

Analysis of Power-Aware Buffering Schemes in Wireless Sensor Networks

YIBEI LING and CHUNG-MIN CHEN

Telcordia Technologies

and

SHIGANG CHEN

University of Florida

We study the power-aware buffering problem in battery-powered sensor networks, focusing on the fixed-size and fixed-interval buffering schemes. The main motivation is to address the yet poorly understood size variation-induced effect on power-aware buffering schemes. Our theoretical analysis elucidates the fundamental differences between the fixed-size and fixed-interval buffering schemes in the presence of data-size variation. It shows that data-size variation has detrimental effects on the power expenditure of the fixed-size buffering in general, and reveals that the size variation induced effects can be either mitigated by a positive skewness or promoted by a negative skewness in size distribution. By contrast, the fixed-interval buffering scheme has an obvious advantage of being eminently immune to the data-size variation. Hence the fixed-interval buffering scheme is a risk-averse strategy for its robustness in a variety of operational environments. In addition, based on the fixed-interval buffering scheme, we establish the power consumption relationship between child nodes and parent node in a static data-collection tree, and give an in-depth analysis of the impact of child bandwidth distribution on the parent's power consumption.

This study is of practical significance: it sheds new light on the relationship among power consumption of buffering schemes, power parameters of radio module and memory bank, data arrival rate, and data-size variation, thereby providing well-informed guidance in determining an optimal buffer size (interval) to maximize the operational lifespan of sensor networks.

Categories and Subject Descriptors: F.2.m [Analysis of Algorithms and Problem Complexity]: Miscellaneous; H.4.0 [Information Systems Applications]: General

General Terms: Algorithms

Additional Key Words and Phrases: Power-aware buffering schemes, algorithm analysis

Authors' addresses: Y. Ling and C.-M. Chen, Applied Research, Telcordia Technologies, 1 Telcordia Dr., Piscataway, NJ, 08854 94301. Shigang Chen, Department of Computer and Information of Science and Engineering, University of Florida, Gainesville, FL 32611.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or permissions@acm.org.
© 2010 ACM 1550-4859/2010/09-ART26 \$10.00
DOI 10.1145/1807048.1807055 <http://doi.acm.org/10.1145/1807048.1807055>

ACM Reference Format:

Ling, Y., Chen, C.-M., and Chen, S. 2010. Analysis of power-aware buffering schemes in wireless sensor networks. *ACM Trans. Sensor Netw.*, 7, 3, Article 26 (September 2010), 33 pages.
DOI = 10.1145/1807048.1807055 <http://doi.acm.org/10.1145/1807048.1807055>

1. INTRODUCTION

A dramatic rise in research interest in power-conscious computing is attributed, in part, to the growing awareness of the greenhouse effect brought about by the exponentially increasing number of computing devices [Xie 2008; Satyanarayanan 1996]. It is also driven by the impetus to meet the long-duration operational requirement of battery-powered sensor networks [Mainwaring et al. 2002; Woo et al. 2003; Culler and Mulder 2004; Gupta and Singh 2003]. This work is motivated by problems arising from power-aware computing in general and by battery-based sensor networks in particular.

A sensor network could be composed of hundreds to thousands of tiny sensor nodes. Each sensor node typically comprises a couple of sensors, memory banks, a radio, and a microcontroller [Hempstead et al. 2005], being equipped with a stripped-down version of the operating system. The sensor node can perform some basic computational tasks such as data measurement, filtering, aggregation, transmission/reception, and packet routing. Once deployed in the field, sensor nodes can self-organize into a perceptive network that enables novel ways to respond to emergencies, habitat monitoring, and around-clock environmental surveillance. The sensor nodes are required to autonomously operate under harsh conditions for several months, even years, without human intervention and maintenance [Mainwaring et al. 2002]. In certain cases, battery replacement or recharge may not even be possible [Mainwaring et al. 2002; Landman and Rabaey 1995]. Thus the premise of sensor networks to detect rarely occurring events or to monitor chronically changing events largely depends on the lifespan of the sensor network. A review of essential features required by sensor-based network applications yields a long list: resilience, fault tolerance, self-organization, and autonomy. Despite such a rich feature set, the core requirement of the sensor network is power conservation.

In a drive to bring power-aware computing to fruition, research efforts have proceeded along three distinct yet closely related tracks: 1) battery technologies; 2) hardware-based technologies; and 3) software-based technologies. Among these technologies, battery technologies appear to be self-contained. Hardware-based and software-based technologies are sharply distinct but mutually dependent as well.

The power conservation requirement fundamentally reshapes how hardware modules should be designed, implemented and assembled. The evolution of the hardware-based approach is a process of continuously replacing power-inefficient components with ultra-low power modules, and substituting general-purpose components with specially designed power efficient ones. Wireless radio and memory components have long been recognized as the biggest power spenders in a sensor node system [Lee et al. 2007]. Realization of this shortcoming has directed research attention toward designing ultra-low power

radio and memory components with multipower mode capability [Lee et al. 2007; Flautner et al. 2002]. However, the hardware multipower mode capability alone does not warrant power efficient computing in practice. The reason is that a transition between operating power modes (from a low-power mode to a high-power mode or vice versa) bears a *resynchronization cost*, that is, a certain amount of energy incurred to demote or to elevate the operating power level. As a result, the availability of hardware multipower mode capability presents a new set of collateral risks of being misused: a blind choice of operating power mode might incur an excessive transition cost that neutralizes the benefits brought out by power-aware hardware design.

To reap the benefit of multipower mode feature in a hardware design, software-based technologies are concerned with the design of algorithms/protocols that can exploit the multipower capability, thus serving as a reinforcer to the hardware-based power-aware technologies. The whole idea underlying the software-based approaches centers around the exploitation of quiescence in workload, linking the power mode of a component to its workload characteristics.

Lee et al. [2007] introduced a power-aware buffer cache management scheme called PABC for compressing and migrating active pages in both user-space and kernel-space onto a few memory units. Their experimental study indicated that the PABC scheme can reduce the energy consumption of the buffer cache by an impressive 63%. Flautner et al. [2002] observed that in practice the hot (active) cache only accounts for a small subset of on-chip caches for most of time. This observation leads to an architectural design that exploits such a workload pattern to place the cold cache into drowsy mode, thereby saving a substantial power consumption. The experimental studies showed that about 80%–90% of cache can be maintained in a drowsy (idle) mode without affecting performance by more than 1%. Ling and Chen [2007], on the other hand, derived closed form optimal buffering strategies, under the condition that the received data size is entirely devoid of variability and identical to the size of a memory bank. This assumption greatly simplifies mathematical derivation. It, however, appears to be inadequate in capturing the essence of sensor networks in a realistic setting.

It is widely recognized that idle listening is the major energy spender in sensor networks. For example, experimental study shows that 99% of energy is dissipated on idle listening if a node is always turned on [Lin and Stankovic 2005; Shnayder et al. 2004]. Many power management protocols are proposed to reduce power consumption in listening. Asynchronous low power listening (ALPL) scheme uses duty cycling to reduce the listening energy. A node is required to periodically wake up and check the radio channel. In general, the energy saving on listening at receiver is at the expense of sender, because the sender must open the radio channel long enough to ensure correct message reception. Synchronous Low Power Listening scheme (SLPL) improves on the ALPL scheme in its ability to coordinate sender's transmit mode with the receiver's periodic check [Ye et al. 2002; Jurdak et al. 2007]. The weakness of SLPL is that it demands a high-quality time synchronization among a group of nodes, which incurs a nonnegligible amount of energy. In addition, the design of an energy efficient wake-up/sleep protocol is often application

dependent and complicated in practice. Hence, it is hard to design a general power management system based on wake-up/sleep scheduling.

A radically different approach, called radio-triggered wake-up power management, is proposed by Lin and Stankovic [2004], Lin and Stankovic [2005], and Ansari et al. [2009]. It uses a radio-triggered circuit as one interrupt input of the processor. The circuit itself does not require any power supply and is powered by the radio signals themselves. As a result, the radio-triggered power scheme allows nodes to sleep without need for periodic wake-up to check channel signals, thereby completely eliminating listening power consumption.

In this article we study the power-aware buffering problem by exploiting the multipower mode in radio and memory components and the radio-triggered power management. The optimization objective is to minimize power consumption in the context of two buffering paradigms: the well-known *fixed-size* and the lesser-known *fixed-interval* buffering schemes. In particular, we focus on the size variability-induced effect on these power-aware buffering schemes.

To the best of our knowledge, the effect of size variability on power consumption of buffering schemes has not been addressed before. Our analysis provides insight into the poorly understood effect of size variability on the power-aware buffering schemes, thereby providing a theoretical guidance for performance tuning in practice. The novelty of this article is its adoption of asymptotic analysis, which allows us to model the limitation of power-aware buffering schemes without sacrificing simplicity and elegance.

The remainder of this article is organized as follows. Section 2 presents relevant definitions and prerequisite theorems that facilitate derivation of the main theorems. Section 3 presents the exposition of theoretical analysis for both power-aware fixed-size and fixed-interval buffering schemes. Section 4 compares the performance between the fixed-size and fixed-interval buffering schemes in both the absence and presence of size variation. Section 5 discusses the gain of power-aware buffering schemes over power-oblivious ones in terms of power conservation, with some examples to illustrate the effect of power-aware buffering on the lifespan of sensor nodes. The power consumption relationship between the parent and child nodes in a data collection tree is presented. Section 6 concludes this article.

2. BACKGROUND

Multipower mode radio and memory components are the main hardware prerequisites of power-aware buffering schemes in this paper. The efficacy of power-aware hardware design relies on the ability of software-based approach to exploit the potential of power-aware hardware design.

To study the performance of power-aware buffering schemes, let us discuss at length the power-mode transition pattern of radio and memory components. We assume that nodes use the radio-triggered power management, thus do not incur listening power consumption.

The power-mode of a multipower radio component can be subsumed into (1) the sleep mode and (2) the active mode. A sleep-mode radio inhibits data transmission/reception. An active-mode radio permits data transmission/reception

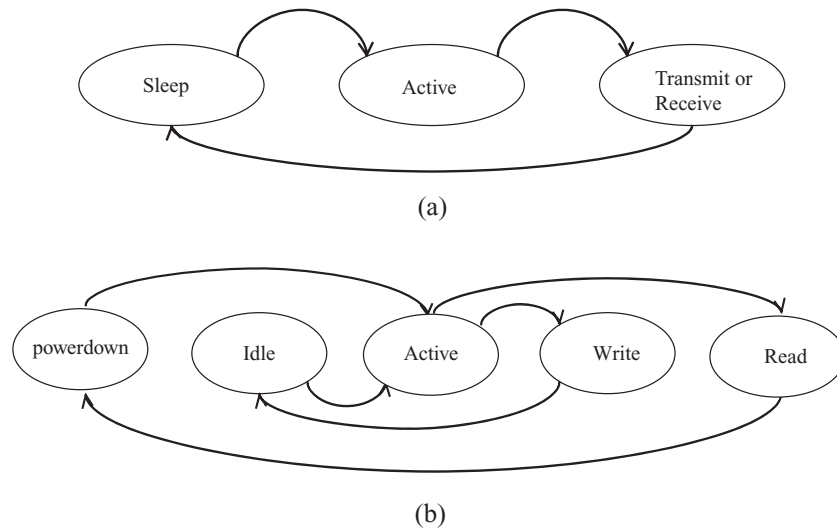


Fig. 1. (a) Power state transition diagram of wireless radio module (b) Power state transition diagram of memory bank.

but incurs more power than when in sleep mode. To save power consumption, the radio is placed into sleep mode most of the time; it is only elevated to active mode (by a radio-triggered wakeup component) when data transmission/reception is needed. After completing data transmission/reception, the radio is put back to sleep mode. The *sleep-active-transmit-sleep* transition diagram in Figure 1(a) forms a typical power-aware radio working pattern.

A *memory bank* refers to the minimum size of a memory unit whose power mode can be independently altered [Hempstead et al. 2005]. Its power mode could be broadly classified into three categories: (1) the powerdown mode, (2) the idle mode, and (3) the active mode. A powerdown-mode memory bank means that the voltage supply to the memory bank is cut off, resulting in a sizable reduction in current leakage [Flautner et al. 2002; Tarjan et al. 2006]. The idle (sleep or drowsy) mode is the minimum power mode that preserves the stored information but inhibits writing and reading of data. An idle-mode or powerdown-mode memory bank must be reinstated to the active mode before a read/write operation can be performed. An active mode memory bank not only retains the stored information but also allows the data to be written/read. The power consumption in a powerdown-mode memory bank is negligibly small. An idle-mode memory bank consumes less power and has less functionality than an active-mode memory bank.

In an ideal power-saving data buffering scenario, the power-mode transition could be divided into (1) the write power-mode transition and (2) the read power-mode transition. A write power-mode transition is initiated by an interrupt of radio-triggered circuit. The sensor node first powers up the memory banks from the powerdown mode to the active mode, and then writes data into the memory banks. After completing a *write* operation, the involved memory banks are demoted to the idle mode to preserve power while retaining the

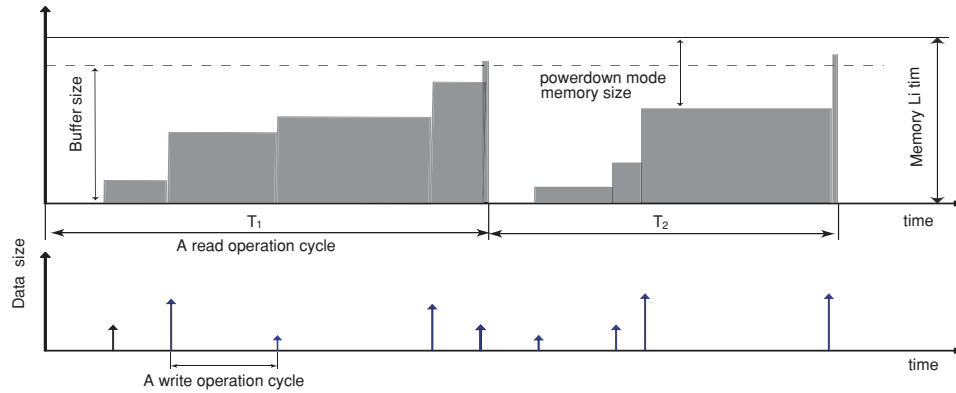


Fig. 2. Evolution of powerdown-mode memory size (blue) and idle-mode memory size (red).

stored information. The *powerdown-active-write-idle* power mode transition in Figure 1(b) forms a *write* operation cycle.

A read power-mode transition is initiated when a specified buffer size or interval threshold is reached. Thus, a *read* power-mode transition may comprise more than one *write* power-mode transition cycle, depending on the specified buffer size. It involves loading, transmitting and clearing up all the buffered data. To do so, it elevates the power mode of the buffered memory banks from idle mode to active mode. Once reading of data is completed, the corresponding memory banks are put back to powerdown mode. The *idle-active-read-powerdown* power mode transition in Figure 1(b) forms a *read* operation cycle, which is synchronized with the *sleep-active-transmit-sleep* power mode transition in the radio. In other words, data transmission in the radio is initiated immediately right after reading/loading buffered data from the memory banks.

The bottom graph in Figure 2 depicts the size of arrival data (collected via sensors) as a function of time. The top graph shows the evolution of powerdown-mode memory size (total memory size minus stair height) and of idle-mode memory size (stair height). Observe that the powerdown-mode memory size shrinks as the arrival data are accumulated in idle-mode memory banks. The size of idle-mode memory banks grows in a stair-like fashion when it is less than the prescribed buffer size. Once it hits the prescribed buffer size and a transmission of buffered data is initiated. This forms a *read* power-mode transition cycle for memory banks, as well as a radio power-mode transition cycle. In practice, the duration of the *read* transition cycle may fluctuate widely: it could be very sensitive to the buffering policy, data arrival rate and data size distribution. In order to analyze the power-aware buffering issues, we begin with two buffering policies as follows:

Definition 1. A buffering policy is said to be *stationary* if its decision depends only on its current state and not on the time. A buffering policy is said to be *deterministic* if its decision is a *deterministic* function of the current state.

- (1) Fixed-size buffering scheme. Data transmission is initiated immediately when a fixed (prescribed) buffer size is reached.

- (2) Fixed-interval buffering scheme. Data transmission is commenced periodically with a fixed time interval.

There is a clear distinction between the well-known fixed-size and lesser-known fixed-interval buffering schemes: the threshold of the fixed-size buffering scheme depends on the size of the buffered data, and that of the fixed-interval one depends on a specified time interval. By definition 1, the fixed-size buffering scheme is stationary while the fixed-interval buffering scheme is deterministic. For notational convenience, we use the superscripts *FS* and *FI* to denote the fixed-size and fixed-interval buffering schemes throughout the article. Before delving into a detailed derivation, we introduce relevant notions and essential prerequisites.

Definition 2. Let x be a random variable following a probability distribution \mathcal{F} , i.e., $x \sim \mathcal{F}$, the skewness of x , denoted by $\gamma(x)$, is defined as

$$\gamma(x) = \frac{E[(x - \mu_x)^3]}{(E[x - \mu_x]^2)^{3/2}} = \frac{\mu_3}{\sigma_x^3}. \quad (1)$$

The coefficient of variation of x , denoted by $c_v(x)$, is defined as

$$c_v(x) = \frac{\sqrt{E[(x - \mu_x)^2]}}{E[x]} = \frac{\sigma_x}{\mu_x}, \quad (2)$$

where $E[\cdot]$ is the expected function and $\mu_x = E[x]$.

In probability theory, $\gamma(x)$ is the third standardized moment for measuring the degree of asymmetry. It can be further divided into positive ($\gamma(x) > 0$) and negative skewness ($\gamma(x) < 0$). The function $c_v(x)$ is a measure for the degree of dispersion.

Definition 3. An integer-valued random variable \mathbf{n} is said to be a *stopping time* for the sequence x_1, x_2, \dots if the event $\{\mathbf{n} = n\}$ is independent of x_{n+1}, x_{n+2}, \dots for all $n = 1, 2, \dots$

Wald's equation: Suppose y_1, y_2, \dots are iid random variable with finite expectation $E[y_i] = \mu_y$, and \mathbf{n} is a stopping time for y_1, y_2, \dots such that $E[\mathbf{n}] < \infty$, then

$$E \left[\sum_{i=1}^{\mathbf{n}} y_i \right] = E[\mathbf{n}]E[y] = \mu_y E[\mathbf{n}]. \quad (3)$$

The following theorem establishes the asymptotic behavior of stopping time variance w.r.t buffer size b .

THEOREM 1. Let $\{x_i > 0, i \geq 1\}$ be a random positive walk (increment) with mean of $\mu_x = E[x_i] > 0$ and finite variance of σ_x^2 . Let stopping time $\tau(b) = \min\{n \geq 1 : \sum_1^n x_i > b\}$. When b is sufficiently large, the stopping time variance $\sigma_{\tau_b}^2$ becomes

$$\sigma_{\tau_b}^2 = \frac{b\sigma_y^2}{\mu_y^3} + k^* = \frac{bc_v^2(y)}{\mu_y} + k^*, \quad (4)$$

where k^* is expressed as

$$k^* = \frac{5c_v^4(y)}{4} + \frac{1}{12} - \frac{2c_v^3(x)\gamma(y)}{3}, \quad (5)$$

where $c_v(y)$ denotes the coefficient of variation, and $\gamma(y)$ the skewness of y .

Proof of Theorem 1 is given in the Appendix.

Theorem 1 states that in an asymptotic sense, the stopping time variance is linearly proportional to the buffer size b , with a proportionality constant of $c_v^2(y)/\mu_y$ and the intercept k^* determined by both $c_v(x)$ and $\gamma(y)$. It means that $c_v^2(y)/\mu_y$ could play a central role in determining the stopping time variance. The magnitude of intercept (k^*) can be either mitigated by positive skewness or augmented by negative skewness. It is noteworthy that Theorem 1 is a special case of Lau's theorem [Lai and Seigmund 1977, 1979] under the positive random increment condition, which results in a substantial simplification. The following corollaries are special cases of Theorem 1 in which the random walk (increment) is assumed to be exponentially or Erlangly distributed.

COROLLARY 1. *For a given buffer size b , the stopping time variance $\tau(b)$ for an exponential random walk with mean $1/\lambda_e$ is*

$$\sigma_{\tau(b)}^2(\text{exp}) = \lambda_e b, \quad (6)$$

where $\sigma_{\tau(b)}(\text{exp})$ refers to the stopping time variance w.r.t. an exponential random walk (increment).

PROOF. For an exponential random walk, by Definition 2 we get $c_v(y) = 1$, $\gamma(y) = 2$. Substituting $c_v(y)$ and $\gamma(y)$ into (4) leads to (6). \square

COROLLARY 2. *For a given buffer size b , the stopping time variance for an Erlang random walk with parameters (α, λ_α) is*

$$\sigma_{\tau(b)}^2(\text{erlang}) = \frac{\lambda_\alpha b}{\alpha^2} + \frac{1}{12} \left(1 - \frac{1}{\alpha^2}\right), \quad (7)$$

where $\alpha > 1$ is the shape parameter (an integer), λ_α refers to the rate, and $\sigma_{\tau(b)}(\text{erlang})$ refers to the stopping time variance w.r.t. an Erlang random walk (increment).

PROOF. By Definition 2, for an Erlang random walk, we obtain $c_v(y) = \frac{1}{\sqrt{\alpha}}$ and $\gamma(y) = \frac{2}{\sqrt{\alpha}}$. Substitution of $c_v(y)$ and $\gamma(y)$ into (4) yields (7). \square

Consider the differential stopping time variance between the exponential and the Erlang random walks by subtracting (6) from (7).

$$\sigma_{\tau(b)}^2(\text{exp}) - \sigma_{\tau(b)}^2(\text{erlang}) = b\lambda_e - \frac{\lambda_\alpha b}{\alpha^2} + \frac{1}{12} \left(1 - \frac{1}{\alpha^2}\right) \quad (8)$$

Notice that the mean increment size of exponential random walk is $\mu_e = 1/\lambda_e$ and that of the Erlang walk is $\mu_\alpha = \frac{\alpha}{\lambda_\alpha}$. Letting $\mu_\alpha = \mu_e = \mu$, then (8) is reduced

to

$$\sigma_{\tau(b)}^2(exp) - \sigma_{\tau(b)}^2(erlang) = \left(1 - \frac{1}{\alpha}\right) \left(\frac{b}{\mu} - \frac{1}{12} \left(1 + \frac{1}{\alpha}\right)\right), \quad (9)$$

then

$$\sigma_{\tau(b)}^2(exp) - \sigma_{\tau(b)}^2(erlang) \begin{cases} > 0 & \frac{b}{\mu} > \frac{1}{12} \left(1 + \frac{1}{\alpha}\right) \\ \leq 0 & \frac{b}{\mu} \leq \frac{1}{12} \left(1 + \frac{1}{\alpha}\right). \end{cases} \quad (10)$$

Equation (10) means that with the same mean increment size, the stopping time under the exponential random walk (increment) has a wider variance than that under the Erlang walk as long as the condition $b > \mu/12$ is met.

Consider a hyper-exponential random walk (increment) as $\sum_{i=1}^2 p_i \lambda_i \exp(-\lambda_i x)$ where $\sum_{i=1}^2 p_i = 1$ (letting $p_1 = p$, $p_2 = 1 - p$). The differential stopping time variance between the hyperexponential and the exponential walks is $\sigma_{\tau(b)}^2(hp) - \sigma_{\tau(b)}^2(exp)$. Under the same mean increment, it becomes

$$\sigma_{\tau(b)}^2(hp) - \sigma_{\tau(b)}^2(exp) = \frac{b(c_v^2(y) - 1)}{\mu} + k^*, \quad (11)$$

where k^* is explicitly given in (4). This implies that the differential stopping time variance is linearly proportional to the buffer size b , that is, $\sigma_{\tau(b)}^2(hp) - \sigma_{\tau(b)}^2(exp) \propto \frac{b}{\mu}(c_v^2(y) - 1) > 0$ when b is sufficiently large. Namely, in an asymptotic sense, the hyperexponential random walk has a wider variance in the stopping time than the exponential walk under the same mean increment size condition.

Consider the *fixed-size buffering* scheme with a size of b . Define the stopping time, denoted by $\tau(b)$, to be a random variable that takes on values in $[0, \infty)$. One sees that $\tau(b)$ is a function of b and the size distribution of the data $\{y_i > 0 : i \geq 0\}$:

$$\tau(b) = \min \left\{ n : \sum_{i=1}^n y_i \geq b \right\}, \quad (12)$$

where $\tau(b)$ is referred to as the *first ladder epoch* and $\sum_{i=1}^{\tau(b)} y_i$ is called the *first epoch height* [Lai and Seigmund 1977, 1979; Feller 1971].

One key step is to establish a relationship between the mean stopping time (the first ladder epoch) and the mean size of the data distribution. Assume that the sensor node has enough buffer capacity to accommodate *first ladder height* (overshoot) with respect to the buffer size b .

THEOREM 2. *Let $\{y_i > 0, i \geq 0\}$ be the sequence of increment sizes with mean μ_y , and b be the buffer size, the mean stopping time $E[\tau(b)] \approx \frac{b}{\mu_y}$.*

PROOF. By Wald's equation in (3) we obtain the relation $\sum_{i=1}^{\tau(b)-1} y_i < b \leq \sum_{i=1}^{\tau(b)} y_i$. Taking expectation on both sides of this relation yields

$$E \left[\sum_{i=1}^{\tau(b)-1} y_i \right] < b \leq E \left[\sum_{i=1}^{\tau(b)} y_i \right] \implies (E[\tau(b)] - 1)\mu_y < b \leq E[\tau(b)]\mu_y. \quad (13)$$

Table I. Symbols and Meanings

Data Traffic	
λ	Poisson data arrival rate
μ_y	mean value of data size distribution
b_{size}	size of a memory bank
$\lambda\mu_y$	bandwidth
Radio Module	
e_w^{wu}	energy for a radio wakeup
e_w^{RX}	energy for one-byte reception
e_w^{TX}	energy for one-byte transmission ($e_w^{RX} \approx e_w^{TX}$)
Memory Bank	
p_m^{idle}	power of idle state of one memory bank
e_m^{ena}	energy to elevate from powerdown to active
e_m^{dem}	energy to demote from active to idle
e_m^r	energy of reading one byte
e_m^w	energy for writing one byte
e_m^{resyn}	$(e_m^{ena} + e_m^{dem})/2$

Dividing both sides of (13) by μ_y completes the proof. \square

The preceding theorem asserts that the *fixed-size* buffering scheme with a buffer size of b can hold $\frac{b}{\mu_y}$ data packets on average when the data size is randomly distributed with a mean of μ_y , which is in line with our intuition.

For the sake of clarity, we summarize the power parameters in Table I. The subscripts m and w denote the memory bank and radio module. e_m^{ena} and e_m^{dem} refer to the energy required to elevate a powerdown-mode memory bank to active mode, and to demote an active-mode memory bank to idle mode, e_m^{resyn} is a resynchronization cost being equal to the mean value of e_m^{ena} and e_m^{dem} , and $\lambda\mu_y$ the data volume per time unit, termed as *bandwidth* due to conceptual similarity. Since the duration of an active-mode memory bank is extremely short, thus the energy consumed in the active-mode could be reasonably ignored. Similarly, the energy consumed by the active-mode of a radio module is outweighed by e_w^{TX} , e_w^{RX} , and hence is ignored.

3. POWER-AWARE BUFFERING SCHEMES

3.1 Fixed-Size Buffering Scheme

In this section, we consider the fixed-size buffering scheme under randomly distributed data size with Poisson arrival. Assume that data size follows a certain probability distribution with a finite mean of μ_y . Let (x_i, y_i) , $i \geq 0$ be a sequence of random vectors in which $\{x_i, i \geq 0\}$ refers to a random variable denoting the interarrival times of Poisson arrival data and $\{y_i, i \geq 0\}$ be a random variable representing the size of the arrival data. The random variables x_i and y_i are assumed to be mutually independent.

THEOREM 3. *Let λ be a Poisson arrival rate, μ_y be the mean data size, b_{size} be the size of a memory bank, e_w^{wu} be the per radio wakeup energy, and p_m^{idle} be the idle-mode power consumption of a memory bank. Then the optimal buffer*

size b^* for the fixed-size buffering scheme is

$$b^* = \sqrt{\frac{2b_{\text{size}}e_w^{wu}\lambda\mu_y}{p_m^{\text{idle}}} + \mu_y^2 k^*}, \quad (14)$$

where k^* is given in (4).

PROOF. Consider a random vector sequence (x_i, y_i) , $i \geq 0$, where $\{x_i, i \geq 0\}$ represents the arrival time instants (Poisson arrival) and $\{y_i, i \geq 0\}$ is a sequence of received data sizes, with a mean $E[y_i] = \mu_y$ and a variance σ_y^2 .

Define a renewal reward process [Ross 1996] with the cycle length being equal to the time duration of stopping time $\tau(b)$ as follows:

$$L_c = \sum_{i=0}^{\tau(b)} x_{i+1} - x_i, \quad (15)$$

where L_c denotes the length of a renewal cycle, and $x_{i+1} - x_i$, $i \geq 0$ is interarrival times. Letting $s_k = \sum_{i=1}^k x_i$. Thus the total energy $e^{FS}(b)$ over a renewal cycle is

$$\begin{aligned} e^{FS}(b) = & e_w^{wu} + \frac{p_m^{\text{idle}}}{b_{\text{size}}} \sum_{i=1}^{\tau(b)} (s_{\tau(b)} - s_i) y_i + \sum_{i=1}^{\tau(b)} (e_w^{TX} y_i) + \sum_{i=1}^{\tau(b)} (e_w^{RX} y_i + e_w^{wu}) \\ & + \sum_{i=1}^{\tau(b)} (e_m^w + e_m^r) y_i + 2e_m^{\text{resyn}}. \end{aligned} \quad (16)$$

Let us return to explaining each term in (16). $\frac{p_m^{\text{idle}}}{b_{\text{size}}} \sum_{i=1}^{\tau(b)} (s_{\tau(b)} - s_i) y_i$ denotes the accumulated idle-mode energy for the number of memory banks in a renewal cycle, and $\sum_{i=1}^{\tau(b)} e_w^{TX} y_i$ refers to the transmission energy, $\sum_{i=1}^{\tau(b)} (e_m^w + e_m^r) y_i + 2e_m^{\text{resyn}}$ is the total energy required to write/read data into/from the memory banks, plus the resynchronization energy, and $\sum_{i=1}^{\tau(b)} e_w^{RX} y_i + e_w^{wu}$ refers to the total energy for receiving data, plus the energy for radio wakeup for receiving data. The term e_w^{wu} refers to per radio wakeup energy for data transmission. In other words, in each renewal cycle, the transmission radio wakeup occurs only once, while the reception radio wakeup occurs $\tau(b)$ times. Recall that we assume that nodes use the radio-triggered power management scheme, thereby the radio wake-up can be initiated without incurring listening energy. By Wald's equation, we get

$$E[L_c] = E \left[\sum_{i=0}^{\tau(b)} x_{i+1} - x_i \right] = E[\tau(b)] E[x_{i+1} - x_i] = \frac{E[\tau(b)]}{\lambda} = \frac{b}{\lambda\mu_y}.$$

Define $e^{FS}(t)$ to be the accumulated energy consumption at time t , where multiple renewal cycles may have occurred in the time period $[0, t]$. By the renewal reward theory [Ross 1996], the long-run mean average energy consumption is

$$\overline{e^{FS}(b)} \stackrel{\text{def}}{=} \lim_{t \rightarrow \infty} \frac{e^{FS}(t)}{t} = \frac{E[e^{FS}(b)]}{E[L_c]} = \frac{E[e^{FS}(b)]}{\frac{b}{\lambda\mu_y}}, \quad (17)$$

where the unit of $\overline{e^{FS}(b)}$ is the watt (W), rather than the joule (J). Letting $e_w^{TX} = e_w^{RX}$. Taking expectation of the third term in (16) and applying Theorem 2,

gives

$$\begin{aligned}
& E \left[\sum_{i=1}^{\tau(b)} (2e_w^{TX} + e_m^w + e_m^r) y_i + e_w^{wu} + 2e_m^{resyn} \right] \\
&= E[\tau(b)] E \left[(2e_w^{TX} + e_m^w + e_m^r) y_i + e_w^{wu} + 2e_m^{resyn} \right] \\
&= \frac{b}{\mu_y} \left(\mu_y (2e_w^{TX} + e_m^w + e_m^r) + e_w^{wu} + 2e_m^{resyn} \right). \tag{18}
\end{aligned}$$

It follows from (4) and the assumption of independence of y_i and $s_i = \sum_{j=1}^i x_j$, the expectation of the second term in (16) thus becomes

$$\begin{aligned}
E \left[\sum_{i=1}^{\tau(b)} (s_{\tau(b)} - s_i) y_i \right] &= \mu_y E \left[\sum_{i=1}^{\tau(b)} s_{\tau(b)} - s_i \right] = \frac{\mu_y (E[\tau^2(b)] - E[\tau(b)])}{2\lambda} \tag{19} \\
&= \frac{\mu_y (E^2[\tau(b)] + \sigma_{\tau(b)}^2 - E[\tau(b)])}{2\lambda} = \frac{b^2}{\mu_y} + \frac{b(c_v^2(y) - 1) + \mu_y k^*}{2\lambda},
\end{aligned}$$

where k^* is given in (5). Substitution of (18)–(19) into (17) yields

$$\begin{aligned}
\overline{e^{FS}(b)} &= \frac{\lambda \mu_y e_w^{wu} + \frac{p_m^{idle} \mu_y^2 k^*}{2b_{size}}}{b} + \lambda (\mu_y (2e_w^{TX} + e_m^w + e_m^r) + e_w^{wu} + 2e_m^{resyn}) \\
&\quad + \frac{p_m^{idle} \mu_y}{2b_{size}} (c_v^2(y) - 1) + \frac{p_m^{idle} b}{2b_{size}}. \tag{20}
\end{aligned}$$

Solving $\frac{\partial \overline{e^{FS}(b)}}{\partial b} = 0$ yields $b^* = \sqrt{\frac{2e_w^{wu} b_{size} \lambda \mu_y}{p_m^{idle}} + k^* \mu_y^2}$. To prove that b^* is the optimal buffer size, it suffices to show that

$$\lim_{b \rightarrow b^*} \frac{\partial^2 \overline{e^{FS}(b)}}{\partial b^2} = 2 \frac{e_w^{wu} \lambda \mu_y + \frac{p_m^{idle} k^* \mu_y^2}{2b_{size}}}{(b^*)^3} > 0. \tag{21}$$

The proof is thus completed. \square

The following corollary is a special case of Theorem 3.

COROLLARY 3. *When the size of received data is constant and identical to that of a memory bank, the optimal size of power-aware fixed-size buffering is expressed as*

$$n^* = \sqrt{\frac{2\lambda e_w^{wu}}{p_m^{idle}} + \frac{1}{12}} \approx \sqrt{\frac{2\lambda e_w^{wu}}{p_m^{idle}}}, \tag{22}$$

where n^* in (22) refers to the number of memory banks used, hence the optimal buffer size is $b^* = n^* b_{size}$ (b^* is a multiple of memory bank size b_{size}).

PROOF. It is almost trivial and therefore omitted. \square

Theorem 3 takes into account the impact of unevenly distributed data size, thereby generalizing the previous work [Ling and Chen 2007] beyond the

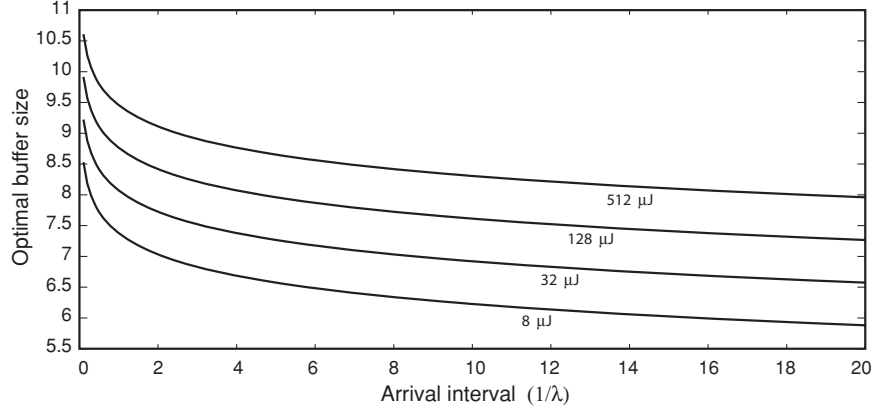


Fig. 3. Optimal buffer size b^* vs. $1/\lambda$: [$p_m^{idle}(0.409\mu W)$, $b_{size}(256b)$, $\mu_y(256b)$, $e_w^{wu}(8\mu J, 32\mu J, 128\mu J, 512\mu J)$].

fixed-size data condition. It shows that the first two moments of data size distribution (mean and variance) alone are not sufficient to capture the dynamics of the power-aware fixed-size buffering. The term $\mu_y^2 k^*$ in (14) represents the impact of varying-size data on the power-aware fixed-size buffering scheme, which is orthogonal to the data arrival rate λ . Such an impact can be quantitatively isolated in the form as:

$$\Delta_v b = \sqrt{\frac{2e_w^{wu} b_{size} \lambda \mu_y}{p_m^{idle}} + \mu_y^2 k^*} - \sqrt{\frac{2e_w^{wu} b_{size} \lambda \mu_y}{p_m^{idle}}}, \quad (23)$$

where $\Delta_v b$ refers to the purely size variation-induced impact on the *fixed-size* buffering scheme. Examination of k^* , at least in principle, can elucidate the respective roles of skewness and coefficient of variation in determining the optimal buffer size b^* . The effect of size variability could be either mitigated or augmented by the skewness in size distribution. A positive skewness alleviates the impact of size variability. In contrast, a negative skewness strengthens the impact of size variability. In this case, k^* is positive and grows polynomially with $c_v(y)$, thereby ensuring $\Delta_v b > 0$. This requires an additional buffer size be allocated in order to accommodate the variability in the data size distribution.

Figure 3 plots the optimal buffer size (b^*) as a function of data arrival interval ($1/\lambda$) when the data size is exponentially distributed with a mean value of 256 bytes. The curves are plotted in a semilog format: the y -axis refers to the optimal buffer size b^* in a \log scale and the x -axis refers to the mean data arrival time $1/\lambda$. It shows that an increase in per radio wakeup energy or in data arrival rate (decreasing data arrival interval) demands a large buffer size to reduce the amortized per radio wakeup cost. This observation agrees with intuition. Combining (14) and (20) gives the overall power consumption of the

fixed-size buffering as follows.

$$\overline{e^{FS}(b^*)} = \underbrace{\frac{p_m^{idle} b^*}{b_{size}} + \frac{p_m^{idle} \mu_y}{2b_{size}} (c_v^2(y) - 1) + \lambda (2e_m^{resyn} + e_w^{wu} + \mu_y (e_m^w + e_m^r))}_{\text{buffering}} + \underbrace{2\lambda \mu_y e_w^{TX}}_{\text{trans/rec}}. \quad (24)$$

Equation (24) yields some interesting observations: the power consumption composition can be roughly divided into two pieces: 1) data transmission/reception power consumption is linearly proportional to bandwidth, that is, $\lambda \mu_y$. 2) data buffering power consumption is quite complicated, hence resists a straightforward explanation: the buffering power consumption not only relies on data arrival rate λ but also depends on the first three moments of the size distribution. Equation (24) shows explicitly that the data buffering power consumption grows asymptotically in proportion to both $(c_v^2(y) - 1)$ and the arrival rate λ , implying that the low-variance data size distribution ($c_v(y) < 1$) consumes less power than the high-variance data size ($c_v(y) > 1$).

3.2 Fixed-Interval Buffering Scheme

In this subsection we study the power-aware fixed-interval buffering scheme, which differs from its power-aware fixed-size counterpart. The following theorem gives a direct relation among the optimal time interval T^* , the power parameter of radio and memory bank, and data rate and the mean size of the data distribution.

THEOREM 4. *Let λ be a Poisson arrival rate, μ_y be the mean size of the data distribution, e_w^{wu} be the per radio wakeup energy, and p_m^{idle} be the idle state power consumption of a memory bank. Then, the optimal interval T^* for the fixed-interval buffering scheme is*

$$T^* = \sqrt{\frac{2e_w^{wu} b_{size}}{p_m^{idle} \lambda \mu_y}}. \quad (25)$$

PROOF. Let T be the interval of the fixed-interval buffering scheme. The fixed-interval buffering is a special case of the renewal process in which the renewal cycle is constant. Hence the energy consumed in a renewal cycle is expressed as

$$\begin{aligned} e^{FI}(T) &= e_w^{wu} + p_m^{idle} \sum_{i=1}^{n(T)} \frac{(T - s_i) y_i}{b_{size}} + \sum_{i=1}^{n(T)} e_w^{TX} y_i + \sum_{i=1}^{n(T)} (e_w^{RX} y_i + e_w^{wu}) \\ &\quad + \sum_{i=1}^{n(T)} (e_m^w + e_m^r) y_i + 2e_m^{resyn}, \end{aligned} \quad (26)$$

where $n(T)$ is a random variable denoting the number of data arrivals within the interval T , y_i is the size of the i th arrived data, and the arrival time $s_i = \sum_{j=1}^i x_j - x_{j-1}$. The term $\sum_{i=1}^{n(T)} (e_w^{RX} y_i + e_w^{wu})$ refers to the total reception energy in the interval T , which involves the energy consumed in receiving the

arrived data $\sum_{i=1}^{n(T)} e_w^{RX} y_i$, and the energy of radio wakeup for data reception $n(T)e_w^{wu}$. Notice that the radio-triggered power scheme does not incur listening power consumption. By the renewal reward theory, the long-run mean average energy consumption is

$$\overline{e^{FI(T)}} \stackrel{\text{def}}{=} \lim_{t \rightarrow \infty} \frac{e^{FI(t)}}{t} = \frac{E[e^{FI(T)}]}{T}. \quad (27)$$

By Wald's equation, the expectation of the second term in (26) is

$$\begin{aligned} \frac{p_m^{idle}}{b_{size}} E \left[\sum_{i=1}^{n(T)} (T - s_i) I_{\{n(T) > 0\}} y_i \right] &= \frac{p_m^{idle} \mu_y}{b_{size}} \int_0^T (T - t) \lambda e^{-\lambda t} \left(\sum_{i=1}^{\infty} \frac{(\lambda t)^{i-1}}{(i-1)!} \right) dt \\ &= \frac{p_m^{idle} \mu_y \lambda}{b_{size}} \int_0^T (T - t) dt = \frac{p_m^{idle} \mu_y \lambda T^2}{2b_{size}}, \end{aligned} \quad (28)$$

where $I_{\{n(T) > 0\}}$ is the indicator function. Letting $e_w^{TX} \approx w_w^{RX}$, the expectation of the third-sixth terms in (26) are simplified as

$$\begin{aligned} E \left[\sum_{i=1}^{n(T)} y_i (2e_w^{TX} + e_m^w + e_m^r) + 2e_m^{resyn} + e_w^{wu} \right] \\ = E[n(T)] E \left[y_i (2e_w^{TX} + e_m^w + e_m^r) + 2e_m^{resyn} + e_w^{wu} \right] \\ = \lambda T \left(\mu_y (2e_w^{TX} + e_m^w + e_m^r) + 2e_m^{resyn} + w_w^{wu} \right). \end{aligned} \quad (29)$$

Combining (27)–(29) gives

$$\overline{e^{FI(T)}} = \frac{E[\xi_1(T)]}{T} = \frac{e_w^{wu}}{T} + \frac{\lambda T p_m^{idle} \mu_y}{2b_{size}} + \lambda \left(\mu_y (2e_w^{TX} + e_m^w + e_m^r) + 2e_m^{resyn} + e_w^{wu} \right) \quad (30)$$

Taking derivative of (30) *w.r.t.* T gives $\frac{\partial \overline{e^{FI(T)}}}{\partial T} = -\frac{e_w^{wu}}{T^2} + \frac{p_m^{idle} \lambda \mu_y}{2b_{size}}$. Resolving $\frac{\partial \overline{e^{FI(T^*)}}}{\partial T^*} = 0$ leads to (25). \square

Examination of T^* in (25) reveals the apparent variability immunity of the fixed-interval buffering scheme since T^* only contains the first moment μ_y of the size distribution. Substituting (25) into (30) gives

$$\overline{e^{FI(T^*)}} = \underbrace{\sqrt{\frac{2p_m^{idle} e_w^{wu} \lambda \mu_y}{b_{size}}}}_{\text{buffering}} + \lambda (2e_m^{resyn} + e_w^{wu} + \mu_y (e_m^w + e_m^r)) + \underbrace{\lambda 2\mu_y e_w^{TX}}_{\text{trans/rec}} \quad (31)$$

In a similar fashion, the power consumption composition in (31) also can be divided into the data transmission/reception and buffering pieces. The data transmission/reception piece is linearly proportional to the bandwidth ($\lambda \mu_y$), while the data buffer one is proportional to the square root of the bandwidth $\sqrt{\lambda \mu_y}$. Although there is very little apparent relationship between the fixed-size and the fixed-interval buffering schemes, both buffering schemes essentially share the same transmission/reception component but differ markedly in their

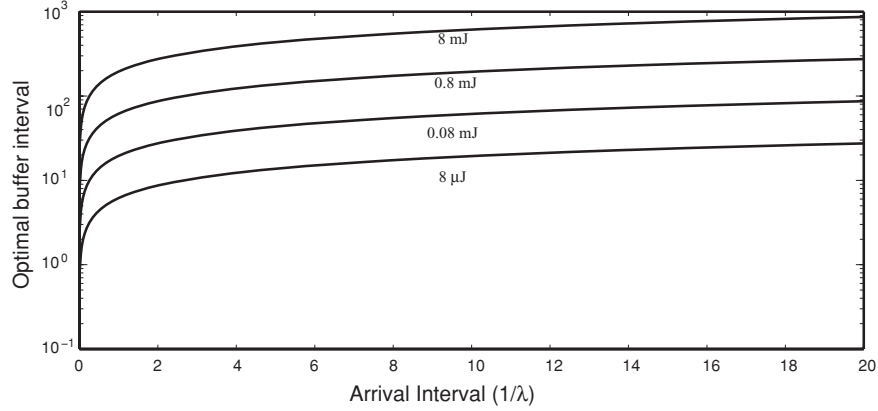


Fig. 4. Optimal buffer interval T^* vs. $1/\lambda$: [$p_m^{idle}(0.409\mu W)$, $b_{size}(256b)$, $\mu_y(256b)$, $e_w^{wu}(8\mu J, 0.08mJ, 0.8mJ, 8mJ)$].

data buffering components: the data buffering component of the fixed-interval buffering scheme is a function of bandwidth. By contrast, that of the fixed-size buffering scheme is linked to the bandwidth and the first three moments of size distribution explicitly expressed in term $\mu_y^2 k^*$.

The curves in a semilog format in Figure 4 show that radio wakeup energy increase results in optimal time interval increase, while increasing idle-mode power consumption in a memory bank reduces the optimal buffer interval. This can be explained intuitively as follows: for a high per radio wakeup energy, a large data buffer (large optimal interval) can effectively reduce the amortized per radio wakeup energy, while a high sleep-mode power consumption would increase the power consumption of buffering, thereby reducing optimal buffer interval T^* .

Let us digress a little bit from the main derivation to examine the no-buffer scheme: a special case of the fixed-interval buffering scheme in which the sensor node transmits data immediately upon receipt of measured data. Mathematically, this corresponds to a case where the mean buffer interval $T = \frac{1}{\lambda}$. The following corollary deals with the no-buffer scheme.

COROLLARY 4. *The long-run mean average energy consumption of the no-buffer scheme, denoted by $e(nb)$, is*

$$\overline{e(nb)} = \lambda(2e_w^{wu} + \mu_y(2e_w^{TX} + e_m^w + e^r) + 2e_m^{resyn}) \quad (32)$$

PROOF. Consider a renewal reward process with the cycle length (L_c) being equal to the data arrival interval $T = 1/\lambda$. Thus the energy consumed in this cycle is

$$e(nb) = e_w^{wu} + (2e_w^{TX} + e_m^w + e_m^r)y_i + 2e_m^{resyn} + e_w^{wu}, \quad (33)$$

where y_i is the size of i th arrived data. (33) is simply attained by removing the buffering factors (terms) in (28). Using the same argument in proving Theorem 4 we get $\overline{e(nb)} = \frac{E[e(nb)]}{\frac{1}{\lambda}}$ in (32) since $E[L_c] = \frac{1}{\lambda}$. \square

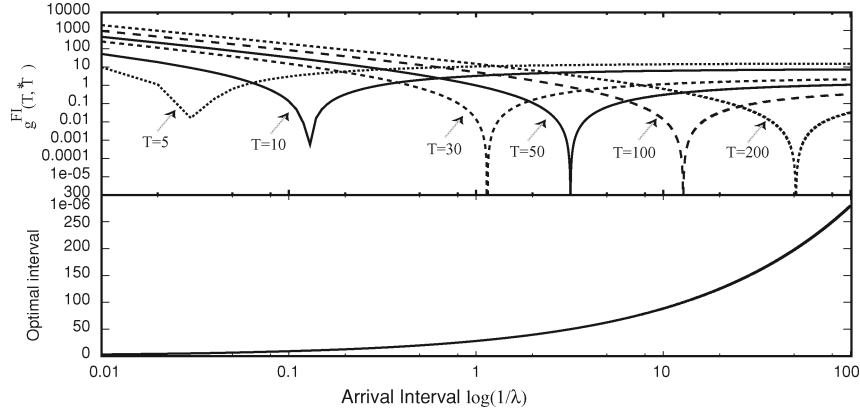


Fig. 5. Top: $g^{FI}(T, T^*)$ vs. arrival interval $\log(1/\lambda)$. Bottom: T^* vs. arrival interval $\log(1/\lambda)$: $[P_m^{idle}(0.409\mu W), e_m^{wu}(80\mu J), b_{size}(128b), \mu_y(64b)]$.

Define a function $g^{FI}(T, T^*)$ to quantify the differential gain of the optimal power-aware fixed-interval buffering over a power-oblivious buffering scheme.

$$g^{FI}(T, T^*) = \overline{e^{FI}(T)} - \overline{e^{FI}(T^*)} = \frac{e_w^{wu}}{T} + \frac{T P_m^{idle} \lambda \mu_y}{2 b_{size}} - \sqrt{\frac{2 P_m^{idle} e_w^{wu} \lambda \mu_y}{b_{size}}}, \quad (34)$$

where T^* is the optimal interval and T is chosen arbitrarily.

Results of $g^{FI}(T, T^*)$ are plotted in Figure 5. The main trend is that the optimal buffer interval T^* grows as the square root of $\frac{1}{\lambda}$ (see bottom graph), and that a dip in each curve occurs when arbitrarily chosen T happens to be in the vicinity of T^* . The differential gain $g^{FI}(T, T^*)$ arises sharply when T deviates from the optimal buffer interval T^* . This implies that a blind selection of buffer interval T is very likely to incur an excessive energy consumption.

4. PERFORMANCE COMPARISON

In this section we attempt to answer two fundamental questions. 1) How much power saving via power-aware buffering can be achieved in comparison to the no-buffer scheme? 2) The fixed-size buffering or its fixed-interval counterpart, which one performs better?

4.1 Comparison between the No-Buffer Scheme and Power-Aware Buffering Schemes

It is evident that the fixed-interval buffering always outperforms the no-buffer scheme as the latter is a special case of the former. Below we compare the no-buffer scheme with the fixed-size buffer one.

The differential power consumption between the no-buffer and optimal fixed-size buffering schemes is expressed as

$$\overline{e^{nb}} - \overline{e^{FS}(b^*)} = \lambda e_w^{wu} - \frac{P_m^{idle}}{b_{size}} \left(b^* + \frac{\mu_y (c_v^2(y) - 1)}{2} \right) > 0 \quad (35)$$

Equation (35) does in fact constitute an *incentive condition* under which the optimal fixed-size buffering scheme outperforms the no-buffer scheme in power conservation. It shows that increasing variability $c_v(y)$ in effect erodes the gain brought out by the power-aware fixed-size buffering scheme, hence shrinks the incentive area. A positive skewness in size distribution can neutralize, to some extent, the size variability-induced impact. While in general this incentive condition could be profoundly affected by various intertwined and correlated factors, we explicitly derive closed-form expressions under some restricted scenarios.

- (1) Exponential data size distribution y with a mean of μ_y . Under this condition, k^* is reduced to zero according to (4), the incentive condition thus becomes

$$\overline{e(nb)} - \overline{e^{FS}(b^*)} = \lambda e_w^{wu} - \frac{p_m^{idle} b^*}{b_{size}} = \lambda e_w^{wu} - \sqrt{\frac{2\lambda p_m^{idle} e_w^{wu} \mu_y}{b_{size}}} > 0 \quad (36)$$

It is obvious that (36) holds as long as $\lambda e_w^{wu} > \frac{2p_m^{idle} \mu_y}{b_{size}}$ is met. This incentive condition can be rewritten in a structurally meaningful form that emphasizes the distinction between hardware parameters and operational requirement as follow

$$\frac{\frac{e_w^{wu}}{p_m^{idle}}}{b_{size}} > \frac{2\mu_y}{\lambda}. \quad (37)$$

Using byte-second as a quantifiable unit, the left-hand side of (37) is related to hardware power parameters: the ratio of radio wakeup energy to the per-byte idle-mode power consumption of a memory bank. The right-hand side, on the other hand, is related to the operational requirement: the ratio of the mean data size to the data arrival rate. For given power parameters and μ_y , there exists a critical value for $\lambda_c = \frac{2\mu_y p_m^{idle}}{b_{size} e_w^{wu}}$. When $\lambda > \lambda_c$, the fixed-size buffering scheme is preferred. Otherwise, the no-buffer scheme is preferred. A high ratio of the per radio wakeup energy to the per-byte idle-mode memory power consumption favors a large buffer size. On the other hand, the benefit of data buffering is diminished as $\frac{\mu_y}{\lambda}$ increases.

- (2) Erlang size distribution y with parameters (α, λ_α) . This corresponds to the case in which $c_v(y) = \frac{1}{\sqrt{\alpha}}$, $\gamma(y) = \frac{2}{\sqrt{\alpha}}$, $\mu_y = \frac{\alpha}{\lambda_\alpha}$, $\sigma_y = \frac{\sqrt{\alpha}}{\lambda_\alpha}$, then $k^* = \frac{1}{12}(1 - \frac{1}{\alpha^2})$. The incentive condition is thus expressed as

$$\overline{e(nb)} - \overline{e^{FS}(b^*)} = \lambda e_w^{wu} - \frac{p_m^{idle}}{b_{size}} \left(b^* - \frac{\mu_y}{2} \left(1 - \frac{1}{\alpha} \right) \right) > 0, \quad (38)$$

where

$$b^* = \sqrt{\frac{2\lambda b_{size} e_w^{wu} \mu_y}{p_m^{idle}} + \frac{\mu_y^2}{12} \left(1 - \frac{1}{\alpha^2} \right)}. \quad (39)$$

Equation (39) shows that, as compared with an exponential size distribution ($\alpha = 1$), an additional buffer size needs to be allocated when the shape

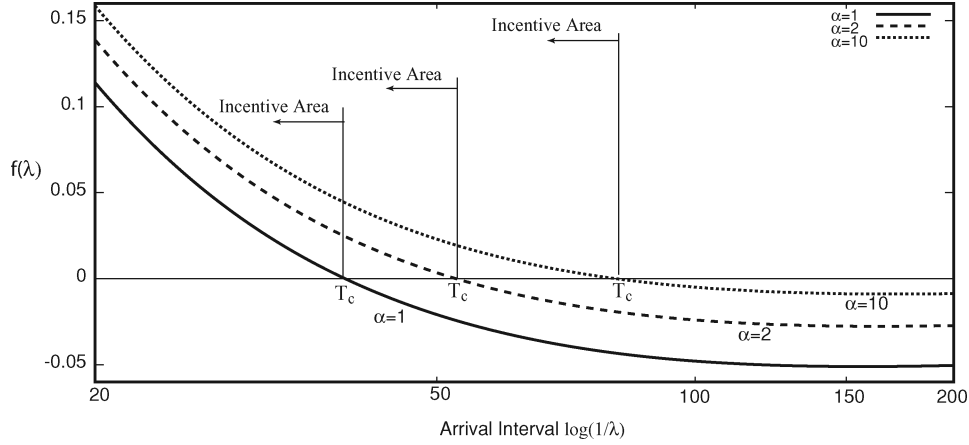


Fig. 6. Incentive area vs. arrival interval: [$p_m^{idle}(0.409\mu W)$, $b_{size}(256b)$, $e_m^{wu}(80\mu J)$, $\mu_y(64b)$].

parameter $\alpha > 1$. To study the impact of data arrival rate λ , we define $f(\lambda) = \bar{e}(nb) - \bar{e}^{FS}(b^*)$. Differentiating $f(\lambda)$ and solving $f'(\lambda^*) = 0$ gives

$$\lambda^* = \frac{\mu_y p_m^{idle}}{24b_{size} e_w^{wu}} \left(11 + \frac{1}{\alpha^2} \right). \quad (40)$$

Observe that $f(\lambda)$ has a global minimum point at λ^* since $f''(\lambda^*) > 0$. This implies that $f(\lambda^*) < f(\lambda)$, $\lambda \in \mathcal{R}^+$ and $\lambda \neq \lambda^*$. Substituting (40) into (38) gives

$$f(\lambda^*) = -\frac{\mu_y p_m^{idle}}{24b_{size}} \left(1 + \frac{12}{\alpha} - \frac{1}{\alpha^2} \right) < 0, \quad \alpha \geq 1 \quad (41)$$

Since $f(0) = \frac{p_m^{idle} \mu_y}{2b_{size}} \left(1 - \frac{1}{\alpha} - \frac{1}{\sqrt{3}} \sqrt{1 - \frac{1}{\alpha^2}} \right) \leq 0$, $\alpha \geq 1$, and $f(\lambda) \rightarrow e_w^{wu} \lambda > 0$ when λ is sufficiently large, one concludes that there exists a critical data rate $\lambda_c > \lambda^*$ such that the no-buffer scheme outperforms the fixed-size buffering scheme when $\lambda \leq \lambda_c$ and the fixed-size buffering scheme is preferred when $\lambda > \lambda_c$. Figure 6 plots $f(\lambda)$ as a function of $1/\lambda$ with different shape parameters in a semilog format. Figure 6 illustrates that increasing shape parameter α (less variation) results in a decreased critical data rate λ_c (increasing $T_c = 1/\lambda_c$), indicating that the smaller the data size variation, the larger the incentive region. It is worth noting that there exists an inherent tradeoff between buffering and responsiveness: the no-buffer scheme achieves real-time responsiveness at the expense of power consumption, while the power-aware buffering to some extent can save power consumption, but at the price of reduced responsiveness.

4.2 Comparison between Fixed-Size and Fixed-Interval Buffering Schemes

One question arises naturally: which buffering scheme is more power efficient? the power-aware fixed-size scheme or the power-aware fixed-interval one. While in general there is no simple answer to this question, there is a definite answer under some special circumstances. We begin with an easy lemma as follows.

LEMMA 1. Let $f(x) = \sqrt{ax} - \sqrt{ax+b} + c$, $x \geq 0$, where a , b , and c are positive, and $\sqrt{b} < c$. Then $f(x) > 0$ for $x \geq 0$.

PROOF. Take derivation of $f(x)$, we obtain

$$f'(x) = \frac{a(\sqrt{ax+b} - \sqrt{ax})}{2\sqrt{ax(ax+b)}} > 0, x > 0 \quad (42)$$

This means that $f(x)$ is monotonically increasing for $x \geq 0$. Then $f(0) = \min_{x \in (0, \infty)} f(x)$. It can be inferred that $f(x) > f(0) > 0$ for $x \geq 0$ since $f(0) = -\sqrt{b} + c > 0$. \square

THEOREM 5. The power-aware fixed-size buffering scheme is more power-efficient than the fixed-interval counterpart when the data size is constant, while both the buffering schemes perform equally well when the data size is exponentially distributed.

PROOF. Define $g(T^*, b^*)$ to denote the power consumption differential between the fixed-interval and fixed-size buffering schemes as follows

$$g(T^*, b^*) = e^{\overline{FI}(T^*)} - e^{\overline{FS}(b^*)} = \sqrt{\frac{2\mu_y \lambda p_m^{\text{idle}} e_w^{wu}}{b_{\text{size}}}} - \frac{p_m^{\text{idle}}}{b_{\text{size}}} \left(b^* + \frac{\mu_y (c_v^2(y) - 1)}{2} \right) \quad (43)$$

(I) Constant data size: Since $\sigma_y = 0$ and $k^* = 1/12$, thus (43) becomes

$$g(T^*, b^*) = \sqrt{\frac{2\mu_y \lambda p_m^{\text{idle}} e_w^{wu}}{b_{\text{size}}}} - \sqrt{\frac{2\mu_y \lambda p_m^{\text{idle}} e_w^{wu}}{b_{\text{size}}} + \frac{(\mu_y p_m^{\text{idle}})^2}{12b_{\text{size}}^2}} + \frac{p_m^{\text{idle}} \mu_y}{2b_{\text{size}}} \quad (44)$$

Let $a = \frac{2\mu_y \lambda p_m^{\text{idle}} e_w^{wu}}{b_{\text{size}}}$, $b = \frac{(\mu_y p_m^{\text{idle}})^2}{12b_{\text{size}}^2}$, $c = \frac{p_m^{\text{idle}} \mu_y}{2b_{\text{size}}}$. Based on Lemma 1, we have $g(T^*, b^*) > 0$, i.e., $e^{\overline{FI}(T^*)} > e^{\overline{FS}(b^*)}$.

(II) Exponential size distribution: Since $\sigma_y = \mu_y$, $c_v(y) = 1$, $\gamma(y) = 2$, thus $k^* = 0$, (43) becomes

$$g(T^*, b^*) = \sqrt{\frac{2\mu_y \lambda p_m^{\text{idle}} e_w^{wu}}{b_{\text{size}}}} - \sqrt{\frac{2\mu_y \lambda p_m^{\text{idle}} e_w^{wu}}{b_{\text{size}}}} = 0. \quad (45)$$

Combining (I) and (II) completes the proof. \square

Consider a hyper-exponential size distribution: $f_h(y) = \sum_{i=1}^2 \frac{1}{\mu_i} p_i \exp(-y/\mu_i)$ and $\sum_{i=1}^2 p_i = 1$. Letting $p_1 = p$, $p_2 = 1 - p$. Since there is no closed-form expression for $g(T^*, b^*)$, thus a numerical method is used to compute different values for $c_v(y)$ and $\gamma(y)$. This is achieved by varying the value of p while maintaining μ_y constant. Figure 7 presents $g(T^*, b^*)$ as a function of $\frac{1}{\lambda}$ with different $c_v(y)$ and $\gamma(y)$. It shows that $c_v(y)$ has a substantial impact on $g(T^*, b^*)$, whereas $\gamma(y)$ plays a marginal role. For example, when $c_v(y) = 1.72$ and $\gamma(y) = 2.72$, $g(T^*, b^*) = e^{\overline{FI}(T^*)} - e^{\overline{FS}(b^*)} < 0$.

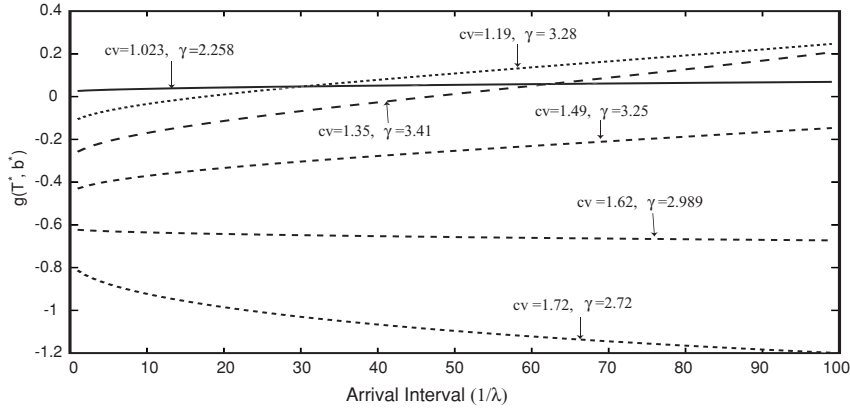


Fig. 7. $g(T^*, b^*)$ vs. $1/\lambda$: [$p_m^{idle}(0.409\mu W)$, $e_m^{wu}(80\mu J)$, $b_{size}(256b)$].

Combining the above analysis and numerical calculation leads to the conclusion that the size variation-induced effect is a non-negligible role in determining the relative advantages of the fixed-interval and fixed-size buffering schemes: when the size distribution is of low-variance, the fixed-size buffering scheme outperforms the fixed-interval counterpart. When the size distribution is of high-variance, the fixed-interval buffering scheme is more energy-efficient than the fixed-size one. Between these two extremes in size variability, the relative power efficiency of these two buffering schemes depends on data arrival rate λ .

To illustrate the efficacy of the power-aware fixed-size buffering scheme over a power-oblivious one, we define a function $g^{FS}(b, b^*)$ as follows

$$g^{FS}(b, b^*) = \overline{e^{FS}(b)} - \overline{e^{FS}(b^*)} = (b^* - b) \left(\frac{\lambda e^{wu} \mu_y + \frac{p_m^{idle} \mu_y^2 k^*}{2b_{size}}}{b \cdot b^*} - \frac{p_m^{idle}}{2b_{size}} \right), \quad (46)$$

where b^* denotes the optimal buffer size (see (14)), and b is arbitrarily chosen ($b \in \mathcal{R}$ and $b \neq b^*$). In practical terms, the amount of power consumed in the idle-mode memory banks is quantized into discrete levels as $p_m^{idle} \lceil \frac{b}{b_{size}} \rceil$, where $\lceil \cdot \rceil$ is the ceiling function. This implies that only certain discrete power states are allowed. For example, if $b_{size} = 128$, then two memory banks will be used when the buffer size b falls within the range of $[128, 256]$.

To visualize this quantization impact, we examine the function $g^{FS}(b, b^*)$ when $k^* = 0$ (an exponential size distribution). Figures 8–9 plot $g^{FS}(b, b^*)$ as a function of $1/\lambda$ under different values of μ (mean data size). The y-axis in the top graph in Figure 8 represents $g^{FS}(256, b^*)$ in the unit of (μW) . The y-axis in the bottom graph in Figure 8 denotes the optimal buffer size b^* in a multiple of b_{size} . The x-axis refers to the mean arrival interval $1/\lambda$ in a log scale.

With increases in $\frac{1}{\lambda}$, the optimal buffer size b^* decreases. The curve of $g^{FS}(256, b^*)$ gradually declines as $\frac{1}{\lambda}$ increases, as shown in Figure 8. In

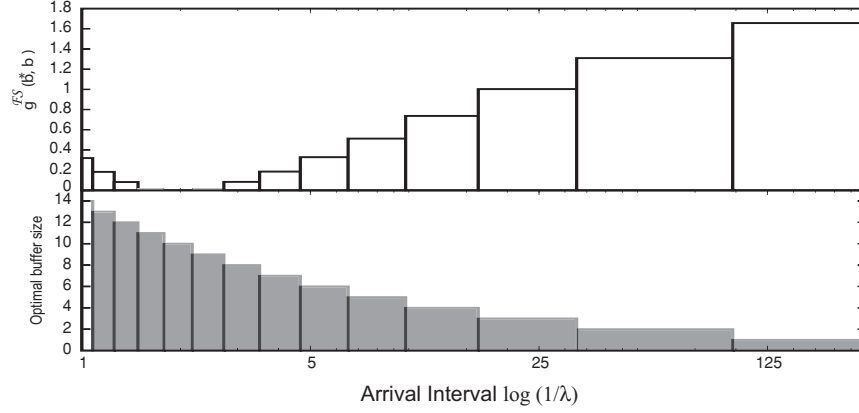


Fig. 8. Top: $g^{FS}(256, b^*)$ vs arrival rate $\log(1/\lambda)$ Bottom: optimal buffer size b^* vs. $\log(1/\lambda)$ [$b_{size}(128b)$, $\mu_y(64b)$, $e_w^{wu}(80\mu J)$, $p_m^{idle} = 0.409\mu W$].

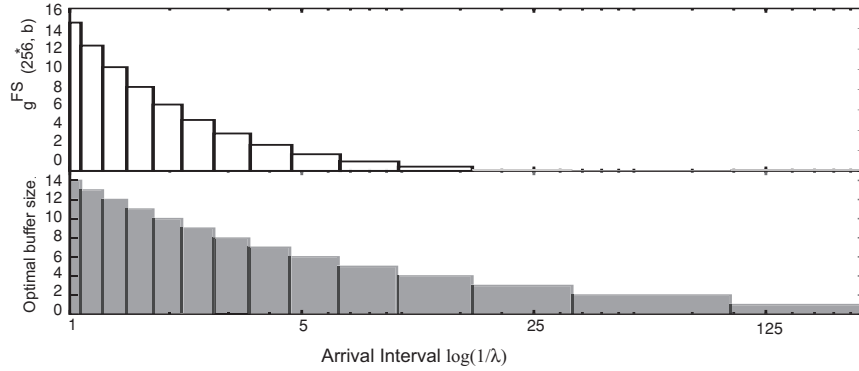


Fig. 9. Top: $g^{FS}(1280, b^*)$ vs. $\log(1/\lambda)$. Bottom: optimal buffer size b^* vs. $\log(1/\lambda)$. [$b_{size}(128b)$, $\mu_y(64b)$, $e_w^{wu}(80\mu J)$, $p_m^{idle}(0.409\mu W)$].

contrast, in Figure 9 with increases in $\frac{1}{\lambda}$, the differential gain $g^{FS}(1280, b^*)$ initially monotonically decreases and then increases after reaching the lowest point at which the buffer size b is optimal at $10b_{size} = 1280$. Observe that the memory-bank-size quantized effect produces a stair-like relation between the amount of power consumed and $\frac{1}{\lambda}$. For example, the optimal buffer size is $6b_{size}$ when $\frac{1}{\lambda} \in [5.435, 7.825)$. The larger the arrival interval $\frac{1}{\lambda}$, the more pronounced (a wider stair space) the quantized effect.

To evaluate the size variation-induced effects, we calculated $g^{FS}(b, b^*)$ under a hyperexponentially distributed data size in which $c_v(y) = 1.732$ and $\gamma(y) = 2.718$ and plot both $g^{FS}(256, b^*)$ and $g^{FS}(1280, b^*)$ as a function of $\frac{1}{\lambda}$ in Figures 10–11, respectively. In comparison to the absence of size variation as shown in Figures 8–9, the size variation effect becomes inconsequential when λ is high, but has a substantial impact when λ is low (see Figure 12).

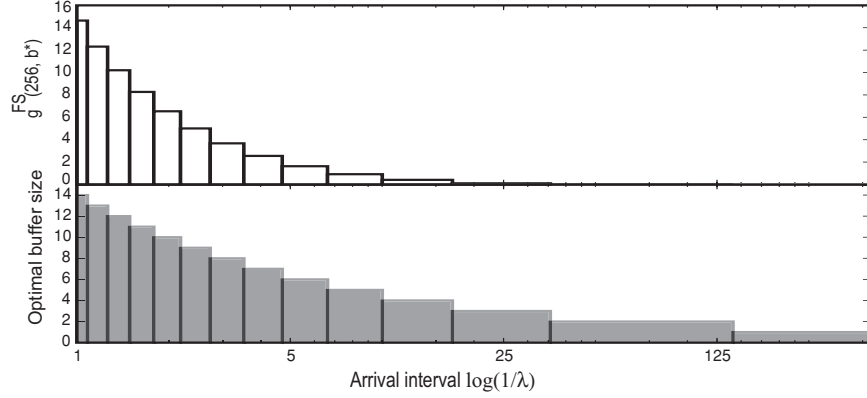


Fig. 10. Top: $g^{FS}(256, b^*)$ vs. $\log(1/\lambda)$ Bottom: optimal buffer size b^* vs. $\log(1/\lambda)$ [b_{size} (128b), μ_y (64b), e_w^{wu} (80 μ J), p_m^{idle} (0.409 μ W), $c_v(y)$ (1.723), $\gamma(y)$ (2.718)].

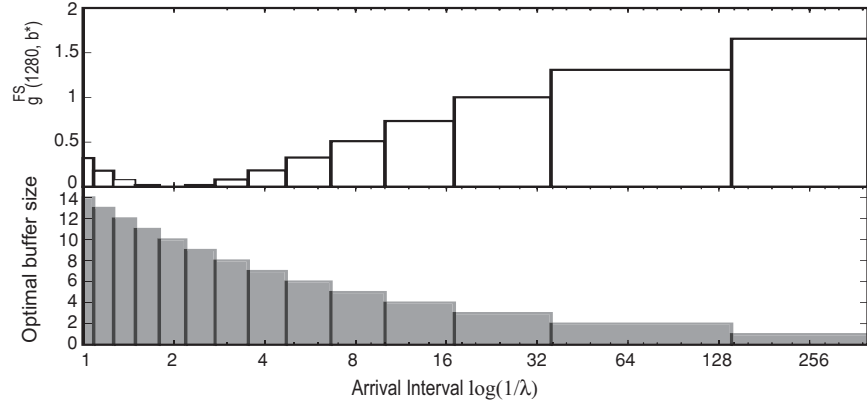


Fig. 11. Top: $g^{FS}(1280, b^*)$ vs. $\log(1/\lambda)$ Bottom: optimal buffer size b^* vs. $\log(1/\lambda)$ [b_{size} (128b), μ_y (64b), e_w^{wu} (80 μ J), p_m^{idle} (0.409 μ W), $c_v(y)$ (1.723), $\gamma(y)$ (2.718)].

This phenomenon can be explained by reference to (23). $\Delta_v b$ in (23) shows that the size variation effect is proportional to the term $\mu_y^2 k^*$. Hence the relative effect of size variation on the optimal buffer size b^* can be expressed as

$$\frac{\Delta_v b}{\sqrt{\frac{2\lambda e^{wu} w b_{size} \mu_y}{p_m^{idle}}}} = \sqrt{1 + \frac{\mu_y p_m^{idle} k^*}{2\lambda e^{wu} b_{size}}} - 1. \quad (47)$$

Equation (47) indicates that for a given size distribution (given $c_v(y)$ and $\gamma(y)$, thereby k^*), the size variability effect becomes prominent for low duty-cycle sensor nodes as (47) increases with decreasing λ (low duty cycle).

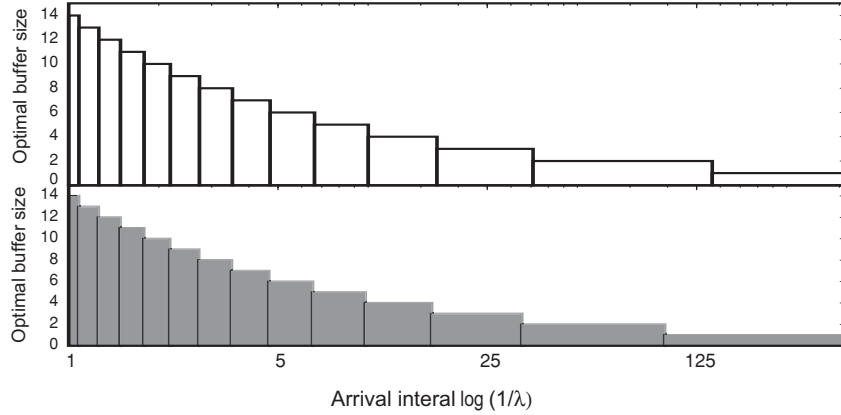


Fig. 12. Top: Optimal buffer size b^* vs. $\log(\frac{1}{\lambda})$: $(c_v(y)(1.723), \gamma(y)(2.718))$. Bottom: Optimal buffer size b^* vs. $\log(\frac{1}{\lambda})$: $(c_v(y)(0)) [b_{size}(128b), \mu_y(64b), e_w^{wu}[80\mu J], p_m^{idle}(0.409\mu W)]$.

Table II. Power Parameters of Radio and Memory Bank

e_w^{wu}	p_m^{idle}	e_w^{TX}	$e_m^r + e_m^w$	e_m^{resyn}
$80 \mu J$	$0.409 \mu W$	$8.976 \mu J/byte$	$36 \cdot 10^{-3} \mu J/byte$	$0.912 \mu J$

5. EFFECT OF POWER-AWARE BUFFERING SCHEMES ON LIFE SPAN

In this section we will use a concrete example to quantify the benefits of power-aware schemes in terms of the lifespan extension.

Power parameters in Table II are either directly obtained or indirectly derived from the literature. In CC2420-802.15.4 radio specification [CC2420 2004], the transmission power is $-25 dBm$ with a data rate of 250 kbps, and the current draw is $8.5 mA(3.3V)$. The 250 kbps is the optimal rate in an ideal environment, which may not make any practical sense. In practical terms, the rate is assumed to be 25 kbps. Consequently, the one-byte transmission energy is calculated as $e_w^{TX} = \frac{8.5 \times 3.3 \times 8}{25} = 8.976 \mu J/byte$. The power consumption for an idle-mode SRAM memory bank is $p_m^{idle} = 0.409 \mu W$ [Hempstead et al. 2005]. Due to unavailability of actual power data of SRAM bank in the literature, the $e_m^r, e_m^w, e_m^{resyn}$ are approximated by using Rambus DRAM power data. A read or write operation on Rambus DRAM takes $60ns$ and consumes $300mW$, that is, $e_m^r = e_m^w = 18 \cdot 10^{-3} \mu J/byte$. A transition from the powerdown mode of a Rambus DRAM to the active mode consumes $152mW$ and takes $6000 ns$ [Fan et al. 2001]. Thus a resynchronization energy is $e_m^{resyn} = 0.912 \mu J$. A radio wakeup energy is assumed to be $e_w^{wu} = 80 \mu J$ as it is not found in literature. Let the supply voltage be $3.3 V$, and the lifespan of two AA batteries be $2700 mAh$ [Levis 2005].

Consider a scenario with a Poisson arrival and constant data size (implying $k^* = 1/12$). Letting $y_i = \mu_y = b_{size} = 128b$. Based on Table II, the comparison results between the optimal fixed-size buffering, optimal fixed-interval buffering, and an power-oblivious buffering with buffer size of 256 are tabulated in Table III.

Table III. Performance Comparison

λ (1/s)	$e^{FS}(b^*)$ (μW)	lifespan (yr)	$e^{FI}(T^*)$ (μW)	lifespan (yr)	$e^{FS}(256)$ (μW)	lifespan (yr)
1	2392.1739	0.4252	2392.3775	0.4252	2424.5010	0.4195
0.9	2153.3299	0.4723	2153.5336	0.4723	2182.0722	0.4661
0.8	1914.4623	0.5313	1914.6659	0.5312	1939.6434	0.5244
0.7	1675.5663	0.6070	1675.7698	0.6069	1697.2146	0.5993
0.6	1436.6355	0.7080	1436.8389	0.7079	1454.7858	0.6992
0.5	1197.6609	0.8493	1197.8641	0.8491	1212.3570	0.8390
0.4	958.6283	1.0610	958.8314	1.0608	969.9282	1.0487
0.3	719.5143	1.4136	719.7172	1.4132	727.4994	1.3981
0.2	480.2728	2.1178	480.4753	2.1169	485.0706	2.0969
0.1	240.7851	4.2242	240.9869	4.2207	242.6418	4.1919

With $\lambda = 0.5$, by (14), the optimal buffer size is calculated as $b^* = 1790.54(b)$, thus the number of memory banks involved is $\lceil \frac{b^*}{b_{size}} \rceil = 14$. By (24), we obtain $\overline{e^{FS}(b^*)} = 1197.660858 \mu W$. The amount of current draw is $\frac{1197.660858}{3.3} = 362.927533 \mu A$, and the lifespan is $\frac{3.3 \times 2700000}{1197.660858 \times 24 \times 365} = 0.849258(\text{yr})$. Similarly, the power consumption of the power-oblivious and optimal fixed-interval schemes are calculated as $\overline{e^{FS}(256)} = 1212.357021 \mu W$ and $\overline{e^{FI}(T^*)} = 1197.864140 \mu W$. This implies that the power-aware fixed-size scheme outlives the power-oblivious one by 3.75761 days and outlives the optimal fixed-interval one by 1.262509 hours.

Table III shows that the role of the power-aware buffering becomes more prominent when the node operates at a low duty cycle. For example, with $\lambda = 1$, the optimal fixed-size buffer scheme outlives the power-oblivious one by 2.069278 days, when $\lambda = 0.1$, the optimal fixed-size buffer scheme outlives the power-oblivious one by 11.797951 days. A side-by-side comparison in Table III suggests that the fixed-size buffering scheme performs slightly better than the fixed-interval counterpart. However, this marginal advantage will be disappeared in the presence of data-size variations.

To see the size variation effect on the fixed-size buffering scheme, we consider a hypothetical symmetrical size distribution ($\gamma(y) = 0$). Thus the optimal buffer size b^* is expressed as

$$b^* = \sqrt{\frac{2\lambda e_m^{wu} \mu_y b_{size}}{p_m^{idle}} + \mu_y^2 \left(\frac{5c_v^4(y)}{4} + \frac{1}{12} \right)}. \quad (48)$$

Based on (24) and Table II, Figure 13 plots the lifespan as a function of $c_v(y)$ under different data arrival rates (λ). One can see a monotonic decline in the lifespan with increasing $c_v(y)$. This trend indicates that for the fixed-size buffering scheme, high data size variation has a detrimental effect that further depletes battery. By contrast, the fixed-interval buffering scheme has an obvious advantage of being eminently immune to data-size variation: its power consumption is only associated to mean data size μ_y , and is independent of data size variance. To illustrate, under two power parameter settings, the lifespan, optimal buffer interval T^* , and the power consumption under different rates are given in Table IV.

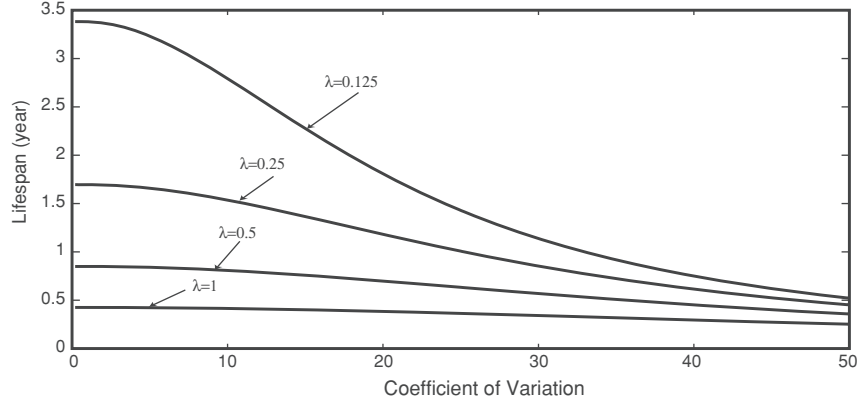


Fig. 13. Life span vs $c_v(y)$ [$b_{size}(128b)$, $\mu_y(128b)$, $e_w^{wu}(80\mu J)$, $p_m^{idle}(0.409\mu W)$].

Table IV. Lifespan under Data Arrival Rates for Fixed-Interval Buffering Scheme

$\lambda(1/s)$	$T^*(s)$	$e^{FI}(T^*)(\mu W)$	lifespan (yr)	$T^*(s)$	$e^{FI}(T^*)(\mu W)$	lifespan (yr)
1	19.79	2392.38	0.425	12.65	3230.78	0.315
0.5	13.99	4780.02	0.213	8.94	6387.46	0.159
0.25	9.89	9553.33	0.107	6.32	12670.13	0.08
0.125	6.99	19097.18	0.053	4.47	25192.07	0.04
0.0625	4.94	38180.97	0.027	3.16	50174.57	0.02
power parameters	$e_w^{wu}(80\mu J)$, $p_m^{idle}(0.409\mu W)$		$e_w^{wu}(800\mu J)$, $p_m^{idle}(10\mu W)$			
	$e_w^{TX}(8.976\mu J/byte)$		$e_w^{TX}(8.976\mu J/byte)$			
	$b_{size}(128b)$, $\mu_y(128b)$, $e_m^r + e_m^w(36 \cdot 10^{-3}\mu J/byte)$					

For environmental monitoring, a sensor-based network forms a data collection tree. Each node gathers local information, and forwards the data from its child nodes to its routing parent (see Figure 14). The sink node then collates the received information into global environmental data. Below we establish an energy consumption relationship between a routing parent node and its child nodes in the context of fixed-interval buffering scheme.

THEOREM 6. *Suppose a parent node has k child nodes in a static data collection tree. Let the i th child have Poisson arrival with a rate of λ_i and a mean size of μ_i . If the parent and child nodes adopt the fixed-interval buffering scheme, then the optimal buffer interval T_p^* for the parent node is*

$$T_p^* = \sqrt{\frac{2e_w^{wu}b_{size}}{p_m^{idle} \sum_{i=1}^k \lambda_i \mu_i}}}. \quad (49)$$

PROOF. It follows from (25) that for the i th child node, its optimal interval is $T_i = \sqrt{\frac{2e_w^{wu}b_{size}}{p_m^{idle}\lambda_i\mu_i}}$. This means that successive transmission from the i th node to its parent node is equally spaced by an interval T_i , with the mean size

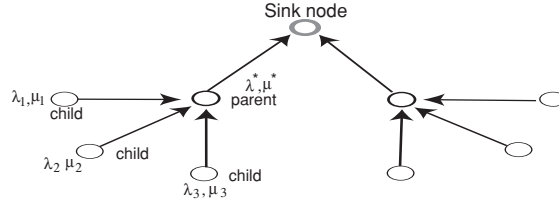


Fig. 14. Data collection tree: bandwidth conservation between parent and child nodes.

of $\lambda_i T_i \mu_i$. Assume that each node has the radio-triggered wakeup capability, thereby incurring no listening power consumption.

Let T be the length of a renewal cycle of the parent node. Each child node independently transmits the buffered data at the rate of $1/T_i$. With respect to the i th child, the corresponding energy consumed by the parent node, denoted by $e^{FI}(T)(i)$, can be decomposed into three pieces:

- (1) The energy for data buffering (idle-mode) at the parent node, denoted by $e_b^{FI}(T)(i)$, is bounded as:

$$\sum_{k=1}^{\lfloor \frac{T}{T_i} \rfloor} \frac{p_m^{idle}(T - kT_i)y_i(k)}{b_{size}} \leq e_b^{FI}(T) \leq \sum_{k=0}^{\lfloor \frac{T}{T_i} \rfloor} \frac{p_m^{idle}(T - kT_i)y_i(k)}{b_{size}}, \quad (50)$$

where $y_i(k)$ is the size of k th data sent by the i child node, and $E[y_i(k)] = \lambda_i T_i \mu_i$.

$$\sum_{k=1}^{\lfloor \frac{T}{T_i} \rfloor} \frac{p_m^{idle}(T - kT_i)\lambda_i T_i \mu_i}{b_{size}} \leq E[e_b^{FI}(T)] \leq \sum_{k=0}^{\lfloor \frac{T}{T_i} \rfloor} \frac{p_m^{idle}(T - kT_i)\lambda_i T_i \mu_i}{b_{size}}. \quad (51)$$

This yields $E[e_b^{FI}(T)(i)] \approx \frac{p_m^{idle} T^2}{2T_i} \frac{\lambda_i T_i \mu_i}{b_{size}} = \frac{p_m^{idle} T^2 \lambda_i \mu_i}{2b_{size}}$.

- (2) The energy for data transmission and for reading/writing data from/into memory banks, plus elevating/demoting the power status of memory banks, is

$$e_t^{FI}(i) = \sum_{k=0}^{n_i(T)} y_i(k)(e_w^{TX} + e_m^w + e_m^r) + 2e_m^{resyn}, \quad (52)$$

$$E[e_t^{FI}(i)] = T \left(\lambda_i \mu_i (e_w^{TX} + e_m^w + e_m^r) + 2e_m^{resyn}/T_i \right),$$

where $n_i(T)$ is the number of transmissions by the i th node over T interval, and its expectation $E[n_i(T)]$ is T/T_i .

- (3) The energy for data reception ($e_r^{FI}(i)$) from the i th child is

$$e_r^{FI}(i) = \sum_{k=0}^{n_i(T)} y_i(k)e_w^{RX} + e_w^{wu}, \quad E[e_r^{FI}(i)] = T \lambda_i \mu_i e_w^{RX} + \frac{T e_w^{wu}}{T_i}, \quad (53)$$

where $n_i(T)$ denotes the number of data receptions at the parent over T , so that the number of radio-wakeups (by a radio-triggered wakeup mechanism) is

T/T_i , and the expected energy in radio wakeup over T is $\frac{T e_w^{wu}}{T_i}$. The energy for data reception is $T \lambda_i \mu_i e_w^{RX} \approx T \lambda_i \mu_i e_w^{TX}$, assuming that $e_w^{RX} \approx e_w^{TX}$.

Thus the average total energy of the parent node with k child nodes over T is:

$$E[e^{FI}(T)] = e_w^{wu} + \frac{p_m^{idle} T^2}{2b_{size}} \sum_{i=1}^k \lambda_i \mu_i + T \sum_{i=1}^k \left(\lambda_i \mu_i (2e_w^{TX} + e_m^w + e_m^r) + \frac{2e_m^{resyn} + e_w^{wu}}{T_i} \right) \quad (54)$$

Notice that only one radio wakeup for data transmission and $\sum_{i=1}^k T/T_i$ radio wakeups for data receptions from k child nodes in each cycle T . Using the same trick in the proof of Theorem 4, the long-run mean average energy is

$$\overline{e^{FI}(T)} = \frac{e_w^{wu}}{T} + \frac{T p_m^{idle}}{2b_{size}} \sum_{i=1}^k \lambda_i \mu_i + \sum_{i=1}^k \left(\lambda_i \mu_i (2e_w^{TX} + e_m^w + e_m^r) + \frac{2e_m^{resyn} + e_w^{wu}}{T_i} \right). \quad (55)$$

Taking derivative of (55) w.r.t. T gives $\frac{\partial \overline{e^{FI}(T)}}{\partial T} = -\frac{e_w^{wu}}{T^2} + \frac{\lambda p_m^{idle}}{2b_{size}} \sum_{i=1}^k \lambda_i \mu_i$. Resolving $\frac{\partial \overline{e^{FI}(T_p)}}{\partial T_p} = 0$ yields (49). \square

Let $v = (\lambda_1 \mu_1, \dots, \lambda_k \mu_k)$ be a bandwidth distribution vector of k child nodes, and $\overline{e^{FI}(T_p)}(v)$ refer to the long-run mean average energy consumption of the parent node under v . Substituting (49) into (55) gives

$$\begin{aligned} \overline{e^{FI}(T_p)}(v) = & \sqrt{\frac{2p_m^{idle} e_w^{wu} \sum_{i=1}^k \lambda_i \mu_i}{b_{size}}} + (2e_w^{TX} + e_m^w + e_m^r) \sum_{i=1}^k \lambda_i \mu_i \\ & + \left(e_m^{resyn} + \frac{e_w^{wu}}{2} \right) \sqrt{\frac{2p_m^{idle}}{e_w^{wu} b_{size}}} \sum_{i=1}^k \sqrt{\lambda_i \mu_i}, \end{aligned} \quad (56)$$

where $\sum_{i=1}^k \lambda_i \mu_i$ refers to the total bandwidth of the parent node.

A natural question arises as to how bandwidth distribution among child nodes affects the overall power consumption of the parent node.

To answer this question, we first introduce the notion of *majorization*, and then provide a lemma to facilitate necessary derivations.

For any vector $x = (x_1, \dots, x_n) \in \mathcal{R}^n$, let $x_{(1)} \leq \dots \leq x_{(n)}$ be the component of x in ascending order, and $x_{\downarrow} = (x_{(1)}, \dots, x_{(n)})$ be the ascending rearrangement of x .

Definition 4. For two vectors $x, y \in \mathcal{R}^n$,

$$x < y \quad \text{if} \quad \begin{cases} \sum_{i=1}^k x_{(i)} \geq \sum_{i=1}^k y_{(i)}, & 1 \leq k \leq n-1 \\ \sum_{i=1}^n x_{(i)} = \sum_{i=1}^n y_{(i)} \end{cases} \quad (57)$$

Then x is said to be *majorized* by y [Marshall and Olkin 1979].

A trivial example is given to illustrate the notion of *majorization*:

$$\left(\frac{1}{n}, \dots, \frac{1}{n}\right) \prec \left(\frac{1}{n-1}, \dots, \frac{1}{n-1}, 0\right) \prec \dots \prec \left(\frac{1}{2}, \frac{1}{2}, 0, \dots, 0\right) \prec (1, 0, \dots, 0)$$

LEMMA 2. *Let $x = (x_1, \dots, x_n), y = (y_1, \dots, y_n)$ be two vectors and g be a concave function. If $x \prec y$, then $\sum_{i=1}^n g(y_i) < \sum_{i=1}^n g(x_i)$.*

The proof can be seen in Marshall and Olkin [1979] and therefore is omitted.

THEOREM 7. *Let $v = (\lambda_1\mu_1, \dots, \lambda_k\mu_k)$ and $v' = (\lambda'_1\mu'_1, \dots, \lambda'_k\mu'_k)$ be two child bandwidth distribution vectors. Letting $\sum_{i=1}^k \lambda_i\mu_i = \sum_{i=1}^k \lambda'_i\mu'_i = \mathcal{B}$. If v is majorized by v' ($v \prec v'$), then the parent node consumes more power under v than under v' , with the same optimal buffer interval T_p . Namely, $\overline{e^{FI}(T_p)}(v) > \overline{e^{FI}(T_p)}(v')$.*

PROOF. Since $\sum_{i=1}^k \lambda_i\mu_i = \sum_{i=1}^k \lambda'_i\mu'_i = \mathcal{B}$, by (49), the optimal interval for the parent node, $T_p = \sqrt{\frac{2e_w^{wu}b_{size}}{p_m^{idle}\mathcal{B}}}$, is identical under both v and v' . Based on lemma 2 that $\sum_{i=1}^k \sqrt{\lambda_i\mu_i} > \sum_{i=1}^k \sqrt{\lambda'_i\mu'_i}$ since $v \prec v'$, we get

$$\overline{e^{FI}(T_p)}(v) - \overline{e^{FI}(T_p)}(v') = \left(e_m^{resyn} + \frac{e_w^{wu}}{2}\right) \sqrt{\frac{2p_m^{idle}}{e_w^{wu}b_{size}}} \left(\sum_{i=1}^k \sqrt{\lambda_i\mu_i} - \sum_{i=1}^k \sqrt{\lambda'_i\mu'_i}\right) > 0. \quad (58)$$

Theorem 7 is thus proved. \square

Let \mathcal{V}_B be a *convex* space formed by a set of vectors satisfying $v = (\lambda_1\mu_1, \dots, \lambda_k\mu_k) \in \mathcal{V}_B$ iff $\sum_{i=1}^k \lambda_i\mu_i = \mathcal{B}$. Let $v_{\leftrightarrow} = (\frac{\mathcal{B}}{k}, \dots, \frac{\mathcal{B}}{k}) \in \mathcal{V}_B$ and $v_{\dagger} = (\mathcal{B}, 0, \dots, 0) \in \mathcal{V}_B$ be two child bandwidth distribution vectors. The vector v_{\leftrightarrow} represents a uniform bandwidth distribution in which each child equally contributes the $\frac{\mathcal{B}}{k}$ bandwidth of the parent, while v_{\dagger} is an extremely uneven bandwidth distribution where only child constitutes the \mathcal{B} of the parent node and the remaining children contribute nothing. Clearly, for any bandwidth distribution $v \in \mathcal{V}_B$, $v_{\leftrightarrow} \prec v \prec v_{\dagger}$. It follows from Theorem 7, we have

$$\overline{e^{FI}(T_p)}(v_{\leftrightarrow}) = \max\{\overline{e^{FI}(T_p)}(v) : v \in \mathcal{V}_B\} \quad (59)$$

$$\overline{e^{FI}(T_p)}(v_{\dagger}) = \min\{\overline{e^{FI}(T_p)}(v) : v \in \mathcal{V}_B\}. \quad (60)$$

By the aid of (58), the power difference of the parent node under v_{\leftrightarrow} and v_{\dagger} is

$$\overline{e^{FI}(T_p)}(v_{\leftrightarrow}) - \overline{e^{FI}(T_p)}(v_{\dagger}) = \left(e_m^{resyn} + \frac{e_w^{wu}}{2}\right) \sqrt{\frac{2p_m^{idle}\mathcal{B}}{e_w^{wu}b_{size}}} (\sqrt{k} - 1). \quad (61)$$

Theorem 7 provides a means of quantifying the impact of uniformity in child bandwidth distribution on power consumption of the parent node. Equation (61) in particular gives the bound on the range of such bandwidth distribution effect, which is proportional to the square root of the number of child nodes.

6. CONCLUSION AND FUTURE WORK

The longevity of battery-powered sensor networks is an essential performance metric of around-clock environmental surveillance and monitoring. This article focuses on the exploitation of power-aware buffering schemes to reduce power consumption of sensor networks based on the radio-triggered power management. It shows that the power-aware buffering is a nonnegligible factor that effectively improves the lifespan of sensor networks and that the power-oblivious buffering is harmful as it is very likely to result in an excessive power consumption.

An in-depth analysis shows that the fixed-size and fixed-interval buffering schemes differ markedly in relation to data size variability. The power-aware fixed-size buffering scheme is implicated in both the skewness and coefficient of variation in the data size distribution, and its performance could deteriorate rapidly when the data size is of high-variance. In contrast, the hallmark of the fixed-interval buffering scheme is its immunity to the data-size variation. The fixed-interval buffering scheme is therefore the buffering choice for its performance stability in a variety of sensor-based application environments. Furthermore, in the context of the fixed-interval buffering scheme, we establish the power consumption relationship between parent and child nodes in a static data collection tree in sensor networks. We show that a uniform bandwidth distribution among child nodes in fact consumes more power of the parent node than an uneven bandwidth distribution.

These findings are valuable in understanding the asymptotic behavior of the power-aware buffering schemes in the presence of size variability. They provide well-informed guidance on determining the optimal buffer size or buffer interval based on the power parameter of radio and memory banks, allowing us to judiciously select a buffering scheme that better tailors to data arrival rate and data size distribution.

Our future work will focus on 1) validating the buffering models in a lab environment, including simulation and system implementation; 2) studying the power-aware buffering issue under real-time constraints. The goal of the new research avenue is to study strategy that can provide optimal trade-off between power-aware buffering and responsiveness.

APPENDIX

We first introduce the notion of first ladder height, then present Lai and Seigmund's remarkable theorem [Lai and Seigmund 1977], which has laid theoretical basis for quantifying the size variation impact on the fixed-size buffering scheme.

Definition 5. Let x_1, x_2, \dots be random variables following certain distribution \mathcal{F} with parameter θ , and S_n be the random walk consisting of the partial sum $S_n = \sum_{i=1}^n x_i$. The first time $\tau = \inf\{n : S_n > 0\}$ that the random walk is positive is called the first ladder epoch and the first positive value S_τ taken by the random walk is called as the first ladder height.

A detailed derivation of Theorem 8 can be found in Lai and Seigmund [1977, 1979]. Keener [1987] later gave a simplified expression for k in terms of moments of ladder height variables.

THEOREM 8. *Let $\{x_i, i \geq 1\}$ be a random walk with a mean of $\mu_x = E[x_1] > 0$ and finite variance σ_x^2 . Let $\tau(b) = \min\{n \geq 1 : \sum_1^n x_i > b\}$, $R_b = S_{\tau(b)} - b$, $M = \min_{n \geq 0} S_n$, $\tau^+ = \tau(0)$, and $H = S_{\tau^+}$. As $b \rightarrow \infty$, the stopping time variance becomes*

$$\sigma_{\tau(b)}^2 = \frac{b\sigma_x^2}{\mu_x^3} + \frac{k}{\mu_x^2} + o(1), \quad (62)$$

where $\tau(b)$ refers to the stopping time and k is the key constant with a rather complicated expression as follows:

$$k = \frac{\sigma_x^2 EH^2}{2\mu_x EH} + \frac{3}{4} \left(\frac{EH^2}{EH} \right)^2 - \frac{2}{3} \frac{EH^3}{EH} - \frac{EH^2 EM}{EH} - 2 \int_0^\infty ER_x P(M \leq -x) dx \quad (63)$$

Based on Theorem 8, we provide a proof of Theorem 1 in this article.

PROOF. Consider random variables with positive increments $\{x_i > 0, i \geq 0\}$ because the received data size is always positive. Let's reexamine the expression for k in (63) under the positive increment condition. Define the ladder epochs $\tau_0 = 0$ and $\tau_1^+ = \tau^+$, and $\tau_{n+1}^+ = \inf\{k > \tau_n^+ : S_k > S_{\tau_n^+}\}$ for $n \geq 1$. For $n > 0$, the $(n+1)$ th ladder height $H_{n+1} = S_{\tau_{n+1}^+} - S_{\tau_n^+}$. Consider $H_{n+1} = S_{\tau_{n+1}^+} - S_{\tau_n^+} = x_{n+1}$ under the positive increment condition, i.e., $H = x$. As a result, we obtain

$$EH = E[x] = \mu_x, \quad EH^2 = E[x^2] = \sigma_x^2 + \mu_x^2, \quad (64)$$

$$EH^3 = E[x^3] = 3\sigma_x^2 \mu_x + \mu_x^3 + \sigma_x^3 \gamma(x)$$

Using the identity $EM = \frac{EH^2}{2EH} - \frac{E[x^2]}{2E[x]}$ [Woodroffe 1976], we obtain that $EM = 0$ when the positive random walk is assumed. Substitution of these identities into (63) yields a simplified expression for k , denoted by k^* , as follows:

$$\begin{aligned} k^* &= \frac{\sigma_x^2(\sigma_x^2 + \mu_x^2)}{2\mu_x^2} + \frac{3}{4} \left(\frac{\sigma_x^2 + \mu_x^2}{\mu_x} \right)^2 - \frac{2}{3} \left(\frac{3\sigma_x^2 \mu_x + \mu_x^3 + \sigma_x^3 \gamma(x)}{\mu_x} \right) \quad (65) \\ &= \frac{5\sigma_x^4}{4\mu_x^2} + \frac{\mu_x^2}{12} - \frac{2\sigma_x^3 \gamma(x)}{3\mu_x} = \frac{5\sigma_x^2 c_v^2(x)}{4} + \frac{\mu_x^2}{12} - \frac{2\sigma_x^3 c_v(x) \gamma(x)}{3}, \end{aligned}$$

Substitution of (65) into (62) completes the proof. \square

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers for their incredibly insightful critiques that help us to significantly improve the quality of the paper.

REFERENCES

- ANSARI, J., PANKIN, D., AND MAHONEN, P. 2009. Radio-triggered wake-ups with addressing capabilities for extremely low power sensor network applications. *Int. J. Wirel. Inform. Netw.* 16, 3, 118–130.

- CC2420. 2004. Ti/chipcon CC2420 datasheet.
<http://inst.eecs.berkeley.edu/cs150/Documents/CC2420.pdf>.
- CULLER, D. E. AND MULDER, H. 2004. Smart Sensors to Network the World. *Scientific Amer.* 85–91.
- FAN, X., ELLIS, C., AND LEBECK, A. 2001. Memory controller policies for DRAM power management. In *Proceedings of the International Symposium on Low Power Electronics and Design*. ACM, 129–134.
- FELLER, W. 1971. *An Introduction to Probability Theory and Its Applications*. Vol. II, 2nd Ed. John Wiley & Sons.
- FLAUTNER, K., KIM, N., MARTIN, S., BLAAUW, D., AND MUDGE, T. 2002. Drowsy caches: simple techniques for reducing leakage power. In *Proceedings of the 29th Annual International Symposium on Computer Architecture*. 148–157.
- GUPTA, M. AND SINGH, S. 2003. Greening of the Internet. In *Proceedings of SIGCOMM*. ACM, 1–8.
- HEMPSTEAD, M., TRIPATHI, N., MAURO, P., WEI, G.-Y., AND BROOKS, D. 2005. An ultra Low Power System Architecture for Wireless Sensor Network Applications. In *Proceedings of the 32nd International Symposium on Computer Architecture*. IEEE Computer Society Press, 208–219.
- JURDAK, R., BALDI, P., AND LOPES, C. V. 2007. Adaptive low power listening for wireless sensor networks. *IEEE Trans. Mobile Comput.* 6, 1–17.
- KEENER, R. 1987. A note on the variance of a stopping time. *Annals Statis.* 15, 4, 1709–1712.
- LAI, T. L. AND SEIGMUND, D. 1977. A nonlinear renewal theory with applications to sequential analysis I. *Annals Statis.* 5, 5, 946–954.
- LAI, T. L. AND SEIGMUND, D. 1979. A nonlinear renewal theory with applications to sequential analysis II. *Annals Statis.* 7, 1, 60–76.
- LANDMAN, P. E. AND RABAEY, J. 1995. Architectural power analysis: The dual bit type method. *IEEE Trans. VLSI Syst.* 3, 2, 173–187.
- LEE, M., SEO, E., LEE, J., AND KIM, J. 2007. PABC: Power-aware buffer cache management for low power consumption. *IEEE Trans. Comput.* 56, 4, 488–501.
- LEVIS, P. 2005. Embedded sensor architecture.
<http://csl.stanford.edu/pal/talks/ee282-f05-sensors.pdf>.
- LIN, G. AND STANKOVIC, J. 2004. Radio-triggered wake-up capability for sensor networks. In *Proceedings of the 10th IEEE Real-Time and Embedded Technology and Applications Symposium*. IEEE Computer Society, 27–36.
- LIN, G. AND STANKOVIC, J. 2005. Radio-triggered wake-up for wireless sensor networks. *Real-Time Syst.* 29, 2–3, 157–182.
- LING, Y. AND CHEN, C. 2007. Energy saving via power-aware buffering in wireless sensor networks. In *Proceedings of the Annual Joint Conferenec of the IEEE Computer and Communications Society (INFOCOM)*.
- MAINWARING, A., CULLER, D., POLASTRE, J., SZEWCZYK, R., AND ANDERSON, J. 2002. Wireless sensor networks for habitat monitoring. In *Proceedings of the 1st ACM International Workshop on Wireless Sensor Networks and Applications*. ACM Press, New York.
- MARSHALL, A. W. AND OLKIN, I. 1979. *Inequalities: Theory of Majorization and Its Applications*. Adademic Press, New York.
- ROSS, S. M. 1996. *Stochastic Processes*. John Wiley & Sons, Inc., New York.
- SATYANARAYANAN, M. 1996. Fundamental challenges in mobile computing. In *Proceedings of the 15th ACM Symposium on Principles of Distributed Computing*. ACM, 1–7.
- SHNAYDER, V., HEMPSTEAD, M., CHEN, B., ALLEN, G. W., AND WELSH, M. 2004. Simulating the power consumption of large-scale sensor applications. In *Proceedings of the International Conference on Embedded Sensor Systems (SenSys'04)*. ACM Press, 188–200.
- TARJAN, D., THOZIYOOR, S., AND JOUPPI, N. P. 2006. CACTI4.0. In *HPL-2006-86*.
- WOO, A., TONG, T., AND CULLER, D. 2003. Taming the underlying challenges of reliable mutihop routing in sensor networks. In *Proceedings of the International Conference on Embedded Sensor Systems (SenSys'03)*. ACM, 14–27.

- WOODROOFE, M. 1976. A renewal theorem for curved boundaries and moments of first passage time. *Annals Probab.* 4, 1, 67–80.
- XIE, T. 2008. SEA: A striping-based energy-aware strategy for data placement in RAID-structured storage systems. *IEEE Trans. Comput.* 57, 748–761.
- YE, W., HEIDEMANN, J., AND ESTRIN, D. 2002. An energy-efficient MAC protocol for wireless networks. In *Proceedings of the Annual Joint Conferenec of the IEEE Computer and Communications Society (INFOCOM)*.

Received May 2009; revised January 2010, April 2010; accepted June 2010