How does Energy Accounting Matter for Energy Management?

Mian Dong
Samsung Research America
mian.dong@samsung.com

Tian Lan
George Washington University
tlan@gwu.edu

Lin Zhong
Rice University
lzhong@rice.edu

Categories and Subject Descriptors: D.4.1 [Process Management]: Scheduling
General Terms: Management, Theory
Keywords: Energy accounting, energy management

1. INTRODUCTION

Energy accounting determines how much a software process contributes to the total system energy consumption. Given the system energy consumption E and the set of processes $\mathbb{N} = \{1, 2, \dots, n\}$ that are active during time T, *energy accounting* is to determine the energy contribution by each process in $\mathbb{N}$. Energy accounting is the foundation for operating system (OS) based energy management that achieves certain trade-off between energy consumption and process utility by controlling how processes are scheduled, e.g., [4, 3].

In this paper, we provide the first investigation of the relation between energy accounting and energy management and answer the fundamental question about it for the first time: *how does energy accounting matter for energy management?* Toward answering the question, we formulate energy management as a utility optimization problem and show, surprisingly, that energy accounting does not matter for widely used budget based energy management (BEM) framework [4, 3]. In the BEM framework, OS first assigns each running process an energy budget and then charges each process from its budget based on energy accounting. A process will not be scheduled if it runs out of its budget. We further prove that this surprising result is due to the fact that BEM is sub-optimal and subsequently provide an optimal energy management (OEM) framework.

2. ENERGY MANAGEMENT AS UTILITY OPTIMIZATION

2.1 Theoretical Foundation

We first show energy management can be formulated as a utility optimization problem given the energy capacity by scheduling processes, i.e., by determining each process's run time.

Process Run Time: We consider n independent processes, denoted by $\mathbb{N} = \{1, 2, \dots, n\}$, and a system that is capable of running up to m processes concurrently during a scheduler period, i.e., a set of processes $\mathbb{S}$ can be scheduled simultaneously only if $|\mathbb{S}| \leq m$. Suppose that during the time horizon of energy management, a coalition of concurrent processes $\mathbb{S}$ receive total run time $T_{\mathbb{S}}$. Note that $T_{\mathbb{S}}$ does not have to be a single continuous time interval; it can be a collection of time intervals. Then, the run time received by an individual process i is given by

$$t_i = \sum_{\mathbb{S}: i \in \mathbb{S}, |\mathbb{S}| \leq m} T_{\mathbb{S}}. \tag{1}$$

Here the summation is over all feasible process coalitions satisfying $|\mathbb{S}| \leq m$ and containing process i. We refer to $T_{\mathbb{S}}$ as the *scheduling decision* for coalition $\mathbb{S}$.

Energy Capacity: The total system energy consumption must be bounded by available energy capacity, C. The energy consumption of a set of processes $\mathbb{S}$ during a scheduling period T_0 is denoted by $E(\mathbb{S})$, and its average power $E(\mathbb{S})/T_0$. Since $T_{\mathbb{S}}$ is the total time that $\mathbb{S}$ is scheduled, our energy management is subject to the energy capacity constraint, i.e.,

$$\sum_{\mathbb{S}: |\mathbb{S}| \leq m} E(\mathbb{S}) \cdot T_{\mathbb{S}} / T_0 \leq C. \tag{2}$$

Utility Function: In our optimization framework, each process is assigned a utility function $U_i(t_i)$, which measures the utility of process i receiving run time t_i. Such a utility optimization has been widely used in wireless communications and networking [2].

Problem Formulation: Energy management is to make scheduling decisions, i.e., $T_{\mathbb{S}}$, in order to maximize the aggregate utility of all processes, i.e., $\sum_{i=1}^{n} U_i(t_i)$, under energy capacity constraint C.

2.2 Budget based Energy Management (BEM)

We consider budget based energy management (BEM), a widely used energy management approach [4, 3], where each process receives an independent energy budget and disburses it according to a process-specific policy. BEM relies on energy accounting to split total energy consumption among individual processes. Let $\phi_i(E(\mathbb{S}))$ denote the energy consumption attributed to process i according to the energy accounting policy during a scheduling period in which $\mathbb{S}$ are scheduled to run and $i \in \mathbb{S}$. After this scheduling period, $\phi_i(E(\mathbb{S}))$ will be deducted from process i's energy budget B_i. When a process runs out of budget, it will not be scheduled any more.

We assume that each process is scheduled with a known probability p_i, until its budget runs out. To determine process run time, we suppose that processes reach zero energy budget in the following order: $\pi(1), \pi(2), \dots, \pi(n)$, where $\pi(k)$ be the kth process that runs out of its budget (If multiple processes reach zero budget simultaneously, their ordering can be arbitrary). Then, the sequence of positive-budget processes $\mathbb{A}_k = \{\pi(k), \pi(k+1), \dots, \pi(n)\}$ for $k = 1, \dots, n$ form a contraction with $\mathbb{A}_{k+1} \subseteq \mathbb{A}_k$. We can partition the system execution into n distinct intervals, each of length T_k and containing a set of $|\mathbb{A}_k| = n - k + 1$ positive-budget processes.

SIGMETRICS'13, June 17-21, 2013, Pittsburgh, PA, USA.
ACM 978-1-4503-1900-3/13/06.

Now we derive energy consumption and run time for BEM. We assume that the time scale of energy management is much larger than that of a scheduling period. That is, the energy budget for processes are determined for periods much longer than a scheduling period. The Law of Large Number applies and allows us to compute the approximated run time that each feasible subset of processes, $\mathbb{S} \subseteq \mathbb{A}_k$ and $|\mathbb{S}| \le m$, receive during T_k:

$$\begin{aligned} T_{\mathbb{S},k} &= T_k \cdot \text{Prob}\{\mathbb{S} \text{ is selected}\} \\ &= \frac{T_k \Delta_{\mathbb{S},k}}{\sum_{\mathbb{S}:\mathbb{S}\subseteq\mathbb{A}_k,|\mathbb{S}|\le m} \Delta_{\mathbb{S},k}} \end{aligned} \tag{3}$$

where $\Delta_{\mathbb{S},k} = \prod_{i\in\mathbb{S}} p_i \prod_{j\in\mathbb{A}_k\backslash\mathbb{S}}(1-p_j)$ is the probability that only subset $\mathbb{S}$ is selected without constraint $|\mathbb{S}| \le m$. Therefore, each process i receives aggregate run time from all feasible coalitions with $i \in \mathbb{S}$ as follows:

$$T_{i,k} = \sum_{\mathbb{S}:\mathbb{S}\subseteq\mathbb{A}_k,|\mathbb{S}|\le m,i\in\mathbb{S}} T_{\mathbb{S},k} = T_k \frac{\sum_{\mathbb{S}:\mathbb{S}\subseteq\mathbb{A}_k,|\mathbb{S}|\le m,i\in\mathbb{S}} \Delta_{\mathbb{S},k}}{\sum_{\mathbb{S}:\mathbb{S}\subseteq\mathbb{A}_k,|\mathbb{S}|\le m} \Delta_{\mathbb{S},k}} \tag{4}$$

Similarly, process i is charged an energy cost by aggregating its attributions from all feasible coalitions $\mathbb{S}$ during T_k:

$$\Phi_i(\mathbb{A}_k, T_k) \triangleq \sum_{\mathbb{S}:\mathbb{S}\subseteq\mathbb{A}_k,|\mathbb{S}|\le m} \phi_i(E(\mathbb{S})) \cdot T_{\mathbb{S},k}/T_0 \tag{5}$$

The objective of BEM is to maximize the aggregate utility by apportioning system energy budget among individual processes. We formulate it as the following optimization:

Problem BEM :

$$\begin{aligned} \text{maximize} \quad & \sum_{i=1}^{n} U_i(t_i) & (6) \\ \text{subject to} \quad & t_i = \sum_k T_{i,k}, \ \forall i, & (7) \\ & \sum_k \Phi_i(\mathbb{A}_k, T_k) \le B_i, \ \forall i & (8) \\ & \mathbb{A}_k = \{\pi(k), \pi(k+1), \ldots, \pi(n)\}, \ \forall k & (9) \\ & \sum_{i=1}^{n} B_i \le C & (10) \\ \text{variables} \quad & \pi, T_k, B_i. & (11) \end{aligned}$$

where both $T_{i,k}$ and Φ_i are functions of T_k, as given in (4) and (5), respectively. The optimization is carried out over all ordering π of n processes. Note that π and T_k together determine process scheduling, i.e., $T_{\mathbb{S},k}$, $\forall k$ and $\forall \mathbb{S} \subseteq \mathbb{N}$, $|\mathbb{S}| \le m$, as shown in (3).

Energy Accounting Policy does not Matter for BEM: Problem BEM formulated in (6-11) clearly relies on the choice of energy accounting policy $\phi_i(\cdot), \forall i$. However, in the following we prove a somehow counter-intuitive result, which shows that the optimal utility value and process tun time of Problem BEM is irrelevant of energy accounting policies.

THEOREM 1. *Problem BEM is independent of energy accounting, i.e., for an arbitrary energy accounting function that satisfies the Efficiency property, i.e., the energy contributions by all processes must add up to be the same as the total system energy consumption, there always exists a set of energy budget B_i, $\forall i$, which achieve the same optimal utility value.* [1]

The intuitive explanation of the proof is as follows. In particular, we show that for a feasible set of run time $t_i, \forall i$, there always exists a way to partition total energy C among individual budgets $B_i, \forall i$ accordingly, no matter what energy accounting policy is used. Suppose one can achieve an optimal utility value with run time $t_i, \forall i$, using an ordering π and a specific energy accounting policy. Then the corresponding energy budget of each process should be calculated from (5) and (8). If one chooses to use another energy accounting policy and would like to achieve the same optimal utility value, the corresponding energy budget should be calculated in the same way, with the new energy accounting policy. The key idea is that as long as the Efficiency property holds, the sum of individual energy budget would remain unchanged and satisfy the total energy constraint $\sum_i B_i = C$.

2.3 Optimal Energy Management (OEM)

We consider an optimization which aims at maximizing aggregate process utilities under a sum energy constraint C. The Optimal Energy Management problem, denoted by Problem OEM, is formulated as follows:

Problem OEM :

$$\begin{aligned} \text{maximize} \quad & \sum_{i=1}^{n} U_i(t_i) & (12) \\ \text{subject to} \quad & t_i = \sum_{\mathbb{S}:i\in\mathbb{S},|\mathbb{S}|\le m} T_{\mathbb{S}}, \ \forall i, & (13) \\ & \sum_{\mathbb{S}} E(\mathbb{S}) \cdot T_{\mathbb{S}}/T_0 \le C, & (14) \\ \text{variables} \quad & T_{\mathbb{S}}. & (15) \end{aligned}$$

The algorithm for solving Problem OEM and its complexity analysis can be found in [1]. While implementing the solution of Problem OEM in practice requires logging the run time of all feasible coalitions and results in higher overhead than that of BEM, it is guaranteed to outperform BEM in energy management.

THEOREM 2. *Problem OEM always dominates Problem BEM. Its optimal utility value is higher than or equal to that of Problem BEM.*

The intuitive explanation of the proof is as follows. Suppose there exists an optimal solution to Problem BEM. That is, to achieve the optimal BEM utility value, one divides total energy constraint C into individual budgets $B_i, \forall i$, schedules processes based on their budget and priority, and charges them accordingly until their energy budget reduces to zero. If all the scheduling decisions and run time (i.e. $\mathbb{S}$ and $T_{\mathbb{S}}$) are recorded during the entire process, one can apply them in Problem OEM and achieve the same process run time, resulting the same utility value. This is feasible because Problem OEM optimizes over all possible process coalitions and has a significant larger number of "control knobs" than Problem BEM does. Therefore, any optimal BEM utility is achievable in OEM. Optimal OEM utility must be higher than or equal to that of Problem BEM.

3. REFERENCES

[1] M. Dong. *Energy accounting and optimization of mobile systems*. PhD thesis, Rice University, Houston, Texas, 2013.

[2] T. Lan, D. Kao, M. Chiang, and A. Subharwal. An axiomatic theory of fairness for resource allocation. In *Proc. IEEE INFOCOM*, 2010.

[3] A. Roy, S. Rumble, R. Stutsman, P. Levis, D. Mazieres, and N. Zeldovich. Energy management in mobile devices with the Cinder operating system. In *Proc. ACM EuroSys*, 2011.

[4] H. Zeng, C. Ellis, A. Lebeck, and A. Vahdat. ECOSystem: Managing energy as a first class operating system resource. In *ACM SIGPLAN Notices*, volume 37, 2002.

---

[1] Proofs for Theorems (1-2) can be found in [1]