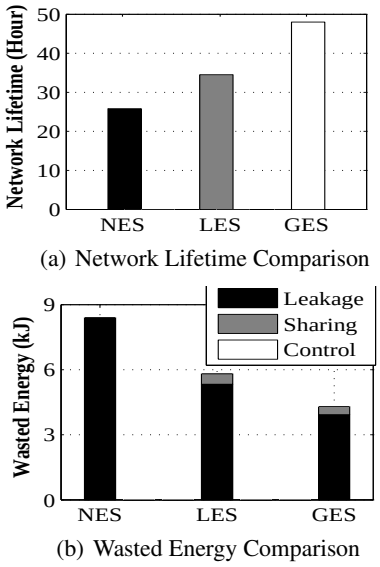


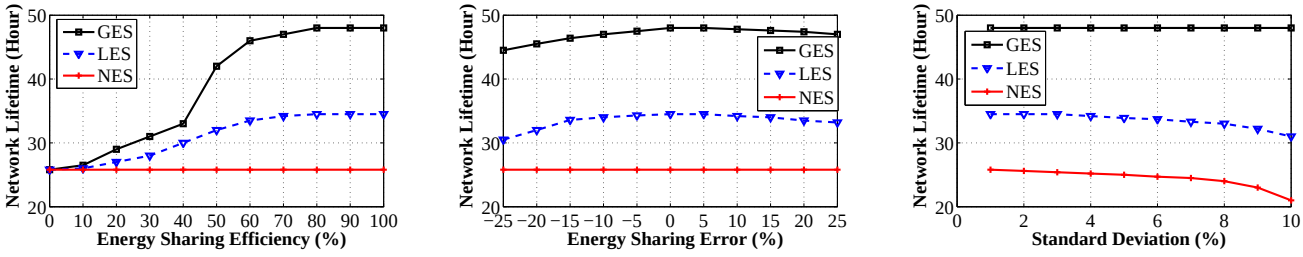
Figure 17. Experiment Site & Harvested Power Distribution



(a) Network Lifetime Comparison

(b) Wasted Energy Comparison

Figure 18. System Performance



(a) Impact of Sharing Efficiency

(b) Impact of Energy Sharing Error

(c) Impact of Unbalanced Workload

Figure 19. System Stability Analysis

vices b and c. This system serves as a baseline to evaluate the effectiveness of the network-wide energy sharing protocols.

Two metrics are used to evaluate the system performance:

• **Network Lifetime:** The time duration before the first device inside the network runs out of power.

• **Wasted Energy:** The summation of the energy leaked away and the energy loss during the energy sharing process. Here the energy loss includes three parts: (i) energy leaked away inside the capacitor array, (ii) energy consumption of the energy sharing control and communication, and (iii) energy loss when energy flows from one device to the other.

8.3 Experiment Setup

In this experiment, we continuously collected solar energy harvesting patterns from 14 locations outside of a 150-feet by 70-feet building (shown in Figure 17) for two days. The collected energy pattern is used as an input to simulate the performance of connected energy routers that work under 3 different modes: global energy sharing (GES), local energy sharing (LES), and no energy sharing (NES), for 48 hours per mode. Each device works under a set of randomly generated working patterns with the mean value of the duty cycle equaling 5%. The same working pattern is applied to the systems with global energy sharing, local energy sharing, and no energy sharing. Each data point on a graph represents the averaged value of 10 runs, and the 95% confidence intervals of the data are within $2 \sim 7\%$ of the mean shown.

8.4 System Performance

Figure 18 compares the performance of the systems working under no energy sharing (NES), local energy sharing (LES), and global energy sharing (GES) modes. Figure 18(a) shows that the system working under the GES mode maintains the aliveness of all the devices during the whole experiment period, so its network lifetime is 48 hours. Without network-wide energy sharing optimization, the devices working under the LES mode share energy only with their directly connected neighbors. The devices run out of energy when their neighbors do not have enough energy to share with them. This is the reason that the system working under the LES mode has a network lifetime of only 34.5 hours. Without energy sharing, the system working under the NES mode can keep all the devices alive for only 25.8 hours.

Figure 18(b) compares the total amount of energy wasted during the experiments. Without energy sharing, 8406J energy is leaked away inside the capacitor arrays of all the devices that are working under the NES mode. Energy sharing between neighboring devices can significantly reduce the energy leaked away. The devices working under the LES mode have a total of 5819.406J energy wasted, which includes (i) 5332J energy leaked away, (ii) 487J energy loss during the energy sharing process, and (iii) 0.406J energy consumed by sending the energy sharing control messages.

With network-wide energy sharing optimization, the devices working under the GES mode can further reduce the total amount of energy wasted to 4306.784J, which includes (i) 3930J energy leaked away, (ii) 376J energy loss during the energy sharing process, and (iii) 0.7836J energy consumed by sending the energy sharing control messages. The experiment results show that the energy consumption overhead of sending the energy sharing control messages is negligible.

8.5 System Sensitivity Analysis

Since our system is designed for many applications in different environments, it is very important to investigate the system's behavior and sensitivity under diverse settings.

8.5.1 Impact of Energy Sharing Efficiency

In this section, we analyze the impact of energy sharing efficiency by varying the value of energy sharing efficiency from 0% to 100%. Figure 19(a) shows the lifetime of the systems working under no energy sharing (NES), local energy sharing (LES), and global energy sharing (GES) modes. When the energy sharing efficiency equals 0, the systems with local energy sharing and global energy sharing have the same lifetime as the system with no energy sharing. As the energy sharing efficiency increases, the network lifetimes of the two systems with energy sharing also increase. When the energy sharing efficiency equals 100%, all of the nodes in the system with global energy sharing are alive all the time (i.e., network lifetime is 48 hours). However, the system with no energy sharing has a network lifetime of only 25.8 hours. Figure 19(a) also shows because the charge efficiency of rechargeable batteries is as low as 6% [37], the battery-based energy storage system cannot take advantage of the energy sharing feature.

8.5.2 Impact of Energy Sharing Error

As discussed in Section 8.1.2, the granularity of the A/D converter may introduce error during the energy sharing. In this experiment, we evaluated the impact of energy sharing error, which varies from -25% to 25%. Here we use a negative value to indicate that the actual energy received is less than the energy expected. For example, if device A wants to receive 100J energy from device B, -25% means device A actually only receives 75J. As shown in Figure 19(b), this energy sharing error could reduce the network lifetime. Moreover, a negative error has more impact on the system performance than a positive error. This is because less energy received by an energy-deficient device could reduce the lifetime of that device.

8.5.3 Impact of Unbalanced Workload

Different applications require the devices inside the network to work under different workloads. For example, the sensor nodes near the base station have a higher communication workload to forward the data packets than the sensor nodes that are far away from the base station. In this section, we investigate the impact of unbalanced workloads. Each device works under a series of randomly generated working patterns that have the same mean value of duty cycles but varied standard deviation values.

Figure 19 shows that the lifetimes of the local energy sharing and no energy sharing systems decrease as the value of the standard deviation increases from 1% to 10%. With network-wide energy sharing optimization, however, the global energy sharing system can always maintain the aliveness of all the devices and achieve the network lifetime of 48 hours under different standard deviation values.

9 Related Work

A number of sensor network systems have been deployed for applications such as military surveillance and tracking [25], assisted living [29], green buildings [15, 22], content sharing [14, 18, 27], localization [26, 7, 48], and video-enabled monitoring [36, 41]. However, most of the sensor devices for the above systems are powered by batteries. Due to limited battery capacity, energy is the major bottleneck for the above applications.

Energy conservation and management is an intensively studied area. Many solutions have been proposed at different layers, including physical layer energy optimization [16], I/O interface design [24], OS layer design [5, 10], link layer design [17], network layer design [6, 31, 49], and application-level energy-aware designs [4, 13, 39]. However, only a few works have focused on energy adaptation [46, 43] and energy storage systems [11, 23]. The differences between these works and ours are that (i) they focus on battery modeling or application adaptations, while ours focuses on capacitor-driven storage system design and control, and (ii) they currently do not incorporate the benefit of energy sharing among multiple devices.

The most closely related work is [42], which focuses on adjusting the sensor device's activities to synchronize the energy demand with supply in a small system powered by a single ultra-capacitor. Instead of adapting the sensor device's activities locally, this paper focuses on energy sharing between sensor devices. In addition, some researchers have investigated the use of multi-cell power sources [40, 45]. Unlike previous approaches, however, this paper introduces energy routers and investigates the energy access and routing protocols among multiple multi-cell power sources.

10 Conclusion

This work supports the concept of energy sharing among networked sensor devices using energy routers. By using a provable optimal energy accumulation and distribution network protocol, we can balance energy supply and demand at individual devices so that network lifetime is extended. Moreover, we can store and redistribute energy efficiently by considering the leakage models of connected devices. Although this work focuses on energy sharing using ultra-capacitor-based storages, the notion of energy sharing can be generically applied to other energy storages such as rechargeable batteries. The two additional challenges we shall address in the future are (i) how to improve charging and discharging efficiency in rechargeable batteries, and (ii) how to accurately predict remaining energy in batteries.

To our knowledge, this work is the first to consider collaboration between data networks and energy networks for efficient energy management. We invested a significant amount of effort to evaluate our design in six real-world settings. The results indicate the effectiveness of our design compared to other designs. In future work, we shall focus more on the design of applications that can benefit from energy sharing.

11 Acknowledgements

This work was supported in part by NSF grants CNS-0917097, CNS-0845994, CNS-0720465, SUTD grant SRG ISTD 2010 002, and a CI Fellowship. We also received partial support from InterDigital and Microsoft Research. We would also like to thank our shepherd, Dr. Deepak Ganesan, and the reviewers for their valuable feedback.

12 References

[1] J. Allred, A. B. Hasan, S. Panichsakul, W. Pisano, P. Gray, J. Huang, R. Han, D. Lawrence, and K. Mohseni. Sensorflock: an airborne wireless sensor network of micro-air vehicles. In *SenSys '07*.

[2] M. Azizyan, I. Constandache, and R. Choudhury. Surroundsense: mobile phone localization via ambience fingerprinting. In *MOBICOM '09*.

[3] G. Barrenetxea, F. Ingelrest, G. Schaefer, M. Vetterli, O. Couach, and M. Parlange. SensorScope: Out-of-the-Box Environmental Monitoring. In *IPSN '08*.

[4] J. Benson, T. O'Donovan, U. Roedig, and C. Sreenan. Opportunistic aggregation over duty cycled communications in wireless sensor networks. In *IPSN '08*.

[5] N. Brouwers, K. Langendoen, and P. Corke. Darjeeling, a feature-rich vm for the resource poor. In *SenSys '09*.

[6] A. Cerpa and D. Estrin. Ascent: Adaptive self-configuring sensor networks topologies. *IEEE Trans. Mob. Comput.*, 3(3), 2004.

[7] H. Chang, J. Tian, T. Lai, H. Chu, and P. Huang. Spinning beacons for precise indoor localization. In *SenSys '08*.

[8] A. G. Ciancio, S. Pattem, A. Ortega, and B. Krishnamachari. Energy-efficient data representation and routing for wireless sensor networks based on a distributed wavelet compression algorithm. In *IPSN '06*.

[9] ClimateMinder Products. `http://www.climateminder.com/nodes.html` .

[10] N. Cooprider, W. Archer, E. Eide, D. Gay, and J. Regehr. Efficient memory safety for tinyos. In *SenSys '07*.

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

[12] F. Miller, A. Vandome, and J. McBrewster. Wireless Energy Transfer. In *Publisher Alphascript Publishing*, 2009.

[13] C. Frank and K. Römer. Algorithms for generic role assignment in wireless sensor networks. In *SenSys '05*.

[14] G. Shen, Y. Li, and Y. Zhang. MobiUS: enable together-viewing video experience across two mobile devices. In *MobiSys '07*.

[15] M. Gauger, O. Saukh, and P. J. Marron. Talk to me! on interacting with wireless sensor nodes. *PerCom*, pages 1–8, 2009.

[16] J. Gummeson, S. S. Clark, K. Fu, and D. Ganesan. On the limits of effective hybrid micro-energy harvesting on mobile crfid sensors. In *MobiSys '10*.

[17] J. Gummeson, D. Ganesan, M. D. Corner, and P. J. Shenoy. An adaptive link layer for range diversity in multi-radio mobile sensor networks. In *INFOCOM '09*.

[18] J.-H. Hauer, V. Handziski, A. Köpke, A. Willig, and A. Wolisz. A component framework for content-based publish/subscribe in sensor networks. In *EWSN '08*.

[19] J. Eriksson, H. Balakrishnan, and S. Madden. Cabernet: Vehicular Content Delivery Using WiFi. In *MobiCom '08*.

[20] M. O. J. Moreno and J. Dixon. Energy-management system for a hybrid electric vehicle, using ultracapacitors and neural networks. *IEEE Trans. Industrial Electronics.*, 53:2(614-623), 2006.

[21] J. Yun, S. Patel, M. Reynolds, and G. Abowd. A Quantitative Investigation of Inertial Power Harvesting for Human-powered Devices. In *UbiComp '08*.

[22] X. Jiang, M. Ly, J. Taneja, P. Dutta, and D. Culler. Experiences with a high-fidelity wireless building energy auditing network. In *SenSys '09*.

[23] X. Jiang, J. Polastre, and D. E. Culler. Perpetual environmentally powered sensor networks. In *IPSN '05*.

[24] K. Klues, V. Handziski, C. Lu, A. Wolisz, D. E. Culler, D. Gay, and P. Levis. Integrating concurrency control and energy management in device drivers. In *SOSP '07*.

[25] V. Kulathumani, A. Arora, M. Sridharan, and M. Demirbas. Trail: A distance-sensitive sensor network service for distributed object tracking. *TOSN*, 5(2), 2009.

[26] B. Kusy, A. Ledeczi, and X. Koutsoukos. Tracking mobile nodes using rf doppler shifts. In *SenSys '07*.

[27] M. Li, T. Yan, D. Ganesan, E. Lyons, P. J. Shenoy, A. Venkataramani, and M. Zink. Multi-user data sharing in radar sensor networks. In *SenSys '07*.

[28] H. Lu, W. Pan, N. D. Lane, T. Choudhury, and A. T. Campbell. Soundsense: scalable sound sensing for people-centric applications on mobile phones. In *MobiSys '09*.

[29] D. Lymberopoulos, A. Bamis, T. Teixeira, and A. Savvides. Behaviorscope: Real-time remote human monitoring using sensor networks. In *IPSN '08*.

[30] S. Moeller, A. Sridharan, B. Krishnamachari, and O. Gnawali. Routing without routes: The backpressure collection protocol. In *IPSN '10*.

[31] S. Nath and P. B. Gibbons. Communicating via fireflies: geographic routing on duty-cycled sensors. In *IPSN '07*.

[32] NESSCAP Ultracapacitor Products. `http://www.nesscap.com/data_nesscap/spec_sheets/Spec%2009.pdf` .

[33] P. Banerjee, I. Perez, L. Lecordier, S. Lee, and G. Rubloff. Nanotubular Metalinsulatormetal Capacitor Arrays for Energy Storage. In *Nature Nanotechnology 4, 292-296, 2009)*.

[34] P. Zhang, C. Sadler, S. Lyon, and M. Martonosi. Hardware Design Experiences in ZebraNet. In *SenSys '04*.

[35] 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.

[36] A. Rowe, D. Goel, and R. Rajkumar. Firefly mosaic: A vision-enabled wireless sensor networking system. In *RTSS '07*.

[37] S. Foster. *The energy efficiency of common household battery charging systems: Results and implications.* `http://www.efficientproducts.org/reports/bchargers/NRDC-Ecos\_Battery\_Charger\_Efficiency.pdf` .

[38] Sea-Bird Electronics, Inc. *Nickel-Metal Hydride (NiMH) Battery Charger and Battery Pack. User's Manual*, 2009.

[39] N. Sharma, J. Gummeson, D. Irwin, and P. Shenoy. Cloudy computing: Leveraging weather forecasts in energy harvesting sensor systems. In *SECON '10*.

[40] H. Shibata, S. Taniguchi, K. Adachi, K. Yamasaki, G. Ariyoshi, K. Kawata, K. Nishijima, and K. Harada. Management of seriallyconnected battery system using multiple switches. In *In Power Electronics and Drive Systems*, 2001.

[41] J. Shin, R. Kumar, D. Mohapatra, U. Ramachandran, and M. H. Ammar. Asap: A camera sensor network for situation awareness. In *OPODIS '07*.

[42] T. Zhu, Z. Zhong, Y. Gu, T. He, and Z. Zhang. Leakage-Aware Energy Synchronization for Wireless Sensor Networks. In *MobiSys '09*.

[43] C. Vigorito, D. Ganesan, and A. Bartoeeee. Adaptive Control of Duty Cycling in Energy-Harvesting Wireless Sensor Networks. In *SECON '07*.

[44] 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.

[45] R. Wiegers, D. Blackketter, and H. Hess. Modelling performance of ultracapacitor arrays in hybrid electric vehicles. In *International Journal of Alternative Propulsion 2006 - Vol. 1, No.1 pp. 32 - 46*.

[46] X. Liu, P. Shenoy, and M. D. Corner. Chameleon: Application Level Power Management. *IEEE Transactions on Mobile Computing*, 2008.

[47] Y. Agarwal, T. Weng, and R. Gupta. Micro-Systems Driving Smart Energy Metering in Smart Grids. In *DAC '07*.

[48] P. Zhang and M. Martonosi. Locale: Collaborative localization estimation for sparse mobile sensor networks. In *IPSN '08*.

[49] T. Zhu, Z. Zhong, T. He, and Z.-L. Zhang. Exploring link correlation for efficient flooding in wireless sensor networks. In *NSDI '10*.