This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE INFOCOM 2008 proceedings.

An Energy Efficiency Evaluation for Sensor Nodes with Multiple Processors, Radios and Sensors

Deokwoo Jung
Electrical Engineering
Yale University
New Haven, Connecticut 06511–0250
Email: deokwoo.jung@yale.edu

Andreas Savvides
Electrical Engineering
Yale University
New Haven, Connecticut 06511–0250
Email: andreas.savvides@yale.edu

***Abstract*—This paper constructs a model for studying the energy efficiency of sensor node architectures featuring a pair of low-end, low-power processor and radio and a pair of high-end, energy efficient processor and radio. Such nodes can have a highly dynamic range of operation ranging from the collection of simple temperature measurements or motion detection all the way up to sophisticated signal processing of sensor data. Our model explores the energy efficiency tradeoffs related to the decision of which processor and which radio should be used for each task. For this we derive a general Semi-Markov Decision process model for maximizing the asymptotic lifetime of two alternate designs, one with dynamic and one with static interconnect. The resulting models are validated with simulation and are applied to the reported measurements from an existing platform with two radios and two processors. Our results show how to quantify the gains of such design with respect to the power consumption properties of each component. Furthermore, based on our power budget calculation, we conclude that the deisgn of reconfigurable interconnect between multiple processors and radios would result in efficiency gains despite the energy overhead such device may incur.**

I. Introduction

In many applications today, wireless sensor nodes are expected to perform a highly dynamic range of operation from simple temperature and motion sensing all the way up to complex digital signal processing tasks such as video and image processing. A close examination of the power consumption profiles of the components of such nodes suggest that nodes can significantly improve their energy efficiency by utilizing pairs of complementary low-end (low-power), high-end (power-efficient) processors and radios. Low-end processors such as the TI's MSP430 on Telos motes and Atmel's AVR 128 on Micaz nodes are low power ($3mW$ and $33mW$ respectively) but have have high energy per bit (0.934mJ/bit) compared to higher-end processors such as the PXA family of processors used in the iMote2 and LEAP sensor nodes (0.046mJ/bit)[4]. Radios exhibit similar properties. The popular CC2420 (low-end) radio requires 979nJ/bit while a higher-end IEEE 802.11 radio consumes 112nJ/bit[4]. Both processors and radios incur an analogous wakeup overhead, that is, high-end devices tend to have a higher startup cost than low-end devices. Given these properties, recent node implementations [4],[5],[8] have demonstrated favorable results by leveraging the aforementioned tradeoffs.

In this paper, we introduce a more systematic analytical framework for evaluating these tradeoffs. We do so by considering two alternative architectures, one with a static and another with dynamic interconnect, shown in Figure 1 (a) and (b). In the architecture with the static interconnect the low-end processor communicates via the low-end radio and the high-end processor communicates via the high end radio. Data from the sensors can therefore flow through one of the two possible processor/radio pairs. The dynamic interconnect architecture features a dynamic interconnect chip that increases the number of possible paths from two to four, that is, data from the sensor can flow through any processor/radio combinations.

The contribution of this paper is in the analysis of the above two architectures featuring static and dynamic interconnect. Using these architectures, we consider the decision problem of how each platform should activate its low/high-end components (i.e. CPU and radios) so as to maximize the node asymptotic lifetime. We answer this question by formulating the problem as a semi-markov decision process and by introducing two main metrics: average cost per stage and energy efficiency gain. We derive a model to evaluate the energy efficiency of alternative design options and to find ways of improving energy efficiency. We illustrate this process by using measurements from the LEAP node reported in [4]. By applying our models we show that an architecture with two processor/radio pairs and reconfigurable interconnect has the potential of increasing lifetime by more than 8 times compared to an existing low-end node (i.e Telos). Using the same reference architecture we show that the use of dynamic interconnect may result in up to 3 times improvement in lifetime compared to a similar architecture with static interconnect. The overall benefit of having these models is to provide an insight of what an architecture can do with respect to the properties and tradeoffs of the power profiles of its individual components. The results provide an upper bound for energy efficiency given the hardware characteristics. We hope that in developing these bounds, the role of each hardware properties can be better understood and would also provide good hints on how to develop energy efficient node software schemes that will approach these bounds.

The rest of the paper is organized as follows. In Section II we survey the related work. In Section III we introduce the

978-1-4244-2026-1/08/$25.00 © 2008 IEEE

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.

problem statement, we provide an overview of the result and we introduce some notation. Section IV provides the details of our model. Section V performs a numerical validation of the model and section VI concludes the paper.

II. Related Work

A discussion on the energy management techniques in wireless sensors networks is outlined in [6] along with a node level platform design optimization. The node level energy efficiency or lifetime analysis has been studied in [9], [2], and [3]. In [9], the authors characterize hardware components at a very detailed level to simulate power consumption of a node as close as possible. In [2], the authors use hybrid automata models for analyzing power consumption of a node at the operating system level (TinyOS). In both [9] and [2], the node lifetime analysis is limited to the specific platforms and do not provide an analytical tool for comparing energy-efficiency among choice of hardware and platforms. The node lifetime analysis applicable to a broad type of platform has been studied in [3]. In this paper, the authors propose an analytical lifetime performance model for trigger-driven and schedule driven node based on semi-markov chain model. In [3], however, the platform architecture is limited to a node with single CPU-radios pair. Our paper expands the result of [3] to an architecture dual CPUs and radios. Therefore, the average power consumption models proposed in [3] is the special case of our results. Sensor node platforms with both low-end and high-end component combination are also developed in [4], [5], [8]. In [8], the authors point out the importance of platforms with a large dynamic range of power and performance for long node lifetime. In the paper, hardware modules with different power and performance characteristics are configured into a microsensor system for various application. The configuration of module is done off-line before deployment. A sensor node with dual CPU and radios capable of reconfiguring its components at post-deployment is developed in [4]. However, the configuration of hardware components is limited to static interconnect. In [5], a sensor node platform with multiple sensors, CPUs, and radios with more flexible interconnect is developed. An dynamic interconnect chip is used to configure intermodule communication. Although these papers demonstrate good examples of nodes with multiple sensors, CPUs and radios, they do not provide a generalized energy efficiency analysis of trends as it is done in the work presented here.

III. Model Overview and Problem Statement

We assume all node types we consider basically follow the same sequential operation (Preprocessing, Processing, and Communication) as shown in Figure 2(a). The Figure describes a trigger-driven node of operation in which the sensors are managed by a low-power pre-processing unit. This preprocessing stage performs a first-order filtering of the data and wakes up a more powerful main processing unit if certain criteria are met. If the desired event types are sensed, it proceeds to make the necessary computations at the processing stage and transmits the outcome with the radio if needed using the communication stage. The asymptotic average power of such trigger-driven model has been studied in [3].

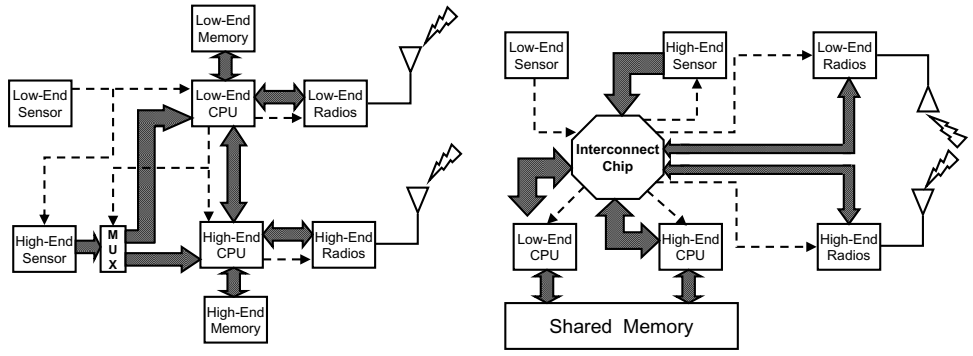
In [3], the asymptotic average power is constant for a given platform. The asymptotic average power in the dual processor/radio node ,however, depends on the decision of a node to use the higher or lower end component for a particular operation. For this case, we are interested in analyzing the smallest asymptotic average power when this new architecture optimally decides which components to utilize for each sensed event. This turns out to be the same as the problem of minimizing the average cost per stage in Figure 2 (b). To argue that this is the case, we note that the physical quantity of cost is energy to process (send) information and a random time is required for state transition. Therefore, the physical meaning of average cost per stage is the asymptotic average power consumption.

If the node continues to repeat the cycle in Figure 2(a) throughout its lifetime, we can conjecture that the average cost problem is equivalent to minimizing the average power consumption per cycle. This can be converted to an equivalent stochastic shortest path problem as shown Figure 2 (b) and (c) by splitting X_0 into X_0 and cost free termination state, t. Then our problem is to solve the shortest path from X_0 to t. In this shortest path problem decision actions occur upon transition to processing stage (X_1) and communication stage (X_2). For static interconnect, the action is taken once before transitioning to the processing stage and the same decision action is used before transitioning to the communication stage as shown 2(b). For the reconfigurable interconnect architecture, the two decision actions taken before transitioning to the processing and communication stages are shown in Figure 2(c). Let $\bar{E}_{00}$ and $\bar{T}_{00}$ denote the expected energy cost and the expected time starting from X_0 (preprocessing stage) to t (termination state) respectively. Then the shortest path problem can be formulated as the minimization of $\frac{\bar{E}_{00}}{\bar{T}_{00}}$. Before getting into the detailed formulation of the problem we first overview the main result of the semi-markov decision process.

A. Result Overview

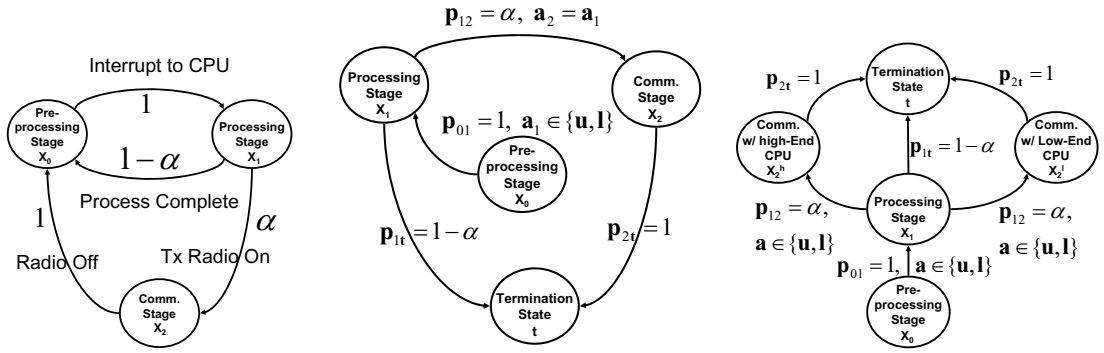
The semi-markov decision process that selects which processor and which radio should be activated for a particular task can be intuitively interpreted by the geometric analysis of Figure 3. The two alternative dual processor/radio architectures from Figures 1 (a) and (b) have different power consumption rates and wake-up energy costs depending on the combination of components switched On/Off. Our result overview in Figure 3 shows that the energy consumption rate is a linear function of the total time, T spent on processing and communication. Our architecture takes On/Off decisions on two pairs of High/Low-End CPUs and radios resulting in four possible linear functions, E_1, E_2, E_3 and E_4, that can be drawn on the Time-Energy(T-E) plane. If the low-end processor and low-end radio were selected, the node will follow the lowest slope line E_1 in Figure 3. Conversly, if the high-end pair (high-end processor, high-end radio) were selected then the node

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.



(a) (b)

Fig. 1: (a) Dual Processor/Radio Architecture with Static Interconnect, (b) Dual Processor/Radio Architecture with Reconfigurable Interconnect



(a) (b) (c)

Fig. 2: (a) Semi-Markov Operation Model of Trigger-Driven Architecture, (b) Associate Shortest Path Structure of Dual Processor/Radio Architecture with Static Interconnect, (c) Associate Shortest Path Structure of Dual Processor/Radio Architecture with Reconfigurable Interconnect.

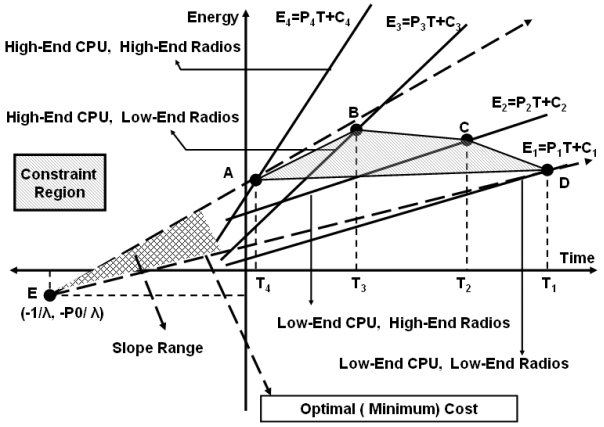


Fig. 3: Illustrated Example of Geometric Interpretation of Optimal Decision in Dual Processor/Radio Architecture : The optimal cost (minimum average cost per stage) is determined by the minimum slope of line which is drawn from E and pass though a constraint region $\overline{ABCD}$. In this example, Dual Processor/Radio node determines to use low-end CPU and radios for the optimal policy.

will follow the steepest slope line E_4. From this it follows that if the architecture was configured statically (as in Figure 1 (b)), the node would be able to operate on either line E_1 or E_4. If however, the interconnect between the processors and radios was dynamically reconfigurable (as in Figure 1 (c)), then two additional combinations would be possible; i.e., high-end processer, low-end radio and low-end processor, high-end radio. This implies that the platform can now follow four alternative lines E_1, E_2, E_3 and E_4, depending on the task at hand. The semi-markov decision process computes the decision functions on the T-E plane given the time variables, T_1, T_2, T_3, and T_4. The decision functions set a constraint region, $\overline{ABCD}$ on T-E plane as shown in the Figure. The average cost per stage (or the average power consumption) then becomes the slope of each line passing though the constraint region from E located by event arrival rate, λ and preprocessing power, P_0. For the example we constructed in Figure 3, the optimum cost is determined by the line with the smallest slope which is $\overline{ED}$. In other words, give a certain set of times taken to process and communicate a particular event, T_1, T_2, T_3, and T_4, $\overline{ED}$ (low-end processor, low-end radio combination) represents the optimum decision since it has the smallest slope on the E-T plane. Similarly, depending on the nature of the events received, the decision algorithm will select the appropriate optimum line between $\overline{EA}$, $\overline{EB}$, $\overline{EC}$ and $\overline{ED}$.

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.

Type	Mode	S_l	S_h	C_l	C_h	R_l	R_h	InC	Power
Low End Node	X_0	On	Off	Off	–	Off	–	–	P_0
	X_1	Off	On	On	–	Off	–	–	P_1
	X_2	Off	Off	On	–	On	–	–	P_3
High End Node	X_0	On	Off	–	Off	–	Off	–	P_0
	X_1	Off	On	–	On	–	Off	–	P_2
	X_2	Off	Off	–	Off	–	On	–	P_6
Static.	X_0	On	Off	Off	Off	Off	Off	–	P_0
	X_1	Off	On	On	Off	Off	Off	–	P_1
		Off	On	Off	On	Off	Off	–	P_2
	X_2	Off	Off	On	Off	On	Off	–	P_3
		Off	Off	Off	On	Off	On	–	P_6
Reconf.	X_0	On	Off	Off	Off	Off	Off	–	P_0
	X_1	Off	On	On	Off	Off	Off	On	P_1
		Off	On	Off	On	Off	Off	On	P_2
	X_2	Off	Off	On	Off	On	Off	On	P_3
		Off	Off	On	Off	Off	On	On	P_4
		Off	Off	Off	On	On	Off	On	P_5
		Off	Off	Off	On	Off	On	On	P_6

TABLE II: Power domain description: Static.: The dual processor/radio architecture with static interconnect, Recon.: The dual processor/radio Architecture with reconfigurable interconnect

The detailed formulation and geometric analysis of this result will be discussed in more detail in section IV.

B. Node Definitions and Assumptions

A set of definitions and variables for formulating the problem are described in Table I. In addition, our notation uses the following symbols; A bar to denote expected value (i.e., the expected value of the variable A is $\bar{A}$), A bold type to denote a vector (i.e., the $\mathbf{x} = (x_1, x_2)$), A dot to denote vector dot product (i.e., $(x_1, x_2) \bullet (y_1, y_2) = x_1y_1 + x_2y_2$), $\|\mathbf{x}\|$ to denote 1-norm[1]. We formally define the set of hardware components as $\mathcal{G} = \{S_l, S_h, C_l, C_h, R_l, R_h, InC\}$ where symbolic definitions are described in Table I.

Our analysis considers a simplified version of the power modes available on sensor nodes. We only consider the ON/OFF state of each components, and eliminate many impractical modes. The power consumption of each component is denoted as P_M, where $M \in \mathcal{G}$. The power modes for simple low/high-end node, static and reconfigurable are shown in Table II. In order to keep consistency and simplicity of notation, we assume that P_{InC} is implicitly added to P_i for the reconfigurable interconnect platform except for P_0 in the table.

In our analysis, we assume that a node has only one type of job per event and the high-end CPU takes less time but costs more power and wake-up energy per job than the low-end CPU , i.e $Y_h < Y_l$, $P_{C_l} < P_{C_h}$, and $C_{1,l} < C_{1,h}$. Likewise, we also assume that the high-end radios takes less time but costs more power and wake-up energy to send information than the high-end radios, i.e $Z_h < Z_l$, $P_{R_l} < P_{R_h}$, and $C_{2,l} < C_{2,h}$

C. Semi-Markov Decision Model Overview

In our formulation we assume simple representative node operation model for a surveillance application. The model described in this paper makes the following assumptions:

[1]p-norm is defined as $\|\mathbf{x}\|_p = (|x_1|^p + |x_2|^p + ..|x_n|^p)^{\frac{1}{p}}$.

1) All nodes cycle three consecutive operations- preprocessing, processing, and communication.
2) The sojourn time at processing and communication stage is small compared to inter-arrival time of events.
3) The time duration a node spends in each state is independent and identically distributed (i.i.d.) with arbitrary distribution.
4) When an event is detected, the node processes it and sends the information to a base station (or another node) with probability α.
5) The base-station can communicate anytime with any radios of end-nodes via 1-hop star network.
6) All nodes have infinite energy resource

The first three assumptions,1,2, and 3 imply that the power state transitions may be modeled as a semi-Markov chain as shown 2(a). More specially, assumption 2 makes it possible to use the assumption 3 by approximating the time between the event interrupt and the end of the communication stage to the event inter arrival time. Later our analysis will condition $\lambda \leq T^* = \frac{\|C_4\| - \|C_1\|}{\|P_4\| - \|P_1\|}$. The basic analysis of node lifetime using semi-markov chain has been shown in [3]. In addition to the semi-markov chain, the dual processor/radio node has to make a decision of turning on and off each component upon transition to next stage for every iteration of the cycle during its lifetime. The decision is made to maximize the expected node lifetime. Therefore this adds a new control (or decision) parameter to the original problem formulation in [3]. This is called semi-markov decision process.

In the semi-markov decision process, decisions are applied at discrete time but costs(energy) are continuously accumulated. Let's $a \in A = \{l, h\}$ denote the decision action where l and h represent 'choosing' low-end and high-end component respectively. We assume at any time a sensor node is in any of three states, X_i, $i \in \{0, 1, 2\}$: Preprocessing ($i = 0$), Processing ($i = 1$), and Communication ($i = 2$). Let's $x(t)$ and $X(n)$ denote a state a sensor node stays at continuous time t and at discrete time n respectively. We further suppose that a cost function is imposed on the model in the following manner; if action a is chosen when in state i, then an immediate transition energy cost $C(i, a)$ is incurred and a cost rate, power $c(i, a)$ is imposed until the next transition occurs. That is, if a transition occurs after t seconds, then the total energy cost incurred is given by $C(i, a) + tc(i, a)$.

In this paper, we do not consider energy overhead caused by network connectivity in this paper (Assumption 5). We consider the asymptotic energy consumption behavior from Assumption 6. The logic behind the seemingly contradicting assumption, is that minimizing energy consumption for infinite time asymptotically leads to maximizing node lifetime. Solving the infinite horizon problem where the accumulated cost will be infinite, it is more meaningful to minimize the average cost per stage starting from a state i rather than minimize total expected cost.

Let $E(T) = \int_0^T g(x(t), a(t))dt$ denote the total energy cost incurred by time T. $g(x(t), a(t))$ represents the energy

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.

Symbol	Description	Symbol	Description
λ	Event Arrival rate, $1/sec$	α	Probability of sending sensing information
Y_l	Processing time per event at low-end CPU,sec	Y_h	Processing time per sensed event at high-end CPU, sec
Z_l	Packet transmission time at low-end radios,sec	Z_h	Packet transmission time at high-end radios, sec
X_0	Preprocessing stage	X_1	Processing stage
X_2	Communication stage	L	Packet length, bit
$C_{1,l}$	Wake-UP energy cost of low-end CPU, mJ	$C_{1,h}$	Wake-UP energy cost of high-end CPU, mJ
$C_{2,l}$	Wake-UP energy cost of low-end radios, mJ	$C_{2,h}$	Wake-UP energy cost of high-end Radios, mJ
S_l	Low end sensor (Preprocessor)	S_h	High end sensor
C_l	Low end CPU	C_h	High end CPU
R_l	Low end radios	R_h	High end radios
InC	Interconnect chip	$\mathcal{G}$	The set of hardware components,
a_1	Decision action for CPUs	a_2	Decision action for radios
P_i	Total power consumption at power mode $i \in \{1...6\}$, mW	P_M	Power consumption of a component, $M \in \mathcal{G}$

TABLE I: List of variables

consumption rate when a node is in state, $x(t)$ with decision action, $a(t)$ at time t. Let $E_n = C(X(n), a_n) + \tau_n c(X(n), a_n)$ denote the energy cost incurred during time n transition interval, τ_n. Then the objective function to be minimized can be defined as following two forms.

$$J_\pi^1(i) = \lim_{t\to\infty} E_\pi\{\frac{E(t)}{t}|X_1 = i\} \quad (1)$$

$$J_\pi^2(i) = \lim_{N\to\infty} \frac{E_\pi\{\sum_{0\le k\le N} E_k|X_1 = i\}}{E_\pi\{\sum_{0\le k\le N} \tau_k|X_1 = i\}} \quad (2)$$

where E_π represents the average under stationary policy π. We note that the cost function equation(1) and (2) are equivalent under the certain condition [10]. Then we can convert (2) into the associate stochastic shortest path problem under recurrent state assumption [1]. Therefore, we can use the following Bellman's Equation to solve the optimal policy, π^* and minimum average energy cost, g.

$$h(i) = \min_{a\in A}\{\bar{C}(i,a) + \sum_{0\le k\le\infty} p_{ij}(a)h(j) - g\bar{\tau}(i,a)\} \quad (3)$$

where $\bar{\tau}(i,a) = \sum_{0\le k\le\infty} p_{ij}(a)\int_0^\infty t dF_{ij}(t|a)$, $\bar{C}(i,a) = C(i,a) + c(i,a)\tau(i,a)$, and $g = J_{\pi^*}^2(i) = \min_\pi J_\pi^2(i)$ for all $i \ge 0$. $p_{ij}(a)$ is the transition probability from X_i to X_j when action a is applied and $F_{ij}(t|a)$ is the cumulative distribution function of the transition time from X_i to X_j at time t given action a.

IV. Semi-Markov Decision Model and Energy-Efficiency Metric

A. Optimal Solution for Semi-Markov Decision Model

The stochastic shortest path structures of semi-markov decision process of the dual processor/radio architectures are shown in Figure 2(b) and (c) for static and reconfigurable interconnect respectively. Then optimization objective is to minimize the average cost of traveling from monitoring state, X_0 to termination state, t which is practically the monitoring state of the next cycle. As shown in Figure 2(c), the state, X_2 is divided into two subsequent states, $X^l{}_2$ and $X^h{}_2$ since the cost rate (power consumption) at X_2 depends on the previous decision; e.g. if a node decides to use the low-power CPU at the processing stage, then cost rate at X_2 should be either of P_3 or P_4 but not any of P_5 and P_6 as shown in Table II. The Bellman equation for Figure 2(c) can be derived using the equation (3) as follows.

$$\begin{aligned}
h^*(x_0) &= \frac{P_0}{\lambda} - \frac{g^*}{\lambda} + h^*(x_1) \\
h^*(x_1) &= \min_{a\in l,h}\{C_{1,l} + P_1\bar{Y}_l - g^*\bar{Y}_l + \alpha h^*(x_2^l), \\
&\qquad C_{1,h} + P_2\bar{Y}_h - g^*\bar{Y}_h + \alpha h^*(x_2^h)\} \\
h^*(x_2^l) &= \min_{a\in l,h}\{C_{2,l} + P_3\bar{Z}_l - g^*\bar{Z}_l, \\
&\qquad C_{2,h} + P_4\bar{Z}_h - g^*\bar{Z}_h\} \\
h^*(x_2^h) &= \min_{a\in l,h}\{C_{2,l} + P_5\bar{Z}_l - g^*\bar{Z}_l, \\
&\qquad C_{2,h} + P_6\bar{Z}_h - g^*\bar{Z}_h\} \quad (4)
\end{aligned}$$

Let's $\mathbf{u} = (a_1, a_2)$ denote the vector of two consecutive decision actions made on path in Figure 2(b) and (c). We also define following vectors.

- $\mathbf{u}_1 = (l,l), \mathbf{u}_2 = (l,h), \mathbf{u}_3 = (h,l), \mathbf{u}_4 = (h,h)$
- $\mathbf{P}_1 = (P_1, P_3)$, $\mathbf{P}_2 = (P_1, P_4)$
- $\mathbf{P}_3 = (P_2, P_5)$, $\mathbf{P}_4 = (P_2, P_6)$
- $\mathbf{T}_{\mathbf{u_k}} = (\bar{Y}_{a_1}, \alpha\bar{Z}_{a_2})$, $\mathbf{C}_{\mathbf{u_k}} = (C_{1,a_1}, \alpha C_{2,a_2})$,

Let $K_T(\mathbf{u}_k)$ and $K_E(\mathbf{u}_k)$ denote the average time and energy spent per sensed event respectively for a decision action, $\mathbf{u}_k$, and defined as follows.

$$K_E(\mathbf{u}_k) = \mathbf{P}_k \bullet \mathbf{T}_{\mathbf{u_k}} + \|\mathbf{C}_{\mathbf{u_k}}\| \quad , \quad K_T(\mathbf{u}_k) = \|\mathbf{T}_{\mathbf{u_k}}\| \quad (5)$$

Then the list of decision functions are derived from (4) as following.

$$\Lambda_{ij} = \frac{K_E(\mathbf{u}_i) - K_E(\mathbf{u}_j)}{K_T(\mathbf{u}_i) - K_T(\mathbf{u}_j)}, \quad \alpha_r = \frac{\bar{Y}_l - \bar{Y}_h}{\bar{Z}_l - \bar{Z}_h} \quad (6)$$

where $i \ne j \in \{1,2,3,4\}$

The hypercube decision space of a 5 tuple vector, $\mathbf{x} = (\lambda, Y_l, Y_h, Z_l, Z_h)$ can be divided into the following decision regions.

- $R_1 = \{\mathbf{x} \mid g(\mathbf{u}_1) > \max(\Lambda_{13}, \Lambda_{12}, \Lambda_{34})\}$
- $R_2 = \{\mathbf{x} \mid \Lambda_{34} > g(\mathbf{u}_1) > \max(\Lambda_{12}, \Lambda_{14})\}$
- $R_4 = \{\mathbf{x} \mid \Lambda_{24} < g(\mathbf{u}_2) < \min(\Lambda_{12}, \Lambda_{34})\}$

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.

- $R_5 = \{\mathbf{x} \mid \max(\Lambda_{12}, \Lambda_{34}) < g(\mathbf{u}_3) < \Lambda_{13}\}$

---

for $(\alpha_r > \alpha)$
- $R_3 = \{\mathbf{x} \mid \max(\Lambda_{34}, \Lambda_{23}) < g(\mathbf{u}_2) < \Lambda_{12}\}$
- $R_6 = \{\mathbf{x} \mid \Lambda_{34} < g(\mathbf{u}_3) < \min(\Lambda_{12}, \Lambda_{23})\}$

---

for $(\alpha_r < \alpha)$
- $R_3 = \{\mathbf{x} \mid \Lambda_{34} < g(\mathbf{u}_2) < \min(\Lambda_{12}, \Lambda_{23})\}$
- $R_6 = \{\mathbf{x} \mid \max(\Lambda_{34}, \Lambda_{23}) < g(\mathbf{u}_3) < \Lambda_{12}\}$

- $R_7 = \{\mathbf{x} \mid \Lambda_{12} < g(\mathbf{u}_4) < \min(\Lambda_{34}, \Lambda_{14})\}$
- $R_8 = \{\mathbf{x} \mid g(\mathbf{u}_4) < \min(\Lambda_{24}, \Lambda_{12}, \Lambda_{34})\}$

Then the optimal cost can be derived from (4) as following compact expression.

$$g_r^* = g(\mathbf{u}_k) = \frac{P_0 + \lambda K_E(\mathbf{u}_k)}{1 + \lambda K_T(\mathbf{u}_k)} \tag{7}$$

where
$k = 1$ if $\mathbf{x} \in (R_1 \vee R_2)$, $k = 2$ if $\mathbf{x} \in (R_3 \vee R_4)$
$k = 3$ if $\mathbf{x} \in (R_5 \vee R_6)$, $k = 4$ if $\mathbf{x} \in (R_7 \vee R_8)$

In case of static interconnect, a node has only one decision action, a_1 in transition between X_0 and X_1 and $a_2 = a_1$ as shown Figure 2(b). Therefore, $h^*(x_2^l) = C_{2,l} + P_3\bar{Z}_l - g^*\bar{Z}_l$ and $h^*(x_2^h) = C_{1,h} + P_6\bar{Z}_h - g^*\bar{Z}_h$ in equation (4). By solving equation (4), the optimal average cost can be obtained as following.

$$g_s^* = g(\mathbf{u}_1) \;\; if \;\; \Lambda_{14} < g(\mathbf{u}_1), \;\; g_s^* = g(\mathbf{u}_4) \;\; if \;\; \Lambda_{14} > g(\mathbf{u}_4) \tag{8}$$

In case of simple low-end and high-end node, the average cost per stage is simply $g(\mathbf{u}_1)$ and $g(\mathbf{u}_4)$ respectively.

$$g_h^* = g(\mathbf{u}_1), \qquad g_l^* = g(\mathbf{u}_4) \tag{9}$$

The optimal solution, g^*, turns out to be the asymptotic average power consumption, $\overline{P}_{ST,td}$ of a node given semi-makrov chain which is derived at [3]. Compared to formula of the $\overline{P}_{ST,td}$, the terms, K_T and K_E are now the function of decision actions. This is very intuitive result since the dual processor/radio node finds the best combination of active components in terms of average power consumption.

B. Energy Efficiency Analysis and Comparison

In this section, we use simplified notations; $K_{\mathbf{u}}$ for $K_E(\mathbf{u})$, T_y for processing time, T_z for communication time. We can convert the optimal cost, (7), into the following linear function of $\mathbf{T}$.

$$K_u = g(\mathbf{u})(\|\mathbf{T}\| - (-\frac{1}{\lambda})) + (-\frac{P_0}{\lambda})$$

The typical forms for (8) and (7), can be viewed as the half-line through $(-\frac{1}{\lambda}, -\frac{P_0}{\lambda})$ shown in Figure 4 and 5 respectively.

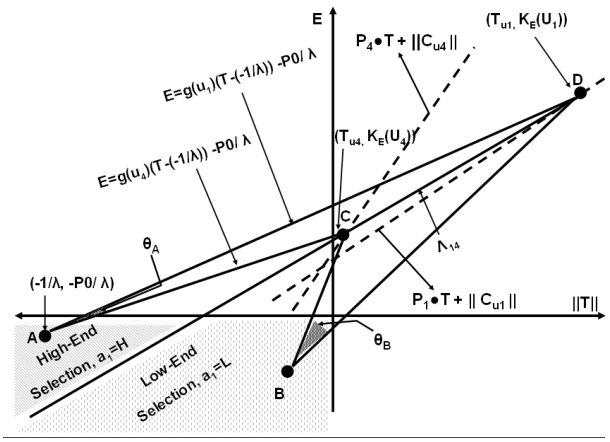


Fig. 4: Geometric Interpretation of Optimal Decision of dual processor/radio node with static interconnect : A decision function Λ_{14} sets the threshold of choosing high-end or low-end CPU (Radios) for event arrival rate λ given power mode ($\mathbf{P}_1$,$\mathbf{P}_4$) and processing/communication time vector (T_y, T_z)

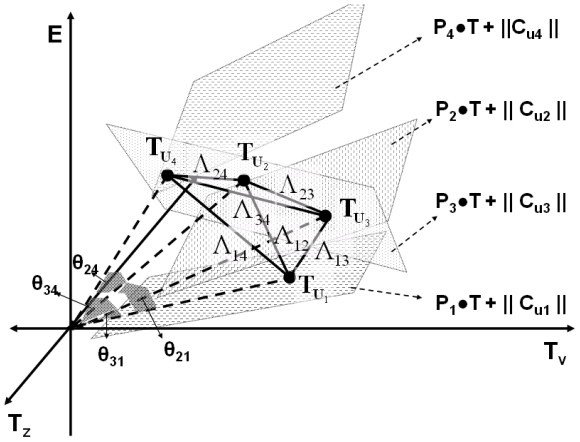


Fig. 5: Geometric Interpretation of Optimal Decision of dual processor/radio node with reconfigurable interconnect: Multiple decision functions, $\Lambda_{12}, ...\Lambda_{34}$ set the thresholds of choosing high-end or low-end CPU (Radios) for event arrival rate λ given power mode ($\mathbf{P}_1$,$\mathbf{P}_2$,$\mathbf{P}_3$,$\mathbf{P}_4$) and processing/communication time vector (T_y, T_z)

In Figure 4, the decision function, Λ_{14} is simply a slope of the line between two points, C and D on the on the 2-dimensional [$\mathbf{T}$-E] plane. Therefore, the decision actions are taken by a simple threshold function on [$\mathbf{T}$-E] plane by (8).

From the figure, we can draw the following interesting observations. Firstly, if the distance between T_{u_4} and T_{u_1} is not far enough for $\Lambda_{14} > 0$, i.e positive slope of $\overline{CD}$, then the node will always use the low-end CPU and radios. This is intuitively reasonable since if the processing (communication) time gap between low-end and high-end CPU (radios) is not significantly large enough, then there is no reason to use the power-demanding high-end CPU (radios). Secondly, the node is more likely to use the low-end pair of CPU and radios for higher event arrival rates. For example, if we fix P_0 and increase λ from point A, then the point approaches to the $a_1 = l$ decision region. On the other hand, if we increase P_0 with constant λ, then the point shifts to the $a_1 = l$ decision region.

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.

The analogous geometry analysis for the sensor node with reconfigurable interconnect can be done in Figure 5. As shown in the Figure, the decision actions are taken by multiple threshold functions in (7) on the 3-dimensional [$\mathbf{T}$-E] plane, (T_y, T_z, E).

The energy-efficiency ratios between the dual processor/radio architecture with static interconnect and the simple low or high-end node are defined as $\gamma_{sl} = \frac{g_s^*}{g_l^*}$ and $\gamma_{sh} = \frac{g_s^*}{g_h^*}$ respectively. Likewise, the energy-efficiency ratios between the dual processor/radio architecture with reconfigurable interconnect and the simple low or high-end node are defined as $\gamma_{rl} = \frac{g_r^*}{g_l^*}$ and $\gamma_{rh} = \frac{g_r^*}{g_h^*}$ respectively. We define an energy efficiency gain of the static and the reconfigurable as $\gamma_s = \max(\gamma_{sl}, \gamma_{sh})$ and $\gamma_r = \max(\gamma_{rl}, \gamma_{rh})$ respectively.

Based on the geometric interpretation, we can derive the upper bound of energy-efficiency gain, γ_s and γ_r. Our bound can be obtained by searching the upper bound of vertex angle,θ_v between the half-lines through $(-\frac{1}{\lambda}, -\frac{P_0}{\lambda})$ and (T_{u_k}, K_{u_k})[2]. For example, in Figure 4, the θ_A and θ_B are the one of the vertex angles on the plane.

The maximum θ_v can be found either of fixing λ or $\mathbf{T}_u$. For fixed $\mathbf{T}_u$, it is easily seen that the vertex angles are maximized when the set of rays is a simplicial cone [3];i.e,$(-\frac{1}{\lambda}, -\frac{P_0}{\lambda})$ is the origin. For example, in Figure 5, the $\theta_{34}, \theta_{31}, \theta_{24}, \theta_{21}$ are the angles in the simplicial cone.

For fixed λ, θ_v is bounded by angles between any two-dimensional plane,$K_{\mathbf{u}_k}$ in (5) given $\lambda < T^* = \frac{\|C_4\| - \|C_1\|}{\|P_4\| - \|P_1\|}$ [4];e.g., θ_E in Figure 4.

Therefore, the upper bound of γ are obtained as following.

$$\gamma < \hat{\gamma} = \min \ [\ \max_{i \neq j} \frac{\|T_{u_i}\| K_{u_j}}{\|T_{u_j}\| K_{u_i}}, \ \max_{i \neq j} \| \frac{\mathbf{P}_i}{\mathbf{P}_j} \| \ ] \qquad (10)$$

where $(i, j) \in \{1, 2, 3, 4\}$ for the reconfigurable and $(i, j) \in \{1, 4\}$ for the static.

The power consumption bound for interconnect chip can be obtained by solving $\gamma_r \leq 1$. Considering additional power cost , P_{InC} we can rewrite equation (7) as follows.

$$g_r^* = g(\mathbf{u}_k) = \frac{P_0 + \lambda K_E(\mathbf{u}_k)}{1 + \lambda K_T(\mathbf{u}_k)} + \frac{\lambda P_{InC} \|T\|}{1 + \lambda \|T\|}$$

Let's denote $\tilde{g}_r = \frac{P_0 + \lambda K_E(\mathbf{u}_k)}{1 + \lambda K_T(\mathbf{u}_k)}$. $\tilde{g}_r$ represents the optimal cost when $P_{InC} = 0$. Then the maximum P_{InC} which allows $\gamma_r \geq 1$ is bounded as following.

$$P_{InC} < \ [\ 1 + \frac{1}{1 + \lambda \|T_k\|}) \ ] \ \delta \qquad (11)$$

where $k = argmin_{k=2,3} \tilde{g}_k$ and $\delta = [\ \max(g_4^*, g_1^*) - \min(\tilde{g}_2, \tilde{g}_3) \ ]$.

[2]We note that λ is controllable through an event filtering algorithm in preprocessor

[3]Let $\{A_1, ..A_n\}$ be a linearly independent set of r vectors in $\mathbf{R}^n$. Then $\{y : y = \alpha_1 A_1 + ..\alpha_r A_r, \alpha_1 \geq 0, ..., \alpha_r \geq 0\}$ is known as a simplicial cone of dimension r.

[4]In general, this assumption (also stated in Assumption 2 at section III) hold because we consider wireless sensor network for surveillance application where the inter-arrival time of event is relatively large.

(11) suggests two important intuitions for maximum power consumption of P_{InC}. Firstly, for large λ the first term diminishes so that the second term,δ dominates the upper bound. In geometric interpretation, δ is the maximum angles among $(\theta_{31}, \theta_{34}, \theta_{21}, \theta_{24})$ in Figure 5. Therefore, the upper bound of P_{InC} will be larger for frequent event (large λ). Secondly, δ approaches to 0 for small λ so that the first term dominates the upper bound but the upper bound will be small. This implies that using interconnect chip might not be effective as at frequent event case.

V. Numerical Results

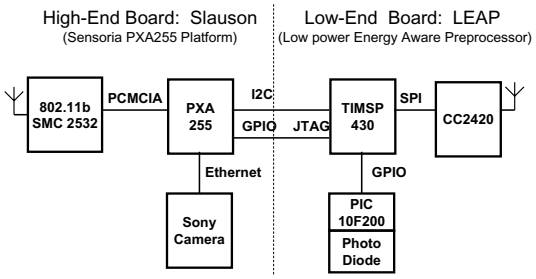


Fig. 6: LEAP architecture

Power	EMAP-TIMSP430		SPM-PXA255	
Mode	Symbol	Power	Symbol	Power
Off	$P^0_{C_l}$	0 mW	$P^0_{C_h}$	0 mW
Active	$P^1_{C_l}$	7.6mW	$P^1_{C_h}$	825mW
Wake-up	$C_{1,l}$	18uJ	$C_{1,h}$	2.2mJ
Power	802.15.4-CC2420		802.11b-SMC2532	
Mode	Symbol	Power	Symbol	Power
Off	$P^0_{R_l}$	0 mW	$P^0_{R_h}$	0 mW
Tx(Max Output)	$P^1_{R_l}$	57.4mW	$P^1_{R_h}$	0.92W
Wake-up	$C_{2,l}$	0.0348mJ	$C_{2,h}$	5mW
Power	Photo diode+PIC		Sony SNC RC30N Camera	
Mode	Symbol	Power	Symbol	Power
Off	$P^0_{S_l}$	0 mW	$P^0_{S_h}$	0 mW
On	$P^1_{S_l}$	0.34 mW	$P^1_{S_h}$	21.6 mW

TABLE III: Power Consumption Specification of component of LEAP

Parameter	Value	Parameter	Value
λ	$0.1/min$	α	1
$\bar{Y}_l$	$1msec \sim 60sec$	$\bar{Y}_h$	$1usec \sim 1sec$
L	$1byte \sim 1Mbyte$	P_{InC}	$1mW$

TABLE IV: Optimization parameters

In this section, we verify the numerical correctness of the formulas and analysis from the previous section. We use the power consumption characteristic of the LEAP platform [4] as a reference platform for the static interconnect architecture and compare the energy-efficiency ratio to an analogous reconfigurable interconnect architecture that uses the same two processors and radios. The LEAP node is partitioned into a general purpose computing module (the Slauson Processor Module)with its associated memory systems and interfaces,

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.

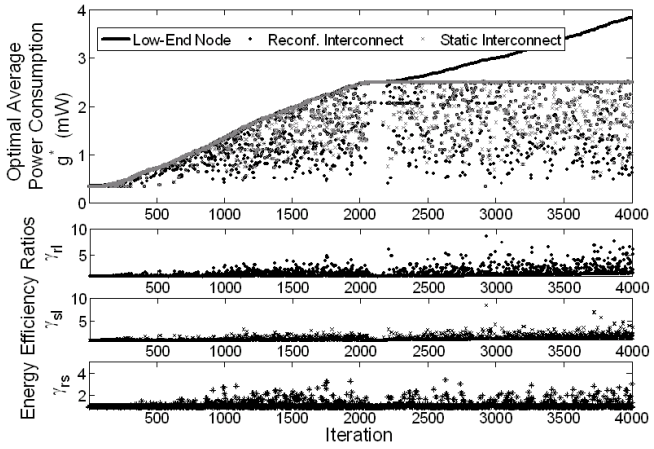


Fig. 7: Energy-Efficiency Comparison ref.to Low-End Node

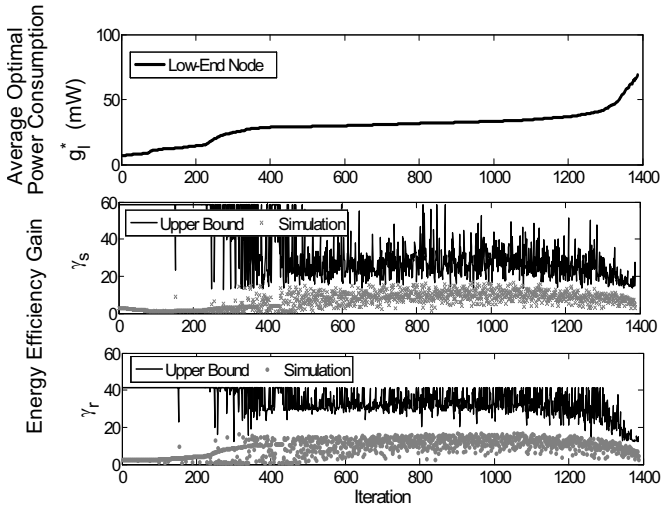


Fig. 8: The Upper Bound of Energy-Efficiency Gain

and a preprocessor module (the EMAP preprocessor) dedicated to low power sensing, energy accounting, and power domain scheduling as shown in Figure 6. In addition to the original LEAP architecture, we assume that a small 8-bit PIC 10F200 microcontroller is attached to the photo diode to act as a preprocessor. We use the limited power modes (Only ON/OFF mode) as shown in Table III.

Energy Efficiency Ratio	Low-End Node Reference		
	Min	Mean	Max
γ_{rl}	1	1.7	8.59
γ_{sl}	1	1.53	8.39
γ_{rs}	1	1.11	3.41
Energy Efficiency Ratio	High-End Node Reference		
	Min	Mean	Max
γ_{rh}	1	1.14	3.60
γ_{sh}	1	1.02	2.85
γ_{rs}	1	1.11	3.49

TABLE V: Energy Efficiency Ratio Comparison

Using the power measurements in Table III, we verify the correctness of decision functions in (7) and (8) given random set of inputs, $\mathbf{x} = (\lambda, Y_l, Y_h, Z_l, Z_h)$. We observe the best energy efficiency gain of the static and reconfigurable interconnect architectures as increasing the average power consumption of simple low-end node (increase g_l^*). The low-end node used in the comparison consists of the same MSP430 processor and a CC2420 as the reference designs of the two architectures.

In order to generate the random set of inputs, $\mathbf{x}$ to increase g_l, we use a non-linear optimization from MATLAB's optimization toolbox ($fmincon$). Our objective is to find a random set, $\mathbf{x}$ to minimize $|g_l^* - g_l|$ given the set of constraints from Table IV. Using the given set of random inputs, we run a discrete event simulation of tracking energy dissipation of a node over large time window with at most $1msec$ timestep resolution. The discrete event simulator selects optimal hardware given $\mathbf{x}$ using the decision functions in (6).The average power consumption is computed by dividing the accumulated energy dissipation by the total elapsed time.

The event inter-arrival times and processing times are generated by the exponential distribution of $\frac{1}{\lambda}$ and $\bar{Y}$ respectively. We also use the path metric for dual-radios, expected transmission time (ETT)[7] in order to simulate the communication times. Let denote B_a and p_a the bandwidth and link loss rate given decision action, $a \in \{l, h\}$ respectively. Then ETT is defined as follows.

$$\bar{Z}_a = \frac{L}{B_a} \frac{1}{1 - p_a}$$

Given $\bar{Z}_a$, data size, L is generated along with the link loss rate, P_a for high-end and low-end radios. The data rates of 802.11 and 802.15.4 radios are $B_h = 20Mbps$,$B_l = 125Kbps$.

Figure 7 shows the simulation results of decision functions, and compares energy-efficiency ratios with respect to our reference low-end node. The top plot in the Figure shows that the energy-efficiency of the dual processor, dual radio architectures is bounded to less than $3mW$ while energy efficiency of the low-end reference node keeps degrading.

The energy-efficiency ratios, for the static and reconfigurable interconnect architectures, γ_{sl} and γ_{rl}, increase as the efficiency of the low-end reference node degrades. The bottom plot for γ_{rs} in the Figure shows that the architecture with the reconfigurable interconnect has better energy-efficiency than its static interconnect counterpart by 3 times. As shown in Table V, the minimum energy efficiency ratios are always 1 which verifies the correctness of the decision functions.

We note that both dual architectures have higher energy-efficiencies (up to 8.59 times) that that of a single processor/radio node using hig-end processor radios. Our result implies the maximum benefit would occur when the node has a high workload.

Figure 8 shows that our analytical bound of energy efficiency ratio, (10) offers reasonably good upper bounds of γ for the dual pair architectures. The top plot in the Figure shows that g_l^* increasing from 1 to 70 mW for iterations at $\lambda = 1/1min$. The second and third plots show that the upper

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.

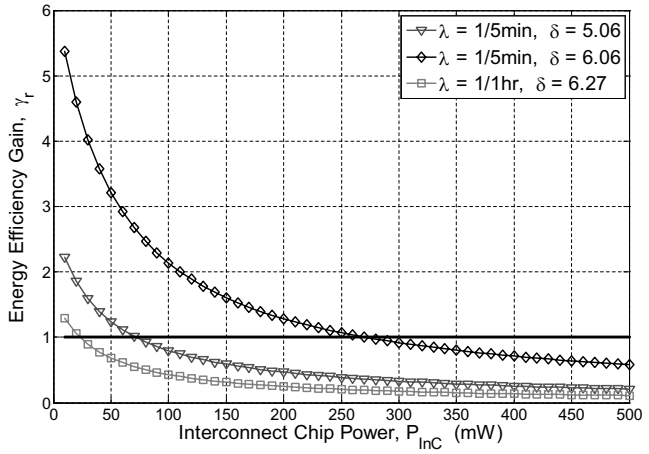


Fig. 9: Energy-efficiency gain w.r.t the power consumption overhead of the interconnect chip

bound range of γ changes dynamically by $T_{\mathbf{u}}$ and also the bound range is limited by $\mathbf{P}$.

Energy Efficiency Gain	Upper Bounding Percentile		
	Max	Mean	Min
γ_s	61.2%	27.0%	1.7%
γ_r	73.2%	30.1%	2.4%

TABLE VI: Energy Efficiency Gain Upper Bound Performance

Table VI shows that the percentile of energy-efficiency ratio over analytical upper bound, i.e $100 \times \frac{\gamma}{\hat{\gamma}}$. As shown in the table, our analytical upper bound bounds $\hat{\gamma}$ close enough at best case, 61.2% and 73.2% for the static and reconfigurable options respectively. On average, our upper bound has 3-4 times larger than γ

Figure 9 shows that the energy-efficiency gain in the reconfigurable case degrades as the power consumption of the interconnect chip increases. When the energy-efficiency gain drops below 1 (the horizontal line in the Figure), using the reconfigurable interconnect architecture will have worse energy efficiency than a single pair node. Given the power consumption measurements of the LEAP node, and the results in Figure 9, we note that to yield a gain the interconnect chip should consume no more than 250mW for the arrival rates considered.

In this particular example we conclude that it is reasonable to pursue the design of a dynamic interconnect since it is very likely that such a deign will consume less than the specified 250mW. For both frequent events ($\lambda = 1/5min$) and rare events ($\lambda = 1/1hour$), the interconnect power consumption bound is well beyond $10mW$. For frequent events, power consumption less than $10mW$ would not significantly degrade energy efficiency of the reconfigurable architecture. The Figure also implies that this architecture would be effective with frequent events since the reduction of power consumption in the interconnect chip will bring significant increase in energy-efficiency ratio compared to the rare event case. As shown in (11), for higher gain difference, δ, we can obtain better energy-efficiency ratio.

VI. Conclusion and Future work

Our work provides the framework to pursue such a design flow for sensor platform with multiple sensors, CPUs, and radios. One of the main problems in such a platform is the policy of switching on and off components to maximize node lifetime (or energy-efficiency). Through our simulations using a set of random inputs for the LEAP node, we have shown that our decision function derived from semi-markov decision model optimally switches on and off high-end and low-end components. In addition, we have shown that interconnect chip allows LEAP node to optimally use low/high-end components to further achieve higher energy efficiency gain. In this way, we demonstrate that LEAP with interconnect chip can potentially increase node lifetime more than 8 times compared to simple low/high-end node in situation where arrival rate is $1/10\ min$.

Although this give a good indication of the bounds, we do note however, that our stationary policy (decision functions) cannot be directly applied on an architecture. To make this possible, a node needs to apply a set of policies which perform well under different workloads and dynamically adjust resource usage according to that policy. We do plan to pursue such a policy as part of our future work in the near future.

Acknowledgment

This work was partially funded through the National Science Foundation award # 0615226 . The authors would also like to acknowledge their collaboators on this project, Mani Srivastava, Deepak Ganesan and Prashant Shenoy for the insightful discussions on this topic.

References

[1] Dimitri P Bertekas. Dynamic programming and optimal control, pp213-218. In *Athena SCientific.*, 2000.

[2] S. Coleri, M. Ergen, and T. Koo. Lifetime analysis of a sensor network with hybrid automata modeling. In *1st ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, September 2002.

[3] D.Jung, T.Teixeira, A.Barton-Sweeney, and A.Savvides. Model-based design exploration of wireless sensor node lifetimes. In *European conference on Wireless Sensor Networks (EWSN) 07*, Jan 2007.

[4] D.McIntire, K.Ho, B.Yip, A.Singh, W.Wu, and W.J.Kaiser. The low power energy aware processing (leap)embedded networked sensor system. In *IPSN '06: Proceedings of the fifth international conference on Information processing in sensor networks*, 2006.

[5] D. Lymberopoulos, Nissanka B.Priyantha, and Feng Zhao. mplatform: a reconfigurable architecture and efficient data sharing mechanism for modular sensor nodes. In *Information Processing in Sensor Networks (IPSN), SPOTS track*, April 2007.

[6] V. Raghunathan, C. Schurgers, S. Park, and M. B. Srivastava. Energy aware wireless microsensor networks. In *IEEE SIGNAL PROCESSING MAGAZINE*, March 2002.

[7] R.Draves, J.Padhye, and B.Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *MobiCom '04: Proceedings of the 10th annual international conference on Mobile computing and networking*, 2004.

[8] B. Schott, M. Bajura, J. Czarnaski, J. Flidr, T. Tho, and L. Wan. A modular power-aware microsensor with 1000x dynamic power range. In *Information Processing in Sensor Networks (IPSN), SPOTS track*, April 2005.

[9] Victor Shnayder, Mark Hempstead, Bor rong Chen, Geoff Werner-Allen, and Matt Welsh. Simulating the power consumption of large-scale sensor network applications. In *SenSys'04*, November 2004.

[10] S.M.Ross. Stocahstic process, second edition, pp213-218. In *San Francisco [Calif.] Holden-Day.*, 1970.

Authorized licensed use limited to: Yonsei University. Downloaded on April 15, 2009 at 08:11 from IEEE Xplore. Restrictions apply.