Power Management in Energy Harvesting Sensor Networks

AMAN KANSAL, JASON HSU, SADAF ZAHEDI, and MANI B. SRIVASTAVA
University of California, Los Angeles

Power management is an important concern in sensor networks, because a tethered energy infrastructure is usually not available and an obvious concern is to use the available battery energy efficiently. However, in some of the sensor networking applications, an additional facility is available to ameliorate the energy problem: harvesting energy from the environment. Certain considerations in using an energy harvesting source are fundamentally different from that in using a battery, because, rather than a limit on the maximum energy, it has a limit on the maximum rate at which the energy can be used. Further, the harvested energy availability typically varies with time in a nondeterministic manner. While a deterministic metric, such as residual battery, suffices to characterize the energy availability in the case of batteries, a more sophisticated characterization may be required for a harvesting source. Another issue that becomes important in networked systems with multiple harvesting nodes is that different nodes may have different harvesting opportunity. In a distributed application, the same end-user performance may be achieved using different workload allocations, and resultant energy consumptions at multiple nodes. In this case, it is important to align the workload allocation with the energy availability at the harvesting nodes. We consider the above issues in power management for energy-harvesting sensor networks. We develop abstractions to characterize the complex time varying nature of such sources with analytically tractable models and use them to address key design issues. We also develop distributed methods to efficiently use harvested energy and test these both in simulation and experimentally on an energy-harvesting sensor network, prototyped for this work.

Categories and Subject Descriptors: C.4 [**Computer Systems Organization**]: Performance of systems—*Modeling Techniques*; C.2.4 [**Computer Systems Organization**]: Computer Communication Networks—*Distributed Systems*

General Terms: Algorithms, Design, Experimentation, Measurement, Performance, Theory

Additional Key Words and Phrases: Adaptive duty cycling, Heliomote, lifetime, power management, energy neutrality

ACM Reference Format:

Kansal, A., Hsu, J., Zahedi, S., and Srivastava, M. B. 2007. Power management in energy harvesting sensor networks. ACM Trans. Embedd. Comput. Syst. 6, 4, Article 32 (September 2007), 38 pages. DOI = 10.1145/1274858.1274870 http://doi.acm.org/ 10.1145/1274858.1274870

Authors' address: Aman Kansal, Microsoft Research; email: kansal@microsoft.com; Jason Hsu, The Aerospace Corporation; Sadaf Zahedi and Mani B. Srivastava, University of California at Los Angeles, Los Angeles, California 90024.

1. INTRODUCTION

Wireless and embedded systems are commonly powered using batteries. For applications where the system is expected to operate for long durations, energy becomes a severe bottleneck and much effort has been spent on the efficient use of battery energy. More recently, another alternative has been explored to supplement or even replace batteries: harvesting energy from the environment. In this paper, we are concerned with the efficient use of harvested energy.

We define an energy-harvesting node as any system which draws part or all of its energy from the environment. A key distinction of this energy from that stored in the battery is that this energy is potentially infinite, though there may be a limit on the rate at which it can be used. For example, a desk calculator using a solar cell is an example of a harvesting node. A network of harvesting nodes will be referred to as a harvesting network. We allow each node in such a network to use the same or different harvesting technologies and some nodes may not be capable of harvesting energy at all.

In a battery-powered device, the typical power-management design goals are to minimize the energy consumption [Sinha and Chandrakasan 2001; Min et al. 2000a] or to maximize the lifetime achieved [Singh et al. 1998; Younis et al. 2002; Shah and Rabaey 2002; Li et al. 2001] while meeting required performance constraints. In an energy-harvesting node, one mode of usage is to treat the harvested energy as a supplement to the battery energy and again, a possible power-management objective is to maximize the lifetime. However, in the case of harvesting nodes, another usage mode is possible—using the harvested energy at an appropriate rate such that the system continues to operate perennially. We call this mode energy-neutral operation: a harvesting node is said to achieve energy-neutral operation if a desired performance level can be supported forever (subject to hardware failure).

In this mode, the power-management design considerations are very different from those of maximizing lifetime. Two design considerations are apparent:

1. Energy-Neutral Operation: How to operate such that the energy used is always less than the energy harvested? The system may have multiple distributed components each harvesting its own energy and the performance then not only depends on the spatiotemporal profile of the available energy, but also on how this energy is used to deliver network-wide performance guarantees.
2. Maximum Performance: While ensuring energy-neutral operation, what is the maximum performance level that can be supported in a given harvesting environment? Again, this depends on the harvested energy at multiple distributed components.

A naïve approach would be to develop a harvesting technology whose minimum energy output at any instant is sufficient to supply the maximum power required by the load. This, however, has several disadvantages, such as high costs, and may not even be feasible in many situations. For instance, when harvesting solar energy, the minimum energy output for any solar cell would be zero at night and this can never be made more than the power required by the load.

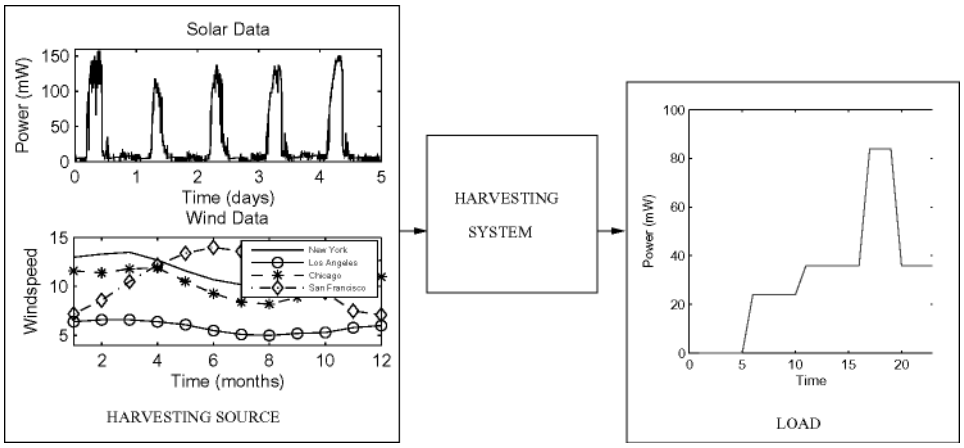


Fig. 1. Harvesting energy from the environment.

A more reasonable approach is to add a power management system between the harvesting source and the load, which attempts to satisfy the energy consumption profile from the available generation profile (Figure 1). We explore this approach in greater detail.

The three main blocks shown in the figure are:

- *Harvesting Source.* This refers to any available harvesting technology, such as a solar cell, a wind turbine, piezo-electric harvester or other transducer, which extracts energy from the environment. The energy output varies with time, depending on environmental conditions, which are typically outside the control of the designer. For instance, Figure 1 shows two possible power output variations with time—a solar cell output on a diurnal scale and wind speeds at four arbitrarily chosen locations [Wind Data 2001] on an annual scale. In a distributed system, multiple such harvesting sources may be present at multiple nodes at different locations.
- *Load.* This refers to the energy consuming activity being supported. A load, such as a sensor node, may consist of multiple subsystems and energy consumption may be variable for its different modes of operation. For instance, the activity may involve sampling a sensor, transmitting the sensed value, and receiving an acknowledgment. Figure 1 shows different power levels of a Mica2 mote in sleep state, and with processor on, with its radio transmitting and receiving. In a harvesting network, the load may be an application layer activity, which requires the expenditure of energy at multiple nodes in the system, such as routing a data packet from one location to another.
- *Harvesting System.* This refers to the system designed specifically to support a variable load from a variable energy-harvesting source when the instantaneous power supply levels from the harvesting source are not exactly matched to the consumption levels of the load. In a harvesting network, this may also involve collaboration among the power-management systems of the constituent nodes to support distributed loads from the available energy. We will focus on the design of this system.

There are two ways in which the load requirements may be reliably fulfilled from a variable supply. One is to use an intermediate energy buffer in the harvesting system, such as a battery or an ultracapacitor. Second, is to modify the load consumption profile according to the availability. In practice, neither of these approaches alone may be sufficient, since the load cannot be arbitrarily modified and energy storage technologies have nonideal behavior that causes energy loss.

1.1 Outline

The goals of the work presented in this paper are to understand the various issues involved in the efficient use of harvested energy and to note how they differ from or are similar to power management in a battery-driven system. This understanding is then used for developing practical power-management strategies for harvesting nodes and networks.

In the next section, we discuss the condition for ensuring energy-neutral operation in more detail. We develop abstractions that help model the variability of energy sources and energy consumption patterns in a general sense, and then adapt these for most common environmental energy sources used for harvesting. In Section 3, we show how these abstractions help derive important system design parameters, such as the minimum battery size needed for efficiently using a given energy source.

In Section 4, we develop practical methods for a harvesting system to achieve energy-neutral operation. It may be noted that the exact energy profile over time and, in some situations, even the expected energy usage may not be known a priori. Our methods learn these variables over time and adapt operation accordingly.

Section 5 discusses the energy-harvesting issues for a harvesting network. The network performance in such a system depends on the operation of multiple nodes and workload allocation may have to be aligned with the availability of energy at each node in such a way as to achieve the overall network performance objective. We provide examples of how optimal performance may be determined in a harvesting network, and some practical methods that attempt to achieve it.

Finally, we summarize the related work in Section 6 and conclude the paper in Section 7.

2. HARVESTING THEORY

This section develops some useful abstractions for energy sources and energy consumers, in order to analyze the requirements for energy-neutral operation. Note that the concept of lifetime is not identical to that in a battery-powered system, since even a node which exhausted its battery may start operating again at the next available energy-harvesting opportunity. Thus, we use a different metric—energy-neutral operation.

Intuitively, energy-neutral operation can be expected in situations where energy used by the system is less than the energy harvested from the environment. A more precise statement of this requirement, however, requires considering the exact system constraints under which energy is used.

Energy sources may be classified into the following types:

1. Uncontrolled but predictable: Such an energy source cannot be controlled to yield energy at desired times but its behavior can be modeled to predict the expected availability at a given time within some error margin. For example, solar energy cannot be controlled. However, models for its dependence on diurnal and seasonal cycles are known and can be used to predict availability. The prediction error may be improved using commonly available weather forecasts for the region where a system is deployed.
2. Uncontrollable and unpredictable: Such an energy source can not be controlled to generate energy when desired and yields energy at times which are not easy to predict using commonly available modeling techniques or the when the prediction model is too complex for implementation in an embedded system. For example, vibrations in an indoor environment may be harvested to yield energy using methods such as Roundy et al. [2004], but predicting the vibration patterns may be impractical.[1]
3. Fully controllable: Energy can be generated when desired. For example, consider self-power flashlights, which the user may shake to generate some energy whenever needed.
4. Partially controllable: Energy generation may be influenced by system designers or users but the resultant behavior is not fully deterministic. For example, an RF energy source may be installed in a hall and multiple harvesting nodes, such as RFID's, may extract energy from it. However, the exact amount of energy produced at each node depends on RF propagation characteristics within the environment and cannot be controlled.

2.1 Conditions for Energy-Neutral Operation

Let us now consider the loads which use the energy source. Suppose the power output from the energy source is $P_s(t)$ at time t, and the energy being consumed at that time is $P_c(t)$. The following three cases can be separated to model the energy behavior of a load and write the physical condition on energy conservation. These conditions will help us derive requirements on $P_s(t)$ and $P_c(t)$, which allow energy-neutral operation to be guaranteed.

- **Harvesting system with no energy storage:** The first case considers a harvesting system that has a transducer to extract energy from the environment and this energy is directly used by the load. There is no facility to store energy. For example, consider the device in Paradiso and Feldmeier [2001], which generates energy from the press of a button and this energy is used to transmit a radio packet during the button press itself. A water-powered flour mill is another example: the mill operates while the water is flowing.
  For such harvesting devices, the device can operate at all t when

$$P_s(t) \geq P_c(t) \tag{1}$$

[1]The unpredictable nature is purely an engineering consideration and we do not attempt to prove when a particular energy availability function is unpredictable. It is likely that having a sufficiently sophisticated model for any phenomenon renders it predictable.

Any energy received at times when $P_s(t) < P_c(t)$ is wasted. Also, when $P_s(t) \geq P_c(t)$, the energy, $P_s(t) - P_c(t)$ is wasted.

- **Harvesting system with ideal energy buffer.** In many instances, the energy generation profile may be very different from the consumption profile. To help support this scenario, consider a device that has an ideal mechanism to store any energy that is harvested. The stored energy may be used at any time later. The ideal energy buffer is defined to be a device that can store any amount of energy, does not have any inefficiency in charging, and does not leak any energy over time. For this case, the following equation should be satisfied for all non negative values of T:

$$\int_0^T P_c(t)dt \leq \int_0^T P_s(t)dt + B_0 \quad \forall \quad T \in [0, \infty) \tag{2}$$

where B_0 is the initial energy stored in the ideal energy buffer. Note that condition (1) is sufficient to ensure condition (2) but not necessary.

- **Harvesting system with nonideal energy buffer.** The above two cases are extremes of a spectrum and may not be typical. A more practical case is that of a harvesting system, which has a battery or an ultracapacitor to store energy. Such an energy, storage mechanism is not ideal in the sense defined in the previous case: the energy capacity is limited, the charging efficiency, η, is strictly less than 1 and some energy is lost through leakage. The conditions arising because of energy conservation and buffer size limit are discussed below. First define a rectifier function $[x]^+$ as follows:

$$[x]^+ = \begin{cases} x & x \geq 0 \\ 0 & x < 0 \end{cases}$$

Then, energy conservation leads to:

$$B_0 + \eta \int_0^T [P_s(t) - P_c(t)]^+ dt - \int_0^T [P_c(t) - P_s(t)]^+ dt - \int_0^T P_{leak}(t)dt \geq 0 \quad \forall T \in [0, \infty) \tag{3}$$

where $P_{leak}(t)$ is the leakage power for the energy buffer. This does not account for the energy buffer size. The buffer size limit requires the following additional constraint to be satisfied:

$$B_0 + \eta \int_0^T [P_s(t) - P_c(t)]^+ dt - \int_0^T [P_c(t) - P_s(t)]^+ dt - \int_0^T P_{leak}(t)dt \leq B \quad \forall T \in [0, \infty) \tag{4}$$

where B is the size of the energy buffer. Note that while Eq. (3) is a sufficient and necessary condition to be satisfied by all allowable $P_s(t)$ and $P_c(t)$, the condition (4) is only sufficient but not necessary—some functions not satisfying this may be allowable. This happens because excess energy not used or stored in the buffer can be dissipated as heat from the system. In this case, the left-hand side of Eq. (3) will be strictly greater than zero, by the amount of energy wasted. The condition (4) becomes necessary if wasting energy is not allowed.

The above conditions are stated for general forms of P_s and P_c. Next, we will develop models that help characterize practical energy sources and loads. For these models, we will derive the requirements for energy-neutral operation, namely, the relationships between P_s, P_c, and B.

2.2 System Models and Observations

Consider, first, the case of a harvesting system with no energy storage. Here, if $P_c(t)$ is a binary valued function, such as for a device that can either be active, at a fixed power level or inactive at a zero power level, then no power management is required because the device will automatically be shut down when enough energy is not available. As an example, consider a sensor node installed to monitor the health of heavy-duty industrial motors. Suppose the node operates using energy harvested from the machine's vibrations, the harvested power is greater than the consumed power and the health monitoring function is desired only when the motor is powered on. No power management is required in this case. If on the other hand, $P_c(t)$ can be controlled, such as using dynamic voltage scaling (DVS) [Min et al. 2000b], or by powering off sub systems within the device, then the best power-management strategy is to match the $P_c(t)$ to the available $P_s(t)$. For instance, in the above motor health-monitoring example, suppose that the motor may be operated at variable speeds and the vibration energy is proportional to the motor speed. Then, the sensor node may use DVS to adjust its processing and sampling rate to match the power level available at any time. The monitoring performance will vary with the motor speed.

Consider next the case when the harvesting system has a nonideal energy buffer. In this case, operation at any time t can be ensured by using proper power-management strategies, which store some energy for times when $P_s(t)$ is below desired $P_c(t)$. To this end, we begin with a model to characterize $P_s(t)$.

The first modeling parameter is the average rate at which energy is provided by the source. Second, we wish to characterize the variability of the source in a general sense. Similarly, we need a model for the energy consumption profile.

We define the following model that is motivated by leaky-bucket Internet traffic models [Cruz 1991a; Parekh and Gallager 1993]. However, there is a difference in our model, because while in Internet traffic policing a limit is only needed on the maximum traffic bursts, in harvesting energy, on the other hand, we wish to bound both the maximum and minimum energy outputs.

Definition 2.1 (*$(\rho, \sigma_1, \sigma_2)$ Function*). A nonnegative, continuous and bounded function $P(t)$ is said to be a $(\rho, \sigma_1, \sigma_2)$ function if, and only if, for any value of finite positive real numbers τ and T, the following are satisfied:

$$\int_{\tau}^{\tau+T} P(t)dt \;\leq\; \rho T + \sigma_1 \tag{5}$$

$$\int_{\tau}^{\tau+T} P(t)dt \;\geq\; \rho T - \sigma_2 \tag{6}$$

This model may be used for an energy source or a load. For instance, if the harvested energy profile $P_s(t)$ is a $(\rho_1, \sigma_1, \sigma_2)$ function, then the average rate at which energy is available over long durations becomes ρ_1 and the burstiness

is bounded by σ_1 and σ_2. Similarly, suppose $P_c(t)$ is modeled as a $(\rho_2, \sigma_3, \sigma_4)$ function.

Further, the leakage from the energy buffer is typically modeled using a constant leakage current and, thus, we may take $P_{leak}(t) = \rho_{leak} \forall t$.

For the above forms of energy profiles, evaluating condition (3) leads to:

$$B_0 + \eta \cdot \min\left\{\int_T P_s(t)dt\right\} - \max\left\{\int_T P_c(t)dt\right\} - \int_T P_{leak}(t)dt \;\geq\; 0 \tag{7}$$

$$\Rightarrow B_0 + \eta(\rho_1 T - \sigma_2) - (\rho_2 T + \sigma_3) - \rho_{leak} T \;\geq\; 0 \tag{8}$$

Since the energy models above do not constraint the time intervals for which $P_s > P_c$ or vice versa, we have considered the worst-case scenario. The worst energy utilization occurs when the bursts of energy production from the harvested source are completely nonoverlapping with the bursts of consumption in the load, because this causes all the harvested energy to be first stored in a nonideal buffer and then used. This explains the usage of max and min functions above. Thus, Eq. (8) is sufficient to ensure energy neutral operation but not necessary.

We can ensure energy neutrality by requiring Eq. (8) to be satisfied for all $T \geq 0$. Substituting $T = 0$ yields:

$$B_0 \geq \eta\sigma_2 + \sigma_3 \tag{9}$$

This gives a condition on the initial energy stored in the battery. Next, taking the limit $T \to \infty$ in (8) yields:

$$\eta\rho_1 - \rho_{leak} \geq \rho_2 \tag{10}$$

On the other hand, substituting these energy models in Eq. (4), and again considering the worst-case scenario yields:

$$B_0 + \eta \cdot \max\left\{\int_T P_s(t)dt\right\} - \min\left\{\int_T P_c(t)dt\right\} - \int_T P_{leak}(t)dt \;\leq\; B \tag{11}$$

$$\Rightarrow B_0 + \eta(\rho_1 T + \sigma_1) - (\rho_2 T - \sigma_4) - \rho_{leak} T \;\leq\; B \tag{12}$$

Substituting $T = 0$, we obtain:

$$B_0 + (\eta\sigma_1 - \sigma_4) \leq B \tag{13}$$

Using Eq. (9), this provides a constraint on the required battery size:

$$B \geq \eta(\sigma_1 + \sigma_2) + \sigma_3 - \sigma_4 \tag{14}$$

Also, taking the limit $T \to \infty$ in (12) yields:

$$\eta\rho_1 - \rho_{leak} \leq \rho_2 \tag{15}$$

Intuitively, the above two equations may be interpreted as follows. The battery is required to make up for the burstiness of the energy supply and consumption and the limiting case of $T = 0$ models the situation when energy production or consumption happen in impulsive bursts. Thus, this limiting case yields the maximum battery size required to buffer those energy bursts. The limiting case $T \to \infty$ corresponds to the long-term behavior and, hence, yields the sustainable rates without bursts.

We now consider a special case of the constraints derived above that is important for many practical systems. Recall that Eq. (4) was not a necessary condition and we had noted that some forms of functions P_s and P_c not satisfying it may be feasible. A particularly interesting special case is that of a load, that does not maintain a non-zero consumption rate. Model this load as follows:

$$\int_T P_c(t) \leq \rho_2 T + \sigma_3 \tag{16}$$

$$\int_T P_c(t) \geq 0 \tag{17}$$

Denote this a (ρ_2, σ_3) load. In this case, since the load can have a zero consumption, energy production may exceed consumption and, hence, energy may sometimes have to be wasted when the production exceeds the finite storage capacity. Clearly, the constraints derived from Eq. (4), in particular, the constraints (14) and (15), are no longer needed. Instead of Eq. (4), the relevant requirement is that enough of the harvested energy must be stored to support the maximum consumption in the load. This implies that energy produced and stored should be sufficient to meet the consumption requirements, which leads to the same conditions on ρ_2 and B_0 as before, shown in Eqs. (9) and (10).

The observations above, and the fact that the battery size must be larger than the initial energy store, lead to the following conclusion:

THEOREM 2.2 (ENERGY-NEUTRAL OPERATION). *Consider a harvesting system in which the energy production profile is characterized as a* $(\rho_1, \sigma_1, \sigma_2)$ *function, the load is characterized by a* (ρ_2, σ_3) *function, and the energy buffer is characterized by parameters* η *for storage efficiency, and* ρ_{leak} *for leakage. The following conditions are sufficient for the system to achieve energy-neutrality:*

$$\rho_2 \leq \eta\rho_1 - \rho_{leak} \tag{18}$$

$$B_0 \geq \eta\sigma_2 + \sigma_3 \tag{19}$$

$$B \geq B_0 \tag{20}$$

where B *denotes the capacity of the energy buffer and* B_0 *is the initial energy stored in the buffer.*

The case of a harvesting system with ideal energy buffer can be obtained as a special case of the above by substituting $\eta = 1$, $\rho_{leak} = 0$ and was considered in Kansal et al. [2004].

3. DESIGN IMPLICATIONS AND EXAMPLES

Let us now consider the implications of the above observations for practical harvesting system design. In particular, we will discuss the following three issues: energy buffer size, operational performance level, and the measurement capabilities required in hardware for harvesting.

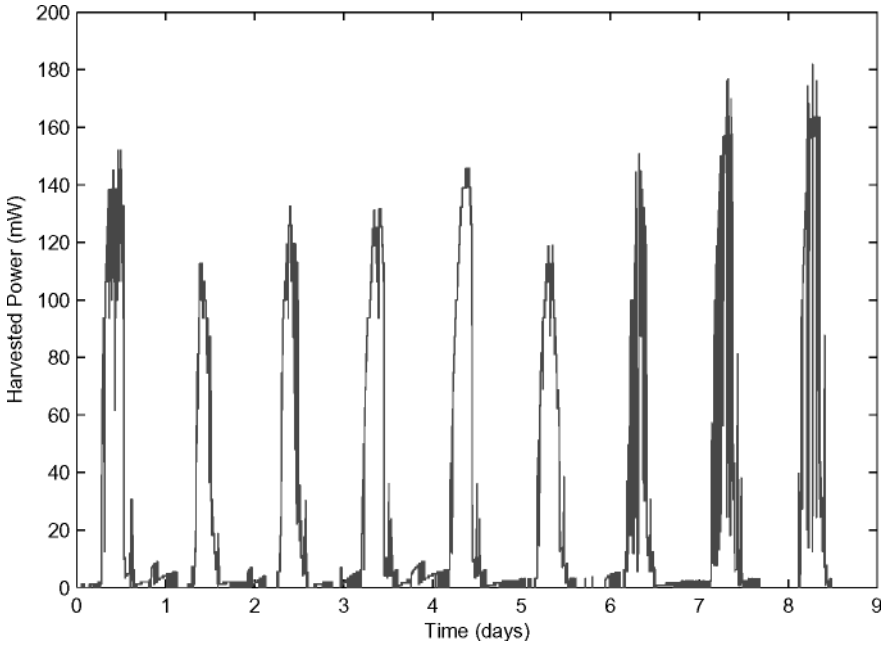


Fig. 2. Solar energy based charging power recorded for 9 days.

Table I. Solar Cell Parameters in Experimental Environment

Parameter	Value	Units
ρ_1	23.6	mW
σ_1	1.4639×10^3	J
σ_2	1.8566×10^3	J

3.1 Buffer Size and Related Considerations

The first direct implication is on the design of the energy buffer required in the harvesting system. As an example, consider a harvesting system that harvests solar energy. The power output from a solar cell [Kansal et al. 2004] is plotted in Figure 2 for 9 days. Assuming that this data is representative of the solar energy received on typical days of operation, this energy generation profile may be characterized by the $(\rho_1, \sigma_1, \sigma_2)$ model in Table I.

Let us assume that the load can be designed to operate at constant power consumption $\eta\rho_1 - \rho_{leak}$, where ρ_{leak} will depend on energy-storage technology used. Then, $\sigma_3 = 0$. The battery size required can be obtained from Theorem 2.2. Several technologies are available to implement this energy buffer, such as NiMH batteries, Li ion batteries, ultracapacitors or NiCd batteries. For instance, for NiMH batteries, $\eta = 0.7$ and the required size becomes 1.30×10^3J. This can be easily provided by an AA-sized NiMH battery, which has a capacity of 1800 mAh, i.e., 7.7×10^3J.

Note that using a larger battery than the above size does not help improve the supported energy-neutral performance level. A larger battery

than that calculated above may, however, be used to provide for practical considerations:

1. In a practical system, there may be some error in learning the harvesting source model parameters and using a larger battery will provide a tolerance for such error.
2. Battery storage capacity degrades with multiple charge–discharge cycles. For instance, after about 500 deep charge-discharge cycles, the storage capacity of an NiMH battery falls to 80% of its original. Using a larger battery will make the discharge cycle shallow, which slows down the degradation significantly—the relationship between depth of discharge and cycle life is logarithmic [Mpower 2005]. For instance, the same degradation, as mentioned above, occurs with 5000 charge–discharge cycles, when each charge discharge is limited to 10% of the battery capacity. Also, the increased capacity will mean that even after degradation the battery capacity is sufficient to meet the required storage size constraint.

Once the battery size has been determined, other practical considerations may help decide which specific energy-storage technology is used to achieve this capacity. For instance, the recommended charging current is high for Li ion batteries, and for the required battery capacity, this may never be supplied by the harvesting source. For NiCd batteries, the charging current is acceptable, but memory effect makes its use for partial charge and recharge cycles inappropriate. Ultracapacitors have a high η, but also high leakage, which makes $\eta\rho_1 - \rho_{leak}$ much smaller than that achieved using batteries. Hence, the NiMH battery seems best suited for this purpose. Additional factors that concern the designer may include the presence of toxic substances, such as in NiCd, the requirement for complex control circuitry, such as required for Li ion, or if the battery is recyclable.

3.2 Achievable Performance Level

Second, we discuss the operational performance level, that can be supported in energy-neutral mode. The calculation in the above example yield $\rho_2 = \eta\rho_1 - \rho_{leak} = 15.92$ mW at $\rho_{leak} = 0.6$ mW for a typical AA sized NiMH battery. Now, if the load consumes more power than this, its performance must be scaled down to this level. Several techniques may be available to scale the performance, depending on the hardware capabilities of the load, such as duty-cycling among low-power modes or dynamic voltage scaling. For instance, consider a sensor node, MicaZ [Motes 2005], as the load in the harvesting node. The maximum power consumption of this load is 90 mW and, hence, to achieve the available ρ_2, one must use a duty-cycle of 17.7% or lower. Practical schemes for achieving this duty cycle are discussed in a later section. Suppose, on the other hand, a Stargate [Stargate 2004] was to be used as the load. The average power consumption of this load is 1500 mW and, hence, a duty cycle of only 1% may be supported. If this duty cycle is not useful for the application, other power-scaling methods, such as DVS, may be used to reduce the power consumption. Ultimately, if the application performance cannot be met with these methods,

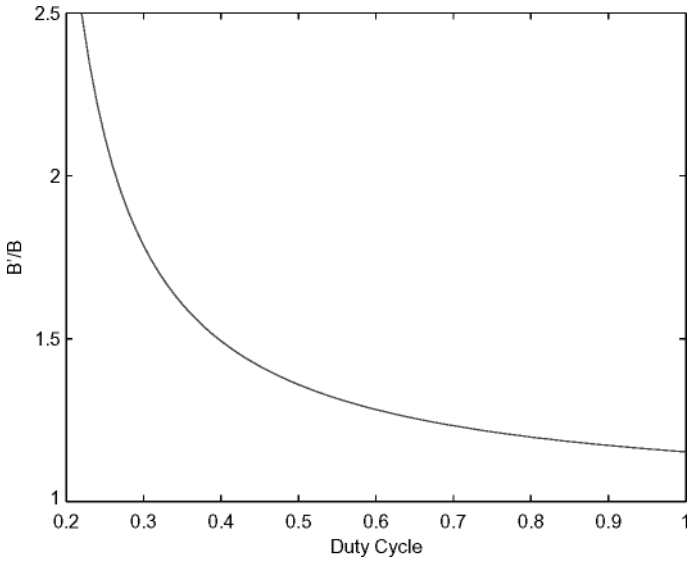


Fig. 3. Increase in battery size if no harvesting used.

the system design may have to be changed to harvest more energy, such as by using larger solar cells.

The duty cycle determined using Theorem 2.2 is for energy-neutral operation. It is very much possible to operate at a performance level above this. Performance is then battery dominated and power-management strategies for that mode may be designed to maximize lifetime. Here, solar energy supplements stored energy to prolong battery life. While the same effect could have been achieved using a larger battery, using a harvesting technology may be beneficial in certain situations. Below, we explore the equivalent battery size increase required for supporting a given power consumption level, to achieve the same lifetime as enabled using the harvesting method. Suppose the harvested energy produced is ρ_1 and the load operates at ρ_2. Suppose the battery size with harvesting is B and the achieved lifetime is L_T. Then:

$$\eta\rho_1 * L_T + B = \rho_2 * L_T \tag{21}$$

$$\Rightarrow L_T = \frac{B}{\rho_2 - \eta\rho_1} \tag{22}$$

Denote the larger battery required to achieve the same lifetime without any harvesting as B'. Then $B' = \rho_2 * L_T$, which gives:

$$B' = \frac{\rho_2 B}{\rho_2 - \eta\rho_1} \tag{23}$$

We have ignored the leakage power for simplicity and, hence, the value of B' required in a practical system will, in fact, be larger than that calculated above. For the harvesting data shown in Figure 2, and the duty-cycle based performance scaling shown in the example MicaZ load above, Figure 3 plots the normalized battery increase B'/B. Clearly, no finite battery size can achieve

energy-neutral operation, and a large increase in battery size is required if operating marginally above the energy-neutral performance level.

Depending on the cost and feasibility of using energy-harvesting as compared to the cost of the larger battery, the appropriate alternative may be chosen.

3.3 Measurement Support

Any power-management algorithm would typically need information about available energy resources. Many battery-operated devices, ranging from handhelds to laptops, provide the facility to monitor the residual battery, which has been used in algorithms for maximizing lifetime [Singh et al. 1998; Younis et al. 2002; Shah and Rabaey 2002; Li et al. 2001; Chang and Tassiulas 2000; Maleki et al. 2003; Rodoplu and Meng 1998; Gallager et al. 1979]. In harvesting nodes, however, monitoring the residual battery is not sufficient. If the above theorems and the corresponding energy source characterization is to be used for implementing practical harvesting-aware power-management schemes, the energy input from the environment must be measured.

The first required measurement is, thus, the amount of environmental energy extracted by the device. A second related measurement is the variability in this energy supply. This is used, for instance, to determine the parameters σ_1 and σ_2 in the above theory. Also, practical power scaling schemes may use this information to assess the certainty in the availability measurement. Third, it may be helpful to know when the environmental energy is available. This happens, for instance, when, to avoid the energy loss because of battery storage inefficiency, delay tolerant tasks are carried out when the environmental supply is directly available.

Let us consider how these parameters can be measured. If the residual battery can be accurately measured and the power consumption of the system, $P_c(t)$ is known, then the following simple scheme can be used: Measure the battery level at times t_1 and t_2 to be $E_b(t_1)$ and $E_b(t_2)$, respectively. The environmental energy extracted between t_1 and t_2, denoted E_e, is then given by:

$$E_e = \left[E_b(t_2) - E_b(t_1) + \int_{t_1}^{t_2} P_c(t) \right]^+ \tag{24}$$

There are however, some problems with this approach:

1. The residual battery energy measurement is typically based on battery voltage. The change in battery voltage with small changes in residual energy, such as a few percentage of battery life, is too small to yield reliable residual energy estimates. This requires that t_1 and t_2 be chosen far apart, making this measurement a slow process.
2. Choosing t_1 and t_2 far apart also makes it hard to measure when the energy was available. Further, the data on variability in energy supply cannot be measured at fine resolutions in time.
3. Knowing $P_c(t)$ accurately is not easy, in practice. Typical devices, in particular sensor nodes, consist of multiple components each of which is used as required by the application and each of which may be individually power

Fig. 4. Heliomote: an energy-harvesting sensor node, which provides environmental energy tracking capabilities.

scaled to minimize consumption. Power consumption thus varies depending on what application is using the device. For instance, the power consumption when transmitting on the radio differs from when receiving or when the radio is deactivated.

A better method to estimate the energy input then is to measure the current flowing out of the harvesting source and its voltage. This immediately yields the instantaneous power input at any time point. Data about when and how much environmental energy is available is directly provided by these measurements. Also, these measurements can be tracked at the desired resolution in time to estimate the variability of the energy source. These measurements are provided in our solar energy-harvesting sensor node, named Heliomote. Unlike many harvesting nodes, which only provide the functionality to extract energy from the environment, the Heliomote also tracks the harvested energy for enabling harvesting-aware power-management. It uses NiMH batteries for energy storage and provides a regulated constant voltage supply to the load. The design of the Heliomote is discussed in detail in Raghunathan et al. [2005] and the hardware designs are provided at Heliomote CVS [2005]. An image of the prototype with weather-resistant and water-proof packaging is shown in Figure 4. The higher accuracy of measurement in this method, indeed, comes at the price of having additional hardware support. However, the more accurate harvesting source and consumption models facilitated by this approach enable much better power-management.

4. POWER-MANAGEMENT ALGORITHMS

We design power-management algorithms for the case of an energy source that is uncontrollable but predictable. In this case, we can characterize it using the model defined in Section 2 or its refinements, and design an algorithm to attempt achieving energy-neutral operation. For an unpredictable source, i.e.,

one which cannot be modeled, guarantees on performance are hard to derive. The case of controlled source trivial as power can be generated as required.

Assume that the energy-generation profile may be characterized using a $(\rho_1, \sigma_1, \sigma_2)$-model. The first step for a harvesting system to ensure energy-neutral operation is to learn the characterization parameters so that, using Theorem 2.2, the sustainable performance level may be determined. The next step then is to adapt the performance level accordingly. Further, the performance scaling scheme may attempt to minimize energy wasted because of battery inefficiency and leakage if it can schedule the workload according to the temporal variations in energy generation. We present a performance-scaling algorithm to address the above steps.

We choose to use duty-cycling between active and low-power modes for the purpose of performance scaling, because most current low-power sensor nodes [Lymberopoulos and Savvides 2005; Motes 2005] provide at least one low-power mode in which the node is practically inactive and power consumption is negligible. More sophisticated hardware may provide multiple power-management options, which may be explored when available.

In battery-powered systems, the lowest tolerable duty cycle is typically chosen in order to extend the achievable lifetime to its maximum. In a harvesting system, our goal is to choose such a duty cycle such that ρ_2, as defined in the model for a load's energy profile, is set to its highest value allowed for energy-neutral operation. This will allow operating at the best possible performance, such as lowest achievable response time. The following two practical considerations however, cause a deviation from this highest value:

1. We do not require that the exact model parameters be available before deployment in any specific environment; rather, our algorithm learns these parameters at run time. To allow for inaccuracies and delays in learning, the node is allowed to operate in an energy-positive mode, so that it may store some energy. This allows operating in energy-negative mode for times when the actual energy harvested falls below the model parameters learned. The objective is to prevent the node from being completely shut down.
2. Energy buffers are not ideal and, hence, using the harvested energy directly rather than first storing it may help allow consuming a higher total energy. Thus, we may change the duty cycle in time rather than operating at a constant ρ_2 calculated theoretically.

In determining the correct strategy to adjust ρ_2, we also need a model for how the application performance is affected by it. To this end, we assume the following relationship between the provided duty cycle, D, and the perceived utility of the system to a user: suppose the utility of the application to the user is represented by $U(D)$ when the system operates at a duty cycle D. Then,

$$U(D) = 0, \quad \text{if} \quad D < D_{min} \tag{25}$$

$$U(D) = k_1 + k_2 D \quad \text{if} \quad D_{min} \leq D \leq D_{max} \tag{26}$$

$$U(D) = k_3 \quad \text{if} \quad D > D_{max} \tag{27}$$

This is a fairly general model and the specific values of D_{min} and D_{max} may be determined from the application requirements. For example, consider a sensor node designed to detect intruders crossing a periphery. The fastest and the slowest speeds of the intruders may be known, leading to a minimum and maximum sensing delay tolerable, and these result in the relevant D_{min} and D_{max} for the sensor node. As another example, consider a routing application where the sleep duration of the duty cycle directly increases the communication delay at each wireless hop. The maximum delay tolerable yields the value of D_{min}.

Our methods are designed in the context of Heliomote hardware, where the harvesting module is a solar cell and the energy buffer is a NiMH battery. The key consideration is that the storage is nonideal and, hence, apart from choosing a duty cycle, which is feasible for energy-neutral operation, we can enhance the performance if the energy generated by the harvesting source is used directly rather than stored in the battery first. For the above utility model, let us first consider the optimal power-usage strategy that is possible for a given energy-generation profile.

For the calculation of the optimal, we assume complete knowledge of the energy availability profile at the node, including the availability in the future. The calculation of the optimal is a useful tool for evaluating the performance of our proposed algorithm. This is particularly useful for our algorithm, since no prior algorithms are available to compare against in this area. Suppose the time axis is discretized into slots of duration ΔT and the duty cycle adaptation calculation is carried out over a window of N_w slots. Define the following discretized versions of the energy profile variables, with the index i ranging over $\{1, .., N_w\}$:

- $P_s(i)$, the power input from the harvested source in slot i. We assume this is constant over the slot duration. The slot duration may be chosen small enough for this assumption to be valid.
- P_c, the power consumption of the load, when in active mode. Most low-power systems have a sleep mode power consumption several orders of magnitude lower than the active mode and we approximate the sleep mode power consumption to zero.
- $D(i)$, the duty cycle used in slot i. This is a variable whose value is to be determined.
- $B(i)$, the residual battery energy at the beginning of slot i. Following this convention, the battery energy left after the last slot in the window is represented by $B(N_w + 1)$. The values of these variables will depend on the choice of $D(i)$.

The duty cycle determines the power consumption as $\rho_2 = D * P_c$ and, hence, adjusting D adjusts ρ_2; the relationship between utility and power consumption is graphically represented in Figure 5.

The performance objective, in view of the utility function discussed in the previous section, is: maximize the average throughput over the time window N_w, subject to a minimum duty cycle, D_{min}, desired in any slot. Also assume that the utility to the user is not increased if the duty cycle increases beyond a particular value D_{max}.

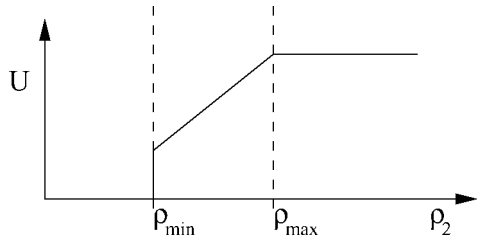


Fig. 5. Relationship between power consumption and application utility.

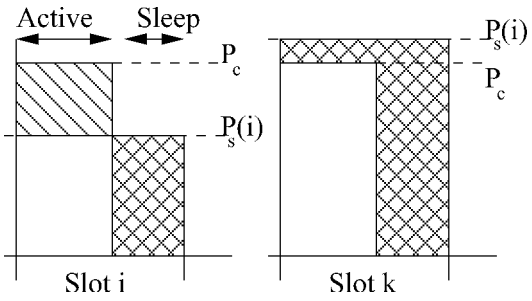


Fig. 6. Energy calculation for direct use and with storage.

We model the effect of storage inefficiency using the battery inefficiency parameter η[2]. The energy used directly from the harvested source and the energy stored and used from the battery may be computed as follows. Figure 6 shows two possible cases for $P_s(i)$ in a time slot—it may either be lower than or higher than P_c, as shown on the left and right, respectively. When $P_s(i)$ is lower than P_c, some of the energy used comes from the battery, while when $P_s(i)$ is higher than P_c, all the energy used is supplied directly from the harvested source. The cross-hatched area shows the energy that is available for storage into the battery, while the hashed area shows the energy drawn from the battery.

Again using the rectifier function $[\cdot]^+$, as defined in Section 2.1, we can write the energy used from the battery in any slot i as:

$$B(i) - B(i+1) = \Delta T D(i)[P_c - P_s(i)]^+ - \eta \Delta T P_s(i)\{1 - D(i)\} - \eta \Delta T D(i)[P_s(i) - P_c]^+$$

In the above equation, the first term on the right-hand side measures the energy drawn from the battery when $P_s(i) < P_c$. The next term measures the energy stored into the battery when the node is in sleep mode; the last term measures the energy stored in active mode if $P_s(i) > P_c$. For energy-neutral operation, we require the battery at the end of the window of N_w slots to be greater than or equal to the starting battery. Clearly, battery level will go down when the harvested energy is not available and the system is operated from stored energy. However, the window N_w is judiciously chosen such that over that duration, we expect the system to be energy neutral. For instance, in the

[2]For other storage technologies, such as ultracapacitors, leakage current may also be a significant factor, but is ignored in our analysis.

case of solar energy harvesting, N_w could be chosen to be a 24-hour duration, corresponding to the diurnal cycle in the harvested energy. This is an approximation, since an ideal choice of the window size would be infinite, but a finite size must be used for analytical tractability. The effect of this approximation is that our solution will behave conservatively for those days when energy input is lower than usual (such cloudy days), even if the battery was over sized enough to sustain that shortage and excess energy is likely to be available in the next window. Further, the battery level cannot be negative at any time.

Stating the above constraints quantitatively, we can express the calculation of the optimal duty cycles as an optimization problem:

$$max \sum_{i=1}^{N_w} D(i) \tag{28}$$

$$\begin{aligned} B(i) - B(i+1) &= \Delta T D(i)[P_c - P_s(i)]^+ - \eta \Delta T P_s(i)\{1 - D(i)\} \\ &\quad -\eta \Delta T D(i)[P_s(i) - P_c]^+ \quad \forall i \in \{1, \ldots, N_w\} \qquad (29) \\ B(1) &= B_0 \qquad (30) \\ B(N_w + 1) &\geq B_0 \\ D(i) &\geq D_{min} \quad \forall i \in \{1, \ldots, N_w\} \\ D(i) &\leq D_{max} \quad \forall i \in \{1, \ldots, N_w\} \qquad (31) \end{aligned}$$

where B_0 is the starting residual battery energy.

The solution to the above optimization problem yields the duty cycles, which must be used in every slot and the evolution of residual battery over the course of N_w slots. Note that while the constraints above contain the nonlinear function $[x]^+$, the quantities occurring within that function are all known constants. The variable quantities occur only in linear terms and, hence, the above optimization problem can be solved using standard linear programming techniques, available in popular optimization toolboxes.

The above optimal will be used as a benchmark. For a practical implementation, we develop an algorithm that attempts to achieve energy-neutral operation without using knowledge of the future energy availability and maximizes the achievable performance within that constraint.

The harvesting-aware power-management strategy consists of three parts. The first part is an instantiation of the energy generation model, which tracks past energy input profiles and uses them to predict future energy availability. The second part computes the optimal duty cycles based on the predicted energy. This step does not use standard linear programming tools, which may be computationally complex for many of the resource constrained low-power sensor nodes but, our computationally tractable method to compute the same solution. The third part consists of a method to dynamically adapt the duty cycle in response to the observed energy generation profile in real time. This step is required since the observed energy generation may deviate significantly from the predicted energy availability and energy-neutral operation must be ensured with the actual energy received rather than the predicted values.

4.1 Energy Prediction Model

We use a prediction model based on an exponentially weighted moving-average (EWMA) filter [Cox 1961]. The method is designed to exploit the diurnal cycle in solar energy, but, at the same time, adapt to the seasonal variations. A historical summary of the energy generation profile is maintained for this purpose. While the storage data size is limited to a vector length of N_w values in order to minimize the memory overheads of the power-management algorithm, the window size is effectively infinite as each value in the history window depends on all the observed data up to that instant. A window size duration is chosen to be 24 hours and each slot is taken to be 30 minutes, as the variation in generated power level is assumed to be small within a 30-minute duration. This yields $N_w = 48$. Smaller slot durations may be used at the expense of a higher N_w. The historical summary maintained is derived as follows. On a typical day, we expect the energy generation to be similar to the energy generation at the same time on the previous days. The value of energy generated in a particular slot is maintained as a weighted average of the energy received in the time slot at that time of the day during all observed days. The weights are exponential, resulting in decaying weights for older data. Let $x(i)$ denote the value of energy generated in slot i as observed at the end of that slot. The historical average maintained for each slot is given by:

$$\bar{x}(i) = \alpha\bar{x}(i-1) + (1-\alpha)x(i) \tag{32}$$

where α is a weighting factor, and $\bar{x}(i)$ is the historical average value maintained for slot i. Substituting $\bar{x}(i-1)$ using a similar equation gives:

$$\bar{x}(i) = \alpha^2\bar{x}(i-2)\alpha(1-\alpha)x(i-1) + (1-\alpha)x(i) \tag{33}$$

If we similarly expand $\bar{x}(i-2)$, and so on, it may be noted that older values of $x(i)$ are weighted by increasing powers of α. Since α is less than 1, the contribution of older values of $x(i)$ becomes progressively smaller. This is referred to as an EWMA filter. In this model, the importance of each day relative to the previous one remains constant because the same weighting factor was used for all days. The average value derived for a slot is treated as an estimate of predicted energy value for the slot corresponding to the same slot of the previous day. This method helps the historical average values adapt to the seasonal variations in energy received on different days.

One of the parameters to be chosen in the above prediction method is the parameter α. To determine a good value for this parameter, we collected energy data over several days and compared the performance of the prediction method for various values of this parameter. The prediction error based on the different values of α is shown in Figure 7. This curve suggests an optimum value of $\alpha = 0.5$ for minimum prediction error and this value will be used in the remainder of this paper. Note that instead of choosing α *a priori*, a dynamic approach that estimates α in real time may also be employed. One method to adapt α, based on the observed error performance of the prediction in previous slots, was provided in Kansal and Karandikar [2001].

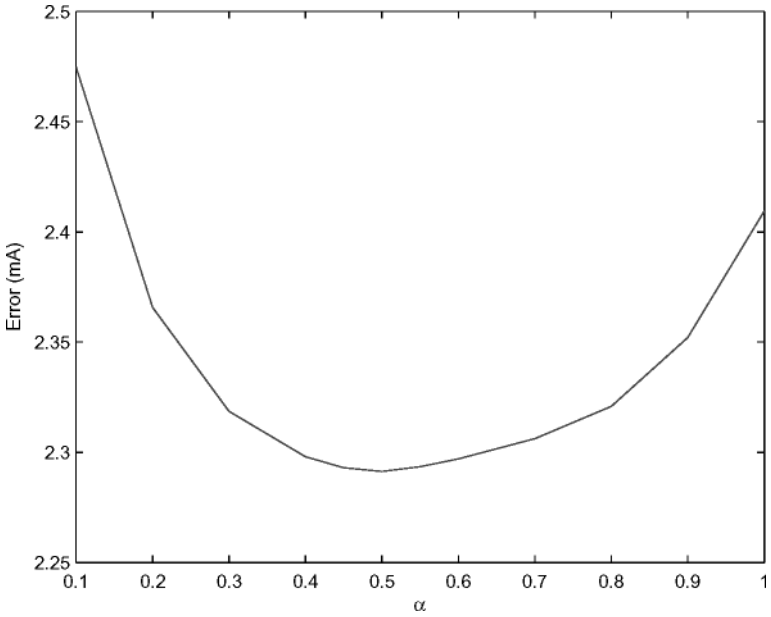


Fig. 7. Choice of parameter α through error evaluation.

4.2 Low-Complexity Solution to the Optimization Problem

The energy values predicted for the next window of N_w slots are used to calculate the desired duty cycles for this window, assuming the predicted values match the observed values. Since, our objective is to develop a practical algorithm for embedded computing systems, we present a simplified method to solve the linear programming problem of Eq. (28).

To this end, we define the sets $\mathcal{S}$ and $\mathcal{D}$ as follows:

$$\mathcal{S} = \{i | P_s(i) - P_c \geq 0\} \tag{34}$$

$$\mathcal{D} = \{i | P_s(i) - P_c < 0\} \tag{35}$$

In the following text, we will refer to $\mathcal{S}$ as sun slots and $\mathcal{D}$ as dark slots. Next we sum up both sides of Eq. (29) over the entire N_w window and write it using the new notation:

$$\sum_{i=1}^{N_w} B(i) - B(i+1) = \sum_{i \in \mathcal{D}} \Delta T D(i)(P_c - P_s(i)) - \sum_{i=1}^{N_w} \eta \Delta T P_s(i) + \sum_{i=1}^{N_w} \eta \Delta T P_s(i) D(i) - \sum_{i \in \mathcal{S}} \eta \Delta T D(i)(P_s(i) - P_c) \tag{36}$$

Noting that the term on the left-hand side is:

$$\sum_{i=1}^{N_w} B(i) - B(i+1) = B(1) - B(N_w + 1) \tag{37}$$

which is the battery energy used over the entire window of N_w slots. This can be set to zero for energy-neutral operation. Equating the right-hand side of Eq. (36) to 0 and performing some algebraic manipulations yields:

$$\sum_{i=1}^{N_w} P_s(i) = \sum_{i\in\mathcal{D}} D(i)\left[\frac{P_c}{\eta} + P_s(i)\left(1-\frac{1}{\eta}\right)\right] + \sum_{i\in\mathcal{S}} P_c D(i) \tag{38}$$

The term on the left-hand side is the total energy received in N_w slots. The first term on the right-hand side can be interpreted as the total energy consumed during the dark slots and the second term is the total energy consumed during the sun slots. We can now replace three constraints (29), (30), and (31) in the original optimization problem (28) with Eq. (38), restating the optimization problem as follows:

$$max \sum_{i=1}^{N_w} D(i) \tag{39}$$

$$\begin{aligned}
\sum_{i=1}^{N_w} P_s(i) &= \sum_{i\in\mathcal{D}} D(i)\left[\frac{P_c}{\eta} + P_s(i)\left(1-\frac{1}{\eta}\right)\right] + \sum_{i\in\mathcal{S}} P_c D(i) \\
D(i) &\geq D_{min} \quad \forall i \in \{1,\ldots,N_w\} \\
D(i) &\leq D_{max} \quad \forall i \in \{1,\ldots,N_w\}
\end{aligned}$$

This form facilitates a low complexity solution that does not require a general linear programming solution. We first notice that in Eq. (38) the coefficients for $\{D(i)|i \in \mathcal{S}\}$ are strictly smaller than $\{D(i)|i \in \mathcal{D}\}$ because $P_c < [\frac{P_c}{\eta} + P_s(i)(1 - \frac{1}{\eta})]\forall i$, as $\eta < 1$. Thus, to maximize the summation, we should use energy in such a way as to maximize $\{D(i)|i \in \mathcal{S}\}$ by using the minimum allowed values for $\{D(i)|i \in \mathcal{D}\}$. Given the D_{max} and D_{min} constraints, initialize the duty cycle assignments as:

$$D(i) = D_{min} \quad \forall \quad i \in \mathcal{D} \tag{40}$$

$$D(i) = D_{max} \quad \forall \quad i \in \mathcal{S} \tag{41}$$

It is very likely that the above initial assignment is not optimal or not even feasible, and we will make the necessary adjustments in the following steps. There are two cases:

- *Case I. Energy is Underallocated.* We have underallocation in the initial assignment, that is:

$$\sum_{i\in\mathcal{D}} D(i)\left[\frac{P_c}{\eta} + P_s(i)\left(1-\frac{1}{\eta}\right)\right] + \sum_{i\in\mathcal{S}} P_c D(i) < \sum_{i=1}^{N_w} P_s(i)$$

  Thus, there is excess energy available which is not being used and this may be allocated to increase the duty cycle in dark slots since the sun slots, are already saturated. The most efficient way to allocate the excess energy is to assign duty cycle D_{max} to the slot with the smallest $D(i)$ coefficients among the dark slots. Thus, the coefficients $[\frac{P_c}{\eta} + P_s(i)(1-\frac{1}{\eta})]$ are arranged in increasing

order and duty cycle D_{max} is assigned to the slots, beginning with the smallest coefficients until the excess energy available, R, (given by the difference of the left- and right-hand sides of Eq. (42)) is insufficient to assign D_{max} to another slot. The remaining energy, R_{last}, is used to increase the duty cycle to a value between D_{min} and D_{max} in the dark slot with the next higher coefficient. Denoting this slot with index j, the duty cycle is given by:

$$D(j) = \frac{R_{last}}{(P_s(j) - P_c)/\eta - P_s(j)} + D_{min}$$

If there is excess energy after allocating D_{max} to all slots, the system can perform higher than has utility for the user and this energy may be arbitrarily allocated.

- *Case II. Energy is Overallocated.* The duty cycles assigned require more energy than is actually available and the energy deficiency, L, is given by:

$$L = \sum_{i \in \mathcal{D}} D(i) \left[\frac{P_c}{\eta} + P_s(i) \left(1 - \frac{1}{\eta} \right) \right] + \sum_{i \in \mathcal{S}} P_c D(i) - \sum_{i=1}^{N_w} P_s(i)$$

Since the duty cycles in the dark slots cannot be reduced below D_{min}, we have to reduce the duty cycles in the sun slots. In order to bring down L to zero, we may reduce the duty cycles during sun slots, uniformly by δ, given as:

$$\begin{aligned} \delta |\mathcal{S}| P_c &= L \\ \delta &= \frac{L}{|\mathcal{S}| P_c} \end{aligned}$$

where $|\mathcal{X}|$ is the set cardinality operator. The duty cycles in the sun slots thus become $D_{max} - \delta$. Note that reducing them by unequal amounts does not yield any advantage in the total throughput supported. Also, if $D_{max} - \delta < D_{min}$, we conclude that the optimization problem is infeasible, implying that energy-neutral operation is not allowed with the given energy and performance requirements.

This solution to the optimization problem requires only simple arithmetic calculations and one sorting step, which can be easily implemented on an embedded platform, as opposed to implementing a general linear program solver.

While our simplified solution saves on the computational overhead required, it is as accurate as the general LP solution. The savings in complexity come from exploiting the specific problem structure.

4.3 Dynamic Duty Cycle Adaptation

The observed energy values may vary greatly from the predicted ones, such as due to the effect of clouds or other sudden changes. It is thus important to adapt the duty cycles calculated using the predicted values, to the actual energy measurements in real time to ensure energy-neutrality.

Denoted the initial duty cycle assignments for each time slot i computed using the predicted energy values as $D(i) = \{1, \ldots, N_w\}$. First we compute the difference between predicted power level $P_s(i)$ and actual power level observed,

$P'_s(i)$ in every slot i. Then, the excess energy in slot i, denoted by X, can be obtained as follows:

$$X = \begin{cases} P_s(i) - P'_s(i) & \text{if} \quad P'_s(i) > P_c \\ P_s(i) - P'_s(i) - D(i)[P_s(i) - P'_s(i)]\left(1 - \frac{1}{\eta}\right) & \text{if} \quad P'_s(i) \leq P_c \end{cases}$$

The above equations follow directly from equation Eq. (38). The first term accounts for the energy difference when actual received power level is more than the power drawn by the load. On the other hand, if the power received is less than P_c, we will need to account for the extra energy used from the battery by the load, which is a function of duty cycle time slot i and battery inefficiency factor η. In order to adjust for the change of total energy and maintain energy-neutral, duty cycles in the future time slots will need to be reduced or increased to account for the error we made in the current time slot. When more energy is received than predicted, then X is positive and that excess energy is available for use in in subsequent slots, while if X is negative, that energy must be compensated from subsequent slots. We consider both cases below:

- *Case* I. $X < 0$. In this case, we want to reduce the duty cycle in the future slots in order to make up for the shortfall in energy. Since our object function is to maximize the total throughput, or duty cycle, we have to reduce the duty cycle for time slots with larger energy costs and these are the slots with lowest energy availability. This is accomplished by first sorting the predicted energy profile P_j where $j > i$. in decreasing order, and then iteratively reduce $D(i)$ to D_{min} until the total reduction in energy consumption is the same as X.
- *Case* II. $X > 0$. Here, we want to increase the duty cycles used in the future to utilize the excess energy received in recent time slot. The duty cycles of future time slots with lowest energy cost should be increased first in order to maximize the total throughput.

Suppose the duty cycle is changed by δ in slot j. Define a quantity $R(j, \delta)$ as follows:

$$R(j, \delta) = \begin{cases} P_c\delta & \text{if} \quad P_s(j) > P_c \\ \delta\left[\frac{P_c}{\eta} + P_s(j)\left(1 - \frac{1}{\eta}\right)\right] & \text{if} \quad P_s(j) \leq P_c \end{cases}$$

This quantity is used to account for power usage as the duty cycle is changed.

The precise procedure to adapt the duty cycle to account for the above factors is presented in Algorithm 1. This calculation is performed at the end of every slot to set the duty cycle for the next slot. The duty cycle change may not be equal in all slots, such as when the energy available for the last slot, in which the duty cycle is being increased, is not sufficient to increase the duty cycle as much as in the other slots.

We claim that our duty cycling algorithm is energy-neutral because a surplus of energy at the previous time slot will always translated to additional energy opportunity for future time slots, and vice versa. The claim may be violated in cases of severe energy shortages when the environmental supply is below predicted values for a duration long enough to exhaust the battery reserve completely. Such an error in learning may occur when the environment behaves

Algorithm 1: Algorithm for duty cycle adaptation.
Input: D: Initial duty cycle, X: Excess energy because of difference from prediction, P_s: Predicted energy profile, i: index of current time slot
Output: D: Updated duty cycles in one or more subsequent slots
ADAPTDUTYCYCLE()

(1) **Iteration:** At each time slot do:
(2) **if** $X > 0$
(3) $P_{sorted} = P_s\{1, \ldots, N_w\}$ sorted in ascending order.
(4) $Q :=$ indices of P_{sorted}.
(5) **for** $k = 1$ **to** $|Q|$
(6) **if** $Q(k) \leq i$ //slot is in the past
(7) continue
(8) **if** $R(Q(k), D_{max} - D(Q(k))) < X$
(9) $D(Q(k)) = D_{max}$
(10) $X = X - R(j, D_{max} - D(Q(k)))$
(11) **else**
(12) //X is insufficient to increase duty cycle to D_{max}
(13) **if** $P_s(Q(k)) > P_c$
(14) $D(Q(k)) = D(Q(k)) + X/P_c$
(15) **else**
(16) $D(Q(k)) = D(Q(k)) + \frac{X}{(P_c/\eta + P_s(Q(k))(1-1/\eta))}$
(17) **if** $X < 0$
(18) $P_{sorted} = P_s\{1, \ldots, N_w\}$ sorted in descending order.
(19) $Q :=$ indices of P_{sorted}.
(20) **for** $k = 1$ **to** $|Q|$
(21) **if** $Q(k) \leq i$ **or** $D(Q(k)) \leq D_{min}$
(22) continue
(23) **if** $R(Q(k), D_{min} - D(Q(k))) > X$
(24) $D(Q(k)) = D_{min}$
(25) $X = X - R(j, D_{min} - D(Q(k)))$
(26) **else**
(27) **if** $P_s(Q(k)) > P_c$
(28) $D(Q(k)) = D(Q(k)) + X/P_c$
(29) **else**
(30) $D(Q(k)) = D(Q(k)) + \frac{X}{(P_c/\eta + P_s(Q(k))(1-1/\eta))}$

completely differently from its historical behavior, since the learning is based on the history only. Using a larger battery than computed optimal for the learned parameters helps provide a safeguard against such errors by providing a larger tolerance in model parameters.

Note that the dynamic duty cycle adaptation method to make up for prediction errors should not be considered a substitute for good prediction methods. When there is an error in prediction, the dynamic adaptation helps ensure energy-neutrality but the system does deviate from the optimal duty cycle allocation, suffering a potential performance loss. When the prediction is higher than the actual energy received by a difference ΔE, we have to make up for the excess energy drawn from the battery by returning $\Delta E/\eta$ in some subsequent slot, hence, wasting the energy $\Delta E(1-\eta)$, which could ideally have been used directly in that subsequent slot. Similarly, if the predicted energy is lower by ΔE, we store it rather than using it and are able to recover only $\eta \Delta E$ in some subsequent slot, thus, again wasting energy $\Delta E(1-\eta)$.

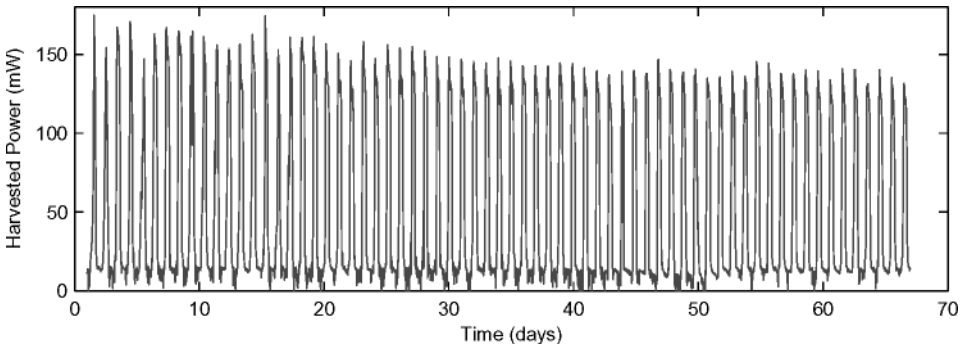


Fig. 8. Harvested energy measured over several days using Heliomote energy-harvesting sensor node.

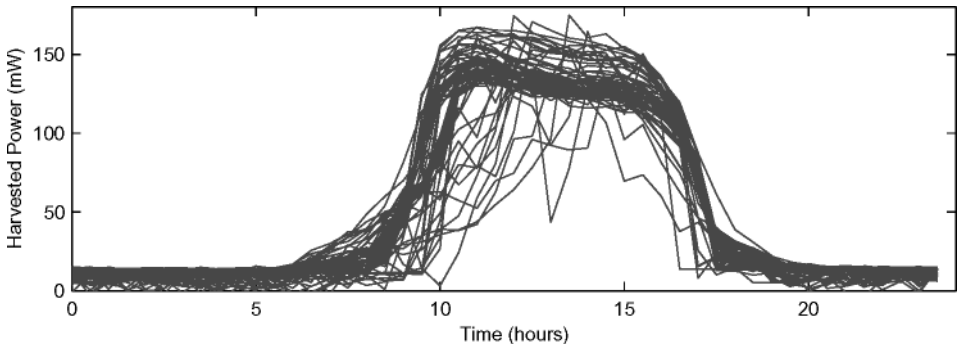


Fig. 9. Harvested energy profile on a diurnal scale.

4.4 Evaluation

The methods proposed above were evaluated using an actual solar energy profile measured using our Heliomote platform. This platform not only tracks the generated energy, but also the energy flow into and out of the battery to provide an accurate estimate of the storage level. The Heliomote was deployed in a residential area in Los Angeles from June 2005 to August 2005, for a total of 67 days. The sensor node used is a Mica2 mote running at a fixed 40% duty cycle with an initially full battery. Battery voltage and net current from the solar panels are sampled at a period of 10 s. The energy generation profile for that duration, measured by tracking the output current and voltage from the solar cell, is shown in Figure 8. The same energy availability profile is plotted on a diurnal scale in Figure 9, with the data from all 67 days overlapped.

Using the above energy profile, we first evaluate the performance of the prediction model. Figure 10 shows the average error in each time slot averaged over that slot on all 67 days. The amount of error is larger during the daytime because that is when factors, such as weather, can cause deviations in received energy, while the prediction made for nighttime is mostly correct. Note that the maximum prediction error during daytime is 20 mW, which is tolerable for the energy generation level close to 150 mW (Figure 9) during daytime.

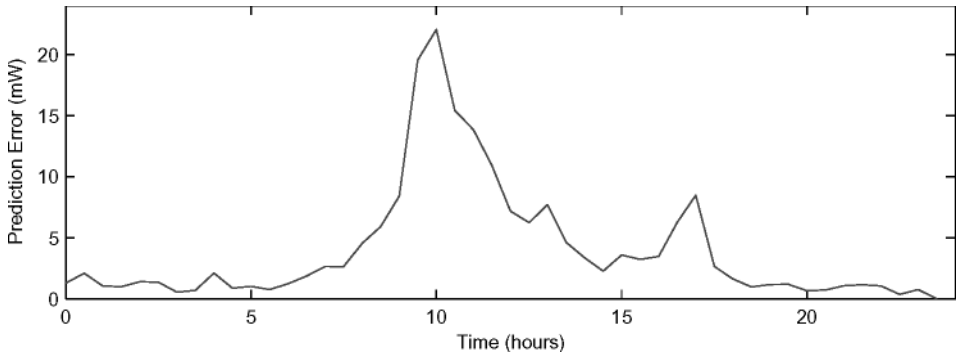


Fig. 10. Error performance of the EWMA based prediction method.

Next, we evaluate the proposed duty cycling algorithm. Prior methods that optimize performance with energy-neutral operation were not available to compare against. Instead, we compare the performance of our algorithm against two extremes: the theoretical optimum calculated using future energy availability and a naïve approach, which attempts to achieve energy-neutrality using a fixed duty cycle without accounting for battery inefficiency. The optimal duty cycles are calculated for each slot using the future knowledge of actual received energy for that slot. For the naïve approach, the duty cycle is kept constant within each day and is computed by taking the ratio of the predicted energy availability and the maximum usage:

$$D(\cdot) = \frac{\eta \sum_{i=1}^{N_w} P_s(i)}{N_w P_c}$$

We compare the performance of our dynamic duty cycle adaptation algorithm to the two extremes with varying battery efficiency. Figure 11 shows the results, using $D_{max} = 0.7$ and $D_{min} = 0.2$. The battery efficiency was varied from 0.5 to 1 and the average duty cycles achieved by the three algorithms are shown on the y-axis.

In addition, we also compare the performance of our algorithm with different values of D_{min} and D_{max} for $\eta = 0.7$, which is typical of NiMH batteries. These results are shown in Table II. The average duty cycles are shown for our proposed dynamic approach and the naïve approach normalized by the optimal duty cycle achievable. The figures and table indicate that our real-time algorithm is able to achieve a performance very close to the optimal feasible. In addition, these results reiterate the importance of harvesting by showing that environmental energy-harvesting with appropriate power-management can achieve much higher duty cycles than those currently used, such as 1% used in battery-based deployments [Mainwaring et al. 2002].

5. HARVESTING NETWORKS

We now consider a distributed network in which some or all the nodes have a harvesting opportunity. Note that even when all the nodes are homogeneously equipped with the same harvesting hardware, the available environmental

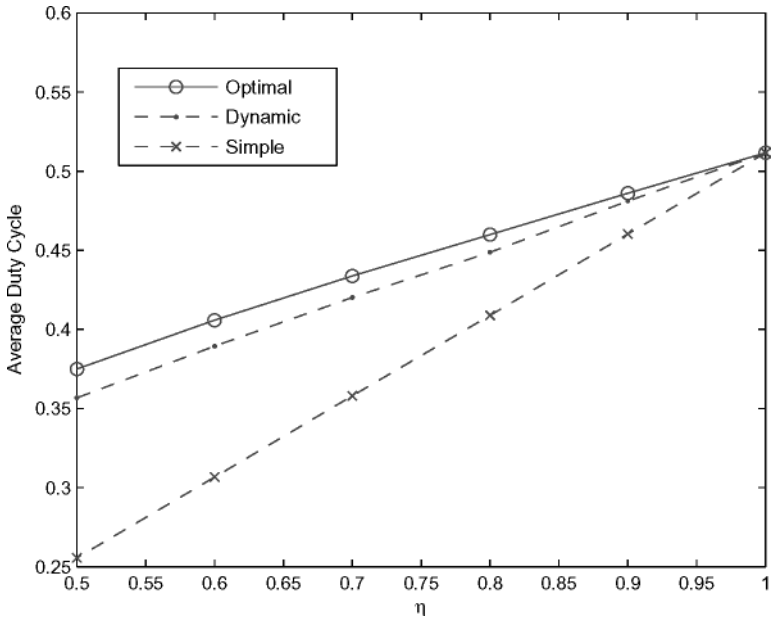


Fig. 11. Performance comparison with varying battery efficiency.

Table II. Performance Comparison with Different Utility Parameters

D_{min}	0.2	0.4	0.05
D_{max}	0.7	0.7	0.7
Dynamic duty cycling	97.3%	96.2%	97.8%
Naïve duty cycling	82.5%	86.2%	82.4%

energy at each node location may not be the same. One of the questions of interest is to determine the performance potential of a given energy environment.

The performance potential from an energy environment depends on the spatiotemporal variation in the energy availability across the network. While the amount of energy available is definitely relevant, the distribution of this energy in space and time significantly affects the network performance. For instance, if large amounts of energy are available, but concentrated only in a small region of the network, the nodes in regions without energy supply will limit the total useful lifetime of the network, beyond which any available energy in other regions may not be able to meet the performance requirements from the system as a whole.

Clearly, each node in the network must achieve energy-neutral (or energy positive) operation for the entire network to be energy neutral. The abstractions developed earlier continue to guide the energy usage at each node. However, performance can be maximized if workload allocation among these nodes is adjusted, depending on the available energy profiles. The performance itself is characterized using application-specific metrics and the characteristics of flexibility in workload allocation are also strongly tied to the application. Below, we consider two example applications, which characterize the two extremes of

the spectrum of sensor network applications: measuring a field phenomenon where the entire field is to be mapped and measuring point events where only sporadic events are reported. Most applications may lie in between the two extremes, for instance, collecting more data than just the point events that occur, but aggregating it to something less than mapping the entire field.

5.1 Example 1: Field Monitoring

Consider an application where a sensor network is deployed for monitoring a spatially distributed field phenomenon and periodic samples are to be collected at each sensor. These are then routed to a base station. The performance metric of interest in this application is the maximum rate at which the field may be sampled.

Assuming uniform sampling across nodes, an equal amount of data R_g is generated at each sensor in unit time. Suppose the network has N nodes, labeled $i = \{1, \ldots, N\}$. Take node 1 to be a base station, which collects all the data and is not energy constrained. The remaining nodes are energy-harvesting sensor nodes.

Assume the following energy consumption model for the sensor node activities, as taken from Heinzelman [2000]. The energy consumption $P_{tx}(i, j)$ at the transmitter i when communicating with a node j, at data rate r is given as:

$$P_{tx}(i, j) = r\left[\alpha_1 + \alpha_2 d(i, j)^2\right] \tag{42}$$

where $d(i, j)$ is the distance between the transmitter and the receiver and α_1, α_2 are radio-dependent constants. The first term models a constant consumption in the radio electronics and the second term models the distance-dependent transmission cost. Suppose the reception energy is P_{rx} at data rate r, and the energy cost of sampling the transducers is P_{sense}. Let the energy-neutral power consumption level supported at node i be denoted $\rho_2(i)$.

To evaluate the performance metric, we need to compute the cost of each route from the selected sensors. However, since the number of routes can be exponential in the number of nodes, it has been found to be more tractable to take an equivalent view of the routes in terms of data flows across each link in the network [Bhardwaj and Chandrakasan 2002; Giridhar and Kumar 2005; Chang and Tassiulas 2000]. Denote the amount of data flowing on a link from node i to j as $f(i, j)$. These flows must be calculated so as to maximize the amount of data, R_g, that can be sensed and routed from every sensor to the base station. We write a linear program to maximize R_g, as follows.

$$\max\{R_g\} \tag{43}$$

subject to:

$$f(i, j) \geq 0 \quad \forall \quad i, j \in \{1, \ldots, N\} \tag{44}$$

$$-\sum_{\substack{s \in \{1, N\} \\ s \neq i}} f(s, i) + \sum_{\substack{d \in \{1, N\} \\ d \neq i}} f(i, d) = R_g \quad \forall \quad i \in \{2, N\} \tag{45}$$

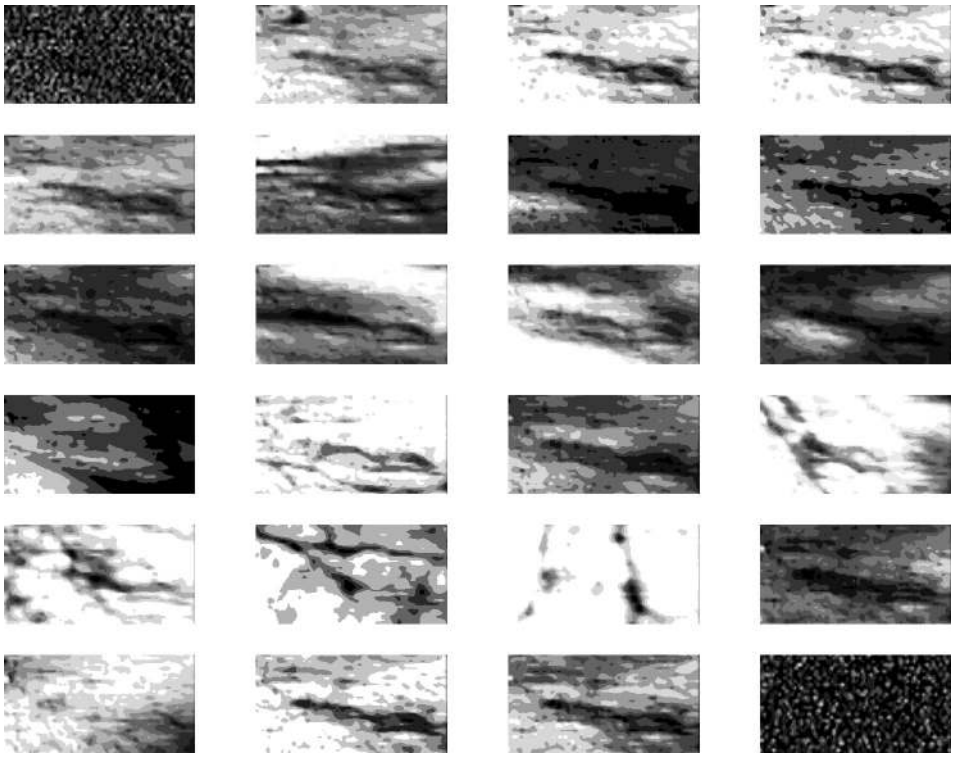


Fig. 12. Spatiotemporal variation in solar energy distribution in an outdoor environment. Lighter shades represent higher light intensity.

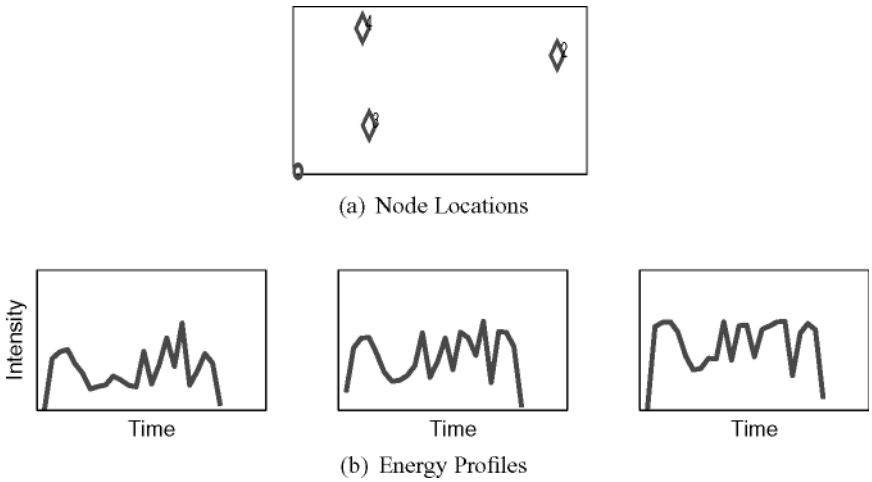
$$\sum_{\substack{d \in \{1,N\} \\ d \neq i}} P_{tx}(i,d) f(i,d) + \sum_{\substack{s \in \{1,N\} \\ s \neq i}} P_{rx} f(s,i) + P_{sense} R_g \leq \rho_2(i) \quad (46)$$

$$\forall \quad i \in \{2,N\}$$

The constraint (44) follows from the fact that the flows $f(i,j)$ for $i,j \in \{1,\ldots,N\}$ must be nonnegative. Constraint (45) states that the total flows must be conserved. Also, energy-neutrality must be achieved, as stated in Eq. (46). We use an inequality sign instead of equality in Eq. (46) because the performance may happen to be constrained by some low-energy nodes and no workload allocation strategy may be able to fully utilize all the energy at some of the energy-rich nodes.

The above linear program is similar in flavor to those presented in Bhardwaj and Chandrakasan [2002], Giridhar and Kumar [2005], and Chang and Tassiulas [2000], for lifetime calculations. Note, however, that we are not maximizing the lifetime, but optimizing the network performance while operating in an energy-neutral mode.

To demonstrate the above calculation for one instance of a solar energy-harvesting network, consider the solar intensity data in Figure 12. Each plot in Figure 12 shows the distribution of energy across the spatial extent of a small



(a) Node Locations

(b) Energy Profiles

Fig. 13. Energy profiles at three randomly chosen locations.

Table III. Harvested Power Levels at Random Locations

Node	ρ_2 (mW)
1	9.8799
2	13.3529
3	16.4718

region in James Reserve [James Reserve]. The multiple plots each show the same spatial region but at different times of day (7:30 AM to 7 PM in winter, at one-half hour interval each). The solar intensity is affected by time of day and movement of tree shadows in the region. The data is collected using a camera and since the ground surface in field of view was a single color, the intensity reflects the relative light-energy availability at each point.

Consider a network with four nodes, where the first node is a base station and the remaining three are energy-harvesting sensor nodes. Node 1 is located at the origin. The three sensor nodes are placed at randomly generated locations shown in Figure 13a in the same energy environment as shown in Figure 12. The temporal energy profile at these locations is shown in Figure 13b for nodes 2,3, and 4.

The light intensity is scaled to a harvested power level by using experimental data from a Heliomote that indicates that maximum intensity in the above environment corresponds to a power level of 90 mW. Ignoring battery inefficiency and leakage, the resultant energy-neutral power consumption levels are listed in Table III for the three nodes.

In addition to these, we use the hardware energy parameters from Heinzelman [2000]: $\alpha_1 = 45 \times 10^{-9} J/b, \alpha_2 = 10 \times 10^{-12} J/bm, P_{rx} = 0.135$ mW, and $P_s = 0.05$ mW, at $r = 1$ Mbps. Substituting these values in the linear program of Eq. (43), we obtain the solution: each sensor can sense at

$R_g = 75.8$ bps. The optimal routes for the random topology seen above are:

$$\begin{aligned} Node_2 &\rightarrow Base : 1.47bps \\ &\rightarrow Node_3 : 30.58bps \\ &\rightarrow Node_4 : 43.74bps \end{aligned}$$

$$\begin{aligned} Node_3 &\rightarrow Base : 75.8bps \\ &\rightarrow Base : 30.58bps \quad \text{Relay data from } Node_2 \end{aligned}$$

$$\begin{aligned} Node_4 &\rightarrow Base : 75.8bps \\ &\rightarrow Base : 43.74bps \quad \text{Relay data from } Node_2 \end{aligned} \tag{47}$$

A suitable packet transmission schedule that achieves the above routes and traffic allocations is thus required to achieve the maximum performance from the harvesting network for this application. While we illustrated the solution using only three nodes, note that the linear program developed above can be solved for a much larger N as well, since the number of variables and constraints is only polynomial in N.

5.2 Example 2: Event Monitoring

Consider a second example where a sensor network is deployed to monitor a field. However, instead of routing all samples, only some special events are transmitted. The network must, however, be prepared to report an event as soon as it is detected. Suppose the sensor nodes are duty cycled to achieve energy-neutrality. Each node enters active mode with a frequency, depending on its duty cycle, to listen for any packets to be relayed. There is a delay in data transmission since each transmission must wait for the next hop node along the route to enter active mode. The performance metric of interest in this application is the latency of data transfer from a sensor node to the base station.

We assume that most of the energy is consumed in listening for possible relay requirements and the events themselves are rare enough so that the routing energy is ignored. Suppose a node must listen for a minimum carrier sense time of T_{cs} to detect if some other node in radio range is trying to transmit. Further suppose that the wake-up protocol used is as follows. Each node not currently involved in communication enters a sleep mode for duration T_{sleep} and wakes up for a duration $3T_{cs}$ to listen for potential transmissions. Each node that wishes to transmit a packet will transmit a wake-up beacon for duration T_{cs}, wait for a response for another duration of T_{cs}, and repeat this process until a response is received. Note that the transmit and receive nodes may not be synchronized and the receiver node is required to listen for $3T_{cs}$ to ensure that the transmission of at least wake-up beacon falls within the listen duration (Figure 14). After sending the response to the wake-up beacon, the receiver is ready to receive the packet and the transmitter can immediately send it. Assuming that the event transmission may begin uniformly randomly

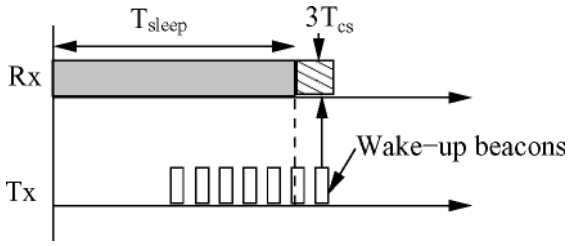


Fig. 14. MAC layer wake-up protocol for communicaion with sleep mode usage.

at any point in time, the average hop delay, D_{hop}, is:

$$D_{hop} = \frac{T_{sleep} + 3T_{cs}}{2}$$

Suppose a route has K hops and the sleep duration at the jth receiver node along this route is $T_{sleep}(j)$. The then, route delay, D_{route}, is:

$$D_{route} = \sum_{j=1}^{K} \frac{T_{sleep}(j) + 3T_{cs}}{2} \tag{48}$$

In the sleep mode, the energy intensive processing and communication subsystems are turned off and only a very low-power analog transducer for event detection may be kept active. The power consumption of the node in this case, P_{avg}, is given by:

$$P_{avg} = \frac{3T_{cs}}{3T_{cs} + T_{sleep}} P_{active}$$

where P_{active} is the power consumption in listen mode.

Here, taking a harvesting-aware approach helps in two ways:

- 1. *MAC Delay.* If each node is harvesting-aware, it may choose its duty cycle based on the maximum allowed ρ_2 at this node. Thus,

$$T_{sleep} = \frac{P_{active} - \rho_2}{\rho_2}(3T_{cs}) \tag{49}$$

  Nodes with better harvesting opportunity can use a lower T_{sleep}. If the network is not harvesting-aware, a conservative sleep duration corresponding to the minimum expected energy-harvesting opportunity at each node would have to be used. If the network has N nodes, then:

$$\rho_{common} = \min_{j \in \{1,..,N\}} \rho_2(j)$$

$$T_{sleep} = \frac{P_{active} - \rho_{common}}{\rho_{common}}(3T_{cs})$$

  Substituting this in Eq. (48), it is easy to see that the total MAC layer delay along the route will be more when harvesting-awareness is not available.
- 2. *Route Delay.* When the MAC layer is harvesting-aware and allows each node to use a different sleep duration, then a harvesting-aware routing protocol can choose the routes, which minimize the total route delay rather than

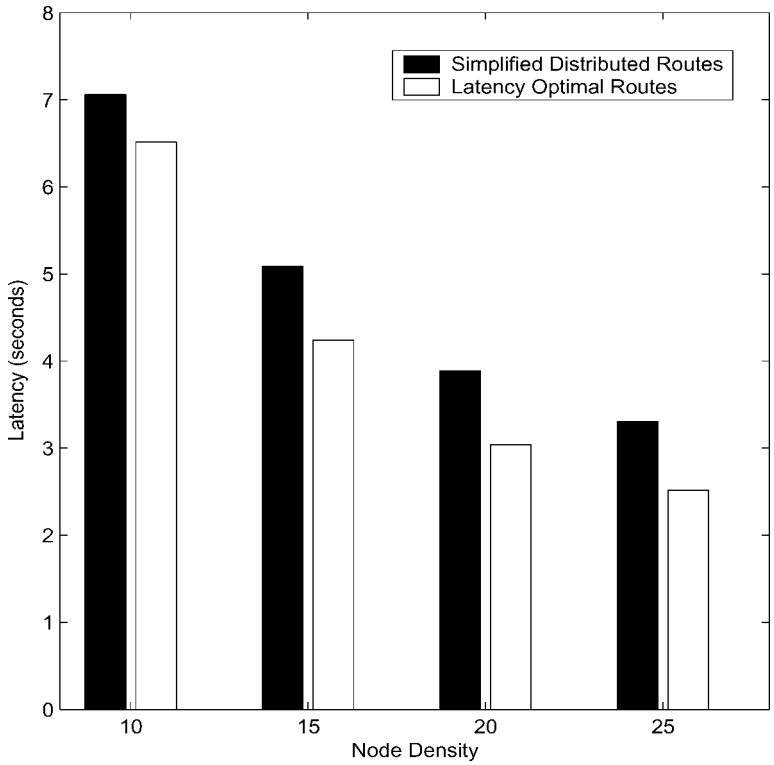


Fig. 15. Maximum path latency observed for simplified distributed routing and for optimal minimum latency routing.

minimizing the number of hops. A distributed routing protocol to allow each node to learn its local ρ_2 and then discover near-minimum delay routes from all nodes to a central base station was presented in Kansal et al. [2004]. A network with 100 nodes was simulated, where each node has a different energy opportunity, randomly generated by perturbing the typical energy collected as measured using our Heliomote hardware. Figure 15 from Kansal et al. [2004] presents the latency performance of distributed routing protocol as compared to an optimal one. The optimal routing protocol calculates the minimum delay routes using the complete delay information. It may be seen that the distributed version performs close to the optimal.

6. RELATED WORK

There is significant interest in energy harvesting for many different types of systems for improving their sustainable lifetimes, such as for wearable computers [Kymisis et al. 1998; Shenck and Paradiso 2001; Starner 1996] and sensor networks [Rahimi et al. 2003]. Several technologies to extract energy from the environment have been demonstrated including solar, motion-based, biochemical, and vibrational energies [Wright et al. 2000; Ieropoulos 2003; M. Rabaey et al. 2000; Paradiso and Feldmeier 2001; Meninger et al. 1999], and many

more are being developed (such as DARPA; Weber [2003]). A method to replenish the energy resources from non-environmental sources was given in Rahimi et al. [2003]. These technologies can provide systems at varying scales with the ability to extract energy from the environment. Solar energy-harvesting sensor node prototypes have also been developed (such as Park et al. [2005], Jiang et al. [2005], and Raghunathan et al. [2005]).

However, there is a need to exploit the available energy in such a way that energy efficiency is maximized and performance guarantees can be provided, which is not addressed by the above projects. We are providing methods to systematically utilize environmental energy resources in a performance-aware manner. The problem we solve is of immediate benefit to all the above research efforts.

Energy efficiency is a major concern in wireless sensor networks [Raghunathan et al. 2002; Min et al. 2000a]. Energy-aware methods to take tasking decisions have been considered before for routing [Singh et al. 1998; Younis et al. 2002; Shah and Rabaey 2002; Li et al. 2001; Chang and Tassiulas 2000; Maleki et al. 2003; Rodoplu and Meng 1998; Gallager et al. 1979], data gathering [Kalpakis et al. 2003], topology management [Xu et al. 2001], and processor sharing [Shang et al. 2002; Rong and Pedram 2003]. Methods have been proposed to collect the residual battery status of a distributed system [Zhao et al. 2002] and also to estimate the future energy consumption at various nodes [Mini et al. 2002]. However, all these methods are based on the residual battery status and do not take into account the environmental energy availability at the nodes.

The first work to take environmental energy into account for routing was Kansal and Srivastava [2003], followed by Voigt et al. [2003]. While these works did demonstrate that environment-aware decisions improve the performance compared to battery-aware decisions for the specific application examples considered, their methods were based on heuristics and did not provide general methods to determine sustainable performance.

We also mention some of the previously used theoretical models, which are related to our models. One approach to modeling bursty sources is given by the (r, b) token bucket traffic regulator [Parekh and Gallager 1993; Parekh 1992; Cruz 1991a; Cruz 1991b] used to model bursty traffic for QoS in Internet. However, that model is not sufficient to model energy sources for harvesting purposes and we introduced a modified model appropriate for this purpose. A special case of our proposed models was considered in Kansal et al. [2004]. Methods to chose appropriate model parameters for the existing models have been explored [Dovrolis et al. 1998; Low and Varaiya 1994], but those methods are aimed at very different objectives and used to characterize packet data traffic sources. We presented related models geared for modeling energy sources and incorporated the additional constraints required.

Methods have also been suggested to evaluate the maximum data throughput from fixed energy resources [Bhardwaj and Chandrakasan 2002; Bhardwaj et al. 2001; Giridhar and Kumar 2005; Kansal et al. 2005]. These methods are again ignoring harvested energy. We showed how the objective function and the constraints change when harvested energy is considered.

7. CONCLUSIONS

We discussed how energy harvesting can be used for powering sensor networks. In addition to supplementing the energy supply in battery-powered systems, energy-harvesting can enable a new mode of operation, namely, the energy-neutral mode in which the system uses only as much energy as is available from the environment. We discussed various issues in enabling this mode and modeling the characteristics of the energy-generation sources and loads. Clearly, the energy-neutral mode is useful only if the performance constraints at the application layer can be satisfied. We presented theoretical foundations and various methods to model the achievable performance in energy-neutral mode. We also presented practical methods to optimize performance in the energy-neutral mode by accounting for observed technology characteristics, such as storage inefficiency. These real-time methods were evaluated using experimental energy data from our prototype hardware and were found to perform within a few percent of the theoretical optimal calculated using complete future knowledge. Further, we also discussed how performance may depend on the spatiotemporal profile of energy availability when a network of harvesting nodes is considered. For this case, we provided two examples, modeling two extremes on the spectrum of sensor networking applications, of how the application layer performance may be modeled for the network as a whole.

The methods presented in this work enable using environmental energy in a harvesting-aware manner and to adapt in real time to the energy availability. This yields significantly higher performance levels compared to the existing approach of using a conservative duty cycle in solar power systems, which is designed for expected worst-case scenarios. The methods discussed here have addressed some of the more common power-scaling mechanisms and usage scenarios. Several other more sophisticated power scaling techniques are available in more advanced low-power hardware and our methods may be modified to exploit all such techniques as suitable for maximizing performance at the application layer. Future work also includes integrating the harvesting-aware routing and link layer methods to provide an energy-neutral communication mechanism usable by a variety of applications, and the development of power-management techniques for unpredictable energy sources.

ACKNOWLEDGMENTS

The authors are grateful to Vijay Raghunathan for a valuable discussion regarding the models presented in this work. This research was funded in part through support provided by DARPA under the PAC/C program, the National Science Foundation (NSF) under award # 0306408, and the UCLA Center for Embedded Networked Sensing (CENS). Any opinions, findings and conclusions or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of the DARPA, NSF, and UCLA CENS.

REFERENCES

BHARDWAJ, M. AND CHANDRAKASAN, A. 2002. Bounding the lifetime of sensor networks via optimal role assignments. In *INFOCOM 2002*. New York. 1587–1596.

Bhardwaj, M., Garnett, T., and Chandrakasan, A. P. 2001. Upper bounds on the lifetime of sensor networks. In *IEEE International Conference on Communications*.

Chang, J.-H. and Tassiulas, L. 2000. Maximum lifetime routing in wireless sensor networks. In *Advanced Telecommunications and Information Distribution Research Program (ATIRP)*. College Park, MD.

Cox, D. R. 1961. Prediction by exponentially weighted moving averages and related methods. *Journal of the Royal Statistical Society. Series B (Methodological) 23,* 2, 414–422.

Cruz, R. L. 1991a. A calculus for network delay, part I: Network elements in isolation. *IEEE Transactions on Information Theory 37*, 114–131.

Cruz, R. L. 1991b. A calculus for network delay, part II: Network analysis. *IEEE Transactions on Information Theory 37*, 132–141.

DARPA. Darpa energy-harvesting projects. http://www.darpa.mil/dso/trans/energy/projects.html.

Dovrolis, C., Vedam, M., and Ramanathan, P. 1998. The selection of the token bucket parameters in the ietf guaranteed service class. Tech. rep., Georgia Institute of Technology.

Gallager, R., Humblet, P., and Spira, P. 1979. A distributed algorithm for minimum weight spanning trees. Tech. Rep. LID-P-906-A, Lab. Information Decision Systems, Massachusetts Institute of Technology. (Oct.).

Giridhar, A. and Kumar, P. 2005. Maximizing the functional lifetime of sensor networks. In *ACM Syposium on Information Processing in Sensor Networks (IPSN)*.

Heinzelman, W. 2000. Application-specific protocol architectures for wireless networks. Ph.D. thesis, PhD thesis, Massachusetts Institute of Technology.

Heliomote CVS 2005. HeliomoteR board design files. http://cvs.nesl.ucla.edu/cvs/viewcvs.cgi/HeliomoteR/.

Ieropoulos, I., Greenman, J., and Melhuish, C. 2003. Imitating metabolism: Energy autonomy is biologically inspired robots. In *Second International Symposium on Imitation of Animals and Artifacts*. Aberystwyth, Wales, UK. 191–194.

James Reserve. James San Jacinto Mountains Reserve. http://www.jamesreserve.edu.

Jiang, X., Polastre, J., and Culler, D. 2005. Perpetual environmentally powered sensor networks. In *IEEE Information Processing in Sensor Networks*. 463–468.

Kalpakis, K., Dasgupta, K., and Namjoshi, P. 2003. Efficient algorithms for maximum lifetime data gathering and aggregation in wireless sensor networks. In *Computer Networks: The International Journal of Computer and Telecommunications Networking 42*, 6, 697–716.

Kansal, A. and Karandikar, A. 2001. Adaptive delay adjustment for low jitter audio over internet. In *IEEE Globecom*. Vol. 4. 2591–2595.

Kansal, A. and Srivastava, M. B. 2003. An environmental energy harvesting framework for sensor networks. In *International symposium on low-power electronics and design*. ACM Press, New York. 481–486.

Kansal, A., Potter, D., and Srivastava, M. 2004. Performance aware tasking for environmentally powered sensor networks. In *ACM SIGMETRICS*.

Kansal, A., Ramamoorthy, A., Srivastava, M. B., and Pottie, G. J. 2005. On sensor network lifetime and data distortion. In *IEEE International Symposium on Information Theory (ISIT)*. 6–10.

Kymisis, J., Kendall, C., Paradiso, J., and Gershenfeld, N. 1998. Parasitic power harvesting in shoes. In *Second IEEE International Conference on Wearable Computing, (ISWC)*. IEEE Computer Society Press, Los Alamitos, 132–139.

Li, Q., Aslam, J., and Rus, D. 2001. Online power aware routing in wireless ad-hoc networks. In *ACM SIGMOBILE*. Rome, Italy.

Low, S. and Varaiya, P. 1994. A simple theory of traffic and resource allocation in atm. In *IEEE Infocom*. 1633–1637.

Lymberopoulos, D. and Savvides, A. 2005. Xyz: a motion-enabled, power aware sensor node platform for distributed sensor network applications. In *Fourth International Symposium on Information Processing in Sensor Networks (IPSN)* 15, 449–454.

Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D., and Anderson, J. 2002. Wireless sensor networks for habitat monitoring. In *First ACM Workshop on Wireless Sensor Networks and Applications* 28. Atlanta, GA.

Maleki, M., Dantu, K., and Pedram, M. 2003. Lifetime prediction routing in mobile ad hoc networks. In *Wireless Communication and Networking Conference*. New Orleans, LA. USA.

MENINGER, S., MUR-MIRANDA, J. O., AMIRTHARAJAH, R., CHANDRAKASAN, A., AND LANG, J. 1999. Vibration-to-electric energy conversion. In *Proceedings of the 1999 international symposium on low-power electronics and design*. ACM Press, New York. 48–53.

MIN, R., BHARDWAJ, M., CHO, S., SINHA, A., SHIH, E., WANG, A., AND CHANDRAKASAN, A. 2000a. An architecture for a power-aware distributed microsensor node. In *IEEE Workshop on Signal Processing Systems (SiPS '00)*.

MIN, R., FURRER, T., AND CHANDRAKASAN, A. 2000b. Dynamic voltage scaling techniques for distributed microsensor networks. In *IEEE Computer Society Workshop on VLSI*. 43–46.

MINI, R. A. F., NATH, B., AND LOUREIRO, A. A. F. 2002. A probabilistic approach to predict the energy consumption in wireless sensor networks. In *IV Workshop de Comunicao sem Fio e Computao Mvel*. Sao Paulo, Brazil.

MOTES 2005. Mica wireless measurement system. Datasheet, Crossbow Technology Inc. http://www.xbow.com/.

MPOWER. 2005. Battery life (and death). http://www.mpoweruk.com/life.htm#dod.

PARADISO, J. A. AND FELDMEIER, M. 2001. A compact, wireless, self-powered pushbutton controller. In *ACM Ubicomp*. Springer-Verlag Berlin Heidelberg, Atlanta, GA. 299–304.

PAREKH, A. K. 1992. A generalized processor sharing approach to flow control in integrated services networks. Ph.D. thesis, Massachusetts Institute of Technology.

PAREKH, A. K. AND GALLAGER, R. G. 1993. A generalized processor sharing approach to flow control in integrated services networks: the single-node case. *IEEE/ACM Transactions on Networking (TON) 1,* 3, 344–357.

PARK, C., CHOU, P. H., AND SHINOZUKA, M. 2005. Duranode: Wireless networked sensor for structural health monitoring. In *IEEE International Conference on Sensors*. Irvine, CA.

RABAEY, J. M., AMMER, M. J., DA SILVA JR., J. L., PATEL, D., AND ROUNDY, S. 2000. Picoradio supports ad hoc ultra-low-power wireless networking. *IEEE Computer.* 42–48.

RAGHUNATHAN, V., SCHURGERS, C., PARK, S., AND SRIVASTAVA, M. 2002. Energy aware wireless microsensor networks. *IEEE Signal Processing Magazine 19,* 2 (March), 40–50.

RAGHUNATHAN, V., KANSAL, A., HSU, J., FRIEDMAN, J., AND SRIVASTAVA, M. B. 2005. Design considerations for solar energy-harvesting wireless embedded systems. In *IEEE Information Processing in Sensor Networks (IPSN)*. Los Angeles, CA. 457–462.

RAHIMI, M., SHAH, H., SUKHATME, G. S., HEIDEMANN, J., AND ESTRIN, D. 2003. Studying the feasibility of energy-harvesting in a mobile sensor network. In *IEEE Int'l Conference on Robotics and Automation*.

RODOPLU, V. AND MENG, T. H. 1998. Minimum energy mobile wireless networks. In *IEEE International Conference on Communications*. Vol. 3. Atlanta, GA. 1633–1639.

RONG, P. AND PEDRAM, M. 2003. Extending the lifetime of a netork of battery powered mobile devices by remote processing: A markovian decision based approach. In *ACM Design Automation Conference (DAC)*. Anaheim, CA.

ROUNDY, S., WRIGHT, P. K., AND RABAEY, J. M. 2004. *Energy Scavenging for Wireless Sensor Networks with Special Focus on Vibrations*. Kluwer Academic, Publ., Novell, MA.

SHAH, R. C. AND RABAEY, J. M. 2002. Energy aware routing for low energy ad hoc sensor networks. In *Proc. IEEE Wireless Communications and Networking Conference (WCNC)*. Vol. 1. Orlando, FL. 350–355.

SHANG, L., DICK, R. P., AND JHA, N. K. 2002. An economics-based power-aware protocol for computation distribution in mobile ad-hoc networks. In *14th IASTED International Conference on Parallel and Distributed Computing and Systems (PDCS)*.

SHENCK, N. S. AND PARADISO, J. A. 2001. Energy scavenging with shoe-mounted piezoelectrics. *IEEE Micro 21,* 3 (May–June), 30–42.

SINGH, S., WOO, M., AND RAGHAVENDRA, C. S. 1998. Power-aware routing in mobile ad hoc networks. In *ACM/IEEE Mobicom*.

SINHA, A. AND CHANDRAKASAN, A. 2001. Dynamic power-management in wireless sensor networks. *IEEE Design and Test of Computers 18*, 62–74.

STARGATE. 2004. Stargate xscale processor platform. http://www.xbow.com/.

STARNER, T. 1996. Human-powered wearable computing. *IBM Systems Journal 35,* 3–4.

VOIGT, T., RITTER, H., AND SCHILLER, J. 2003. Utilizing solar power in wireless sensor networks. In *The 28th Annual IEEE Conference on Local Computer Networks (LCN)*. Bonn/Konigswinter, Germany.

WEBER, W. 2003. Ambient intelligence: industrial research on a visionary concept. In *Proceedings of the 2003 International Symposium on Low-Power Electronics and Design*. ACM Press, New York. 247–251.

WIND DATA 2001. Noaa recorded average wind speed data through 2001. http://www.berner.com/new/energy-windspeed.htm.

WRIGHT, S., SCOTT, D., HADDOW, J., AND ROSEN, M. 2000. The upper limit to solar energy conversion. In *35th Energy Conversion Engineering Conference and Exhibit (IECEC)*. Vol. 1. 384–392.

XU, Y., HEIDEMANN, J., AND ESTRIN, D. 2001. Geography-informed energy conservation for ad hoc routing. In *Proc. ACM Mobicom*. ACM Press, New York. 70–84.

YOUNIS, M., YOUSSEF, M., AND ARISHA, K. 2002. Energy-aware routing in cluster-based sensor networks. In *Proc. 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*.

ZHAO, J., GOVINDAN, R., AND ESTRIN, D. 2002. Residual energy scans for monitoring wireless sensor networks. In *IEEE Wireless Communications and Networking Conference (WCNC'02)*. Vol. 1. Orange County Convention Center, Orlando, FL. 356–362.

Received January 2006; revised May 2006; accepted July 2006