Fairness and Incentive Considerations in Energy Apportionment Policies

EKHIOTZ JON VERGARA, SIMIN NADJM-TEHRANI, and MIKAEL ASPLUND,
Linköping University

The energy consumption of a system is determined by the system component usage patterns and interactions between the coexisting entities and resources. Energy accounting plays an essential role in revealing the contribution of each entity to the total consumption and for energy management. Unfortunately, energy accounting inherits the apportionment problem of accounting in general, which does not have a general single best solution. In this article, we leverage cooperative game theory, which is commonly used in cost allocation problems to study the energy apportionment problem, that is, the problem of prescribing the actual energy consumption of a system to the consuming entities (e.g., applications, processes, or users of the system).

We identify five relevant fairness properties for energy apportionment and present a detailed categorisation and analysis of eight previously proposed energy apportionment policies from different fields in computer and communication systems. In addition, we propose two novel energy apportionment policies based on cooperative game theory that provide strong fairness notion and a rich incentive structure. Our comparative analysis in terms of the identified five fairness properties as well as information requirement and computational complexity shows that there is a tradeoff between fairness and the other evaluation criteria. We provide guidelines to select an energy apportionment policy depending on the purpose of the apportionment and the characteristics of the system.

Categories and Subject Descriptors: C.4 [**Performance of Systems**]: Modeling techniques

General Terms: Design, Performance, Management

Additional Key Words and Phrases: Energy apportionment, energy accounting, cooperative game theory, energy management

ACM Reference Format:
Ekhiotz Jon Vergara, Simin Nadjm-Tehrani, and Mikael Asplund. 2016. Fairness and incentive considerations in energy apportionment policies. ACM Trans. Model. Perform. Eval. Comput. Syst. 2, 1, Article 2 (October 2016), 29 pages.
DOI: http://dx.doi.org/10.1145/2970816

1. INTRODUCTION

Energy consumption and its management is clearly a challenge in computing and communication system design. Energy consumption is obviously of paramount importance for battery-powered devices, but energy efficiency has also become essential for plugged systems due to core drivers such as sustainability and operational expenses.

Even though it can be argued that energy is just another resource subject to common methods for resource management [Neugebauer and McAuley 2001], energy

Authors' addresses: E. J. Vergara; email: ekhiotz.vergara@liu.se; S. Nadjm-Tehrani; simin.nadjm-tehrani@liu.se; M. Asplund; email: mikael.asplund@liu.se; Department of Computer and Information Science, Linköping University, 581 83 Linköping, Sweden.

DOI: http://dx.doi.org/10.1145/2970816

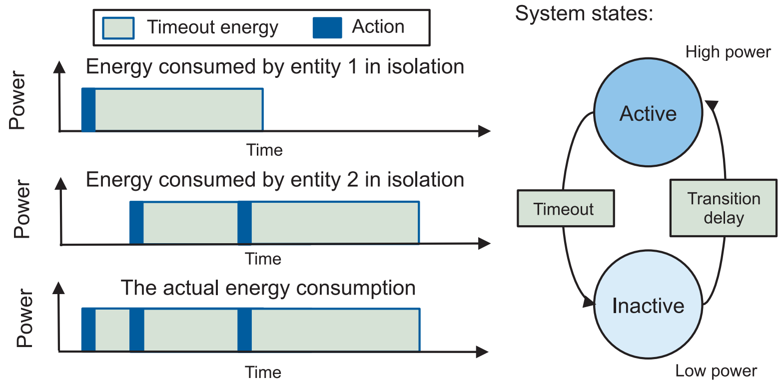

Fig. 1. Energy apportionment example for a common simplification of a system.

management poses its own challenges [Zeng et al. 2002a; Roy et al. 2011]. The energy consumption created by a resource usage is complex since (1) the amount of activity of an entity is often not proportional to the consumed energy, and (2) energy consumption does not necessarily correspond to the *time interval* in which the activity happened. Additionally, the coexisting accesses to shared resources by the different entities in the system aggravate the problem, which results in the actual consumption not being the sum of what the entities would consume using the resource alone.

Energy accounting is the procedure of quantifying, analysing, and reporting the energy consumption of different entities or activities of interest in the system. Efficiently managing energy requires a detailed understanding of the level of resource consumption and energy accounting is a key medium to provide the needed transparency to the system. Energy accounting is also vital to evaluate the energy efficiency of entities in the system. There is evidence supporting the fact that energy awareness can aid in conserving energy [Petersen et al. 2007; Athukorala et al. 2014]. Unfortunately, apportionment is a hard problem of accounting in general that does not have a single best solution for all cases [Lemaire 1984], and energy accounting is no exception.

Selecting an apportionment method, that is, a method to divide the shared energy consumption, entails a set of principles used to achieve some rational division. Thus, we call these methods *energy apportionment policies*, referring to the rule that prescribes the consumed energy to each entity. Examples of entities are different applications using a certain resource, processes of interest or system users. We illustrate the apportionment problem with two instances in the context of mobile and wireless networks, but, as we show later in our work, the problem arises in multiple different contexts.

Figure 1 exemplifies a simple apportionment case. The system is characterised by two states, active and inactive, where an entity can perform an activity only in the active state. After every activity, a timeout is activated that keeps the system consuming energy in the active state for a while. The timeout is a power management mechanism that also serves responsiveness by avoiding the inactive-to-active state transition delay for subsequent activities. Many computing and communication components or systems show a similar behaviour, such as wireless interfaces [Balasubramanian et al. 2009; Vergara et al. 2014b; Pathak et al. 2012; Mittal et al. 2012], hard drives [Papathanasiou and Scott 2003], secure digital cards [Pathak et al. 2012], or even some electric appliances [Reinhardt et al. 2012].

Figure 1 shows the energy consumption of the entities if they were alone in the system (i.e., in isolation). The actual total consumption of the interface is not the sum of entities' consumption in isolation. How should we prescribe a share of the total energy consumption to the different entities in the system? Equally dividing the energy consumption would mean that entity 1 subsidises entity 2. Would a proportional apportionment policy be better? What are the consequences of selecting a method?

Another example is from a wireless sensor network scenario (also applicable to an Internet of Things context) [Fonseca et al. 2008; Kellner 2010], where determining the energy consumption of network nodes due to user-defined activities (e.g., queries) is required. In this scenario, the total energy consumption of the system is determined by the sum of each node's consumption, but the contribution of each node to the total consumption is not necessarily its own consumption. A node contributes to other nodes' energy consumption by being part of the system (e.g., transmissions or beacons). Determining the contribution of each node and the consumption due to the user-defined activities requires a solution to the apportionment problem as well.

Selecting an apportionment policy is a fundamental choice for a system; it not only enables the assessment of the energy consumption of an entity in the system but also, as we show, establishes the "rules of the game" for the system that potentially influence the behaviour of the entities, for example, by providing incentives. The latter aspect is related to fairness of the apportionment that is often hidden at first sight in an energy apportionment policy.

In our work, we show that energy apportionment is a common problem in different areas of computer science, ranging from computing [Flinn and Satyanarayanan 1999b; Ryffel et al. 2009] to data centres [Bertran et al. 2010] to mobile devices [Zhong and Jha 2003; Mittal et al. 2012; Neugebauer and McAuley 2001; Pathak et al. 2012; Roy et al. 2011; Dong et al. 2014] to wireless sensor networks [Sieber and Nolte 2013; Kellner 2010] to energy-efficient buildings [Hay and Rice 2009; Tsao et al. 2014; Thakur et al. 2014; Cheng et al. 2012]. Our work is the first to recognise the common ground of the energy apportionment problem in these areas and builds a unified way to describe and study the problem.

The game-theoretic approach used in this article allows us to formulate the energy apportionment problem as a cost allocation problem, thereby providing new insights into how an energy apportionment policy impacts fairness and potential cooperation incentives for the different entities in the system. The entities can influence each other's energy consumption through cooperation to reduce the total energy consumption. Cooperative game theory has been successfully used to model multiple other real-world practical problems [Fiestras-Janeiro et al. 2011; Moretti and Patrone 2008].

The additional contributions of our work are as follows:

(1) Identify five relevant fairness criteria for energy apportionment. We show that providing incentives to improve the energy efficiency of the system, neglected in previous works, is a relevant purpose of an energy apportionment policy.
(2) Recognise 8 different energy apportionment policies already proposed in different fields to provide a common understanding of the energy apportionment problem. In addition, our work formalises 3 of them not previously presented mathematically.
(3) Propose 2 alternative novel policies based on cooperative game theory not previously used in energy apportionment. These policies provide a strong fairness notion and a rich incentive structure.
(4) Analyse the information requirement and computational complexity of the 10 energy apportionment policies.
(5) Evaluate the fairness of the 10 policies employing the identified five fairness properties.

Our analysis shows that there is a tradeoff between fairness and the other evaluation criteria and provides guidelines to select a policy depending on the purpose of the energy apportionment and the characteristics of the system. In particular, for the five formal

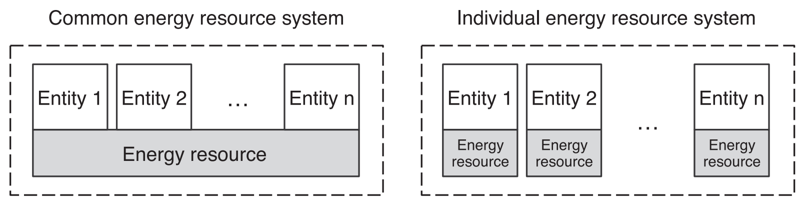

Fig. 2. Illustration of the system models.

fairness properties, we show that the proposed game-theoretical policies provide the strongest fairness. However, these policies have a high information requirement and computational complexity, and thus other simpler policies are identified to provide a satisfactory compromise when the fairest policies become unpractical. We also classify the policies based on the type of input information used and pinpoint that if surrogate information (e.g., usage time) is used in the apportionment, this has to correlate with the energy consumption of the system in order to provide correct incentives to reduce the energy consumption.

Our formulation of the energy apportionment problem using cooperative game theory can resemble cost sharing problems from other fields with extensive literature, for example, economics or social choice [Ghosh and Shah 2015; Moulin and Shenker 2001; Ellis and McGuire 1986], that pursue goals such as economic efficiency. Cost sharing might also be a basis of setting prices or further developing a charging scheme. In contrast, the nature and goal of the energy apportionment problem differs: We need to provide a technical solution to overcome the non-additivity barrier to apportionment (i.e., the sum of entities' consumption in isolation is not equal to the consumption together) and provide transparency to be able to quantify the energy consumed by software entities coexisting in a system.

The rest of the article is structured as follows: Section 2 introduces our system model as well as the background, and Section 3 describes the 10 energy apportionment policies in their application context. Section 4 compares the policies in terms of required information and computational complexity. In Section 5, we leverage game theory to provide a formal study of the fairness characteristics of the different energy apportionment policies. Section 6 summarises and discusses the insights of our analysis, and Section 7 describes the relevant work in the area of energy accounting as well as an overview of energy management. Finally, Section 8 concludes our work and provides future directions.

2. NOTATION AND BACKGROUND

In order to formulate the different energy apportionment policies, we first introduce our system model, a time model employed to formalise some policies and the corresponding notation used in the rest of the article. We also introduce the apportionment policy definition and the required background from cooperative game theory to understand our analysis.

2.1. System Model

An energy consuming system is composed by the set of entities $N = \{1, 2, \dots, n\}$ that cause the system that they are part of to consume energy. We distinguish two types of system interpretations: (1) *common resource*, where the entities share a common energy resource, and (2) *individual resource*, where each entity has an energy resource. Figure 2 shows an illustration of the two types of systems.

Examples of a common resource system are the computers in a datacenter or a battery-powered device shared by different applications, whereas a set of mobile devices or sensor nodes in an ad hoc or sensor network are systems with individual resources.

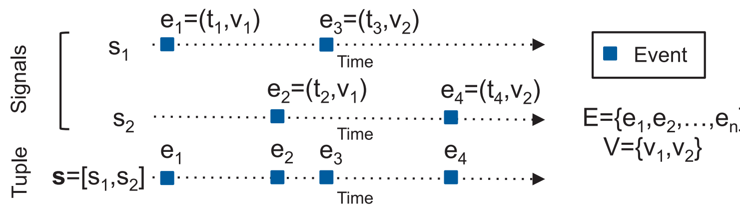

Fig. 3. Example of the time model for two discrete signals and a tuple.

2.1.1. Common Resource System. $E(N)$ is the *total energy consumption* for a system configuration of the N entities sharing a common resource. We assume that equipment to measure $E(N)$ is available, for example, measuring the power consumption of a phone by intercepting the battery terminals [Vergara et al. 2014b] or the mains connected to a home appliance [Cheng et al. 2012].

$E(S)$ is the *isolated energy consumption* of the entities in $S \subset N$. Thus, $E(\{i\})$ is the isolated energy consumption of the single entity $i \in N$. $E(\{i\})$ represents the energy consumption of the entity i in the hypothetical case that it was alone in the system, not considering the interaction of the other entities in the system. This quantity is a construct that is estimated (generally not measurable), for example, a single application running alone on a mobile device or computer.

2.1.2. Individual Resource System. $E_i(N)$ is the *individual energy consumption* of the entity $i \in N$ given a system configuration of entities N. We assume that equipment to measure $E_i(N)$ at each entity i is available. The total system consumption is $\sum_{i \in N} E_i(N)$, for example, the total consumption of a network of sensor nodes is the sum of nodes' consumption in a network configuration.[1]

$E_i(S) : i \in S,\ S \subset N$, is the individual energy consumption of the entity i given a system configuration of entities S. This is a construct that is estimated (generally not measurable). The total energy consumption for a given $S \subset N$ is $\sum_{i \in S} E_i(S)$.

2.1.3. Discrete-Event Time Model. The formalisation of some policies require us to model time, and thus we employ a discrete-event formal framework introduced in the context of real-time systems [Lee 1999]. This section provides an overview of the model.

Atomic events occur along a physical timeline where every event $e = (t, v)$ is defined by a time point $t \in \mathbb{R}$ and a value $v \in V$. The value v is a flexible representation specific to the system.

A *discrete-event signal* s is a set of events and it cannot have two identical events. D is the set of all all signals ($s \in D$). The definition of a functional signal [Lee 1999] states that the events in any s can be enumerated chronologically, there is a single event per t, and that between any two time points there is a finite number of events.

Tuples of signals are used for naming purposes. A tuple $\mathbf{s}$ of N signals is defined as $\mathbf{s} = [s_1, \ldots, s_N]$ where the set of all such tuples is D^N.

Figure 3 shows an example of two discrete signals s_1 and s_2 with two value types (v_1 and v_2) that are part of a tuple $\mathbf{s}$.

2.2. Energy Apportionment Policy

An *energy apportionment policy* Π prescribes the share $\pi_i \in \mathbb{R}$ to an entity $i \in N$ of the total system consumption of the system composed by N entities. A policy may use different input information to apportion the total energy cost among the entities. Equations (1) and (2) define Π for a common and individual resource system, respectively:

[1] $E(N)$ is only applicable to the common resource case.

$$\Pi(E(N), N, \dots) = (\pi_1, \pi_2, \dots, \pi_n) \text{ in } R^n, \tag{1}$$

$$\Pi(E_1(N), \dots, E_n(N), N, \dots) = (\pi_1, \pi_2, \dots, \pi_n) \text{ in } R^n. \tag{2}$$

An apportionment policy should completely apportion the total energy cost: $\sum_{i \in N} \pi_i = E(N)$ and $\sum_{i \in N} \pi_i = \sum_{i \in N} E_i(N)$ for common and individual resource systems, respectively. This is referred to as *efficiency* in the next section.

2.3. Game Theory for Energy Apportionment

Cooperative game theory has been extensively employed as a method to model and solve cost division problems. We show that the energy apportionment problem can be formulated as a cost allocation problem using cooperative game theory in order to determine the energy contribution of the entities in the system.

A transferable utility game (TU-game) in coalitional form is defined by a set of players N (the entities) and a characteristic function c that models cost. A coalition is a subset $S \subseteq N$, where N is the grand coalition and $\emptyset$ is the empty coalition. The characteristic function assigns a cost to each coalition S, $c : 2^N \to \mathbb{R}$, where $c(\emptyset) = 0$ by convention. We employ the *system energy function* E as the characteristic function c to model the energy consumption of the system. This mapping considers the energy consumption of the system to be the total cost of the system. An n-player game is defined by the pair (N, E), where the players will divide $E(N)$ given the power structure in the grand coalition expressed by the characteristic function.

Properties of characteristic functions and their structure classify the games providing insights about the usage of apportionment policies. A cooperative game is *monotone* if $E(S_1) \leq E(S_2)$ when $S_1 \subseteq S_2 \subseteq N$. A game is *subadditive* if $E(S_1) + E(S_2) \geq E(S_1 \cup S_2)$ given $S_1, S_2 \subseteq N$ and $S_1 \cap S_2 = \emptyset$, which means that the coalitions have an incentive to cooperate since they lower the energy consumption (i.e., cost) together. An *essential* game is characterised by $E(N) < \sum_{i \in N} E(\{i\})$. Finally, a game is *concave* if $E(S_1 \cup \{i\}) - E(S_1) \geq E(S_2 \cup \{i\}) - E(S_2)$ for $i \in N$, $S_1 \subset S_2 \subseteq N \backslash \{i\}$. This implies that the incentive for joining a coalition increases when the coalition grows.

Given all possible divisions of the total cost, a single point solution concept for a cooperative game (N, E) provides a unique vector representing the cost division (i.e., apportionment) to each player based on a given criteria. Solution concepts developed for cooperative TU-games can be employed to prescribe a share of the total consumption to each entity in the system. The resulting π can fulfil certain properties. *Efficiency* states that the total energy consumption is completely apportioned: $\sum_{i \in N} \pi_i = E(N)$. We consider efficiency a minimum requirement for an apportionment policy, and, thus, for our analysis all the policies satisfy the efficiency property.

Individual rationality, $\pi_i \leq E(\{i\}), \forall i \in N$, implies that a single player contributes with lower or equal energy consumption when cooperating than being alone. An apportionment is an *imputation* if it satisfies efficiency and individual rationality. We refer to the set of apportionments that satisfy individual rationality and efficiency for a given E as the imputation set $I(E)$.

The game and apportionment properties will be used to compare different energy apportionment policies.

3. ENERGY APPORTIONMENT POLICIES

We first categorise energy apportionment policies based on the type of input information. An *energy-based policy* (EB) apportions the system energy using different constructs with energy as the only type of input information. A *surrogate-based policy* (SB) may employ any type of information except energy. A *hybrid policy* would combine energy and surrogates.

Table I. Summary of the 10 Energy Apportionment Policies and Their Purpose

Nr.	Name	Type	Description
1	Equal division	EB[c]	Apportion the total energy consumption equally to all entities.
2	Proportional to isolation	EB	Apportion proportionally to each entity's isolated energy.
3	Marginal contribution[a]	EB	Apportion proportionally to each entity's marginal contribution.
4	Isolation energy and remainder	EB	Isolated consumption and apportion[b] the rest equally.
5	Shapley value	EB	Apportion the average marginal contribution with respect to all permutations on the set of entities.
6	Nucleolus	EB	Minimise the unhappiness of the most unhappy entity.
7	Least-squares nucleolus	EB	Minimise the variance of all entities' unhappiness.
8	Proportional to isolation	SB[d]	Apportion proportionally to each entity's activity, using some notion of activity (surrogate).
9	Time slicing	SB	For each time interval apportion equally among the active entities. If no entity is active, then apportion equally among all entities.
10	Last active	SB	For each time interval apportion equally among the active entities. If no entity is active, then apportion to the last active entity.

[a]Marginal contribution of an entity: Energy when an entity is running in the system minus when it is not running.
[b]Isolated energy consumption: Energy consumption of an entity if this was alone in the system (see Section 2.1.1).
[c]EB: Energy-based apportionment policy.
[d]SB: Surrogate-based apportionment policy.

This section presents 10 different policies. We survey 5 of the most used energy-based apportionment policies, propose 2 new policies originated from cooperative game theory, and formalise 3 existing surrogate-based policies not described mathematically in earlier work.We briefly explain each policy and provide a brief description of application examples. The application of the policies and their different contexts are explained in Section 7 (related works). Table I summarises the 10 policies and describes in brief their purpose.

3.1. Energy-Based Policies

This section describes seven energy-based policies. All policies can be formulated for common and individual resource systems, but due to space limitation and to simplify the presentation, we present them using the common resource notation.

Policy 1: Equal division: The total energy consumption is divided equally by the number of entities:

$$\pi_i = \frac{E(N)}{|N|}. \tag{3}$$

This apportionment policy is regarded as the simplest policy and has been proposed in the context of energy-efficient buildings [Hay and Rice 2009].

Policy 2: Proportional to isolation: The resulting apportionment is proportional to the isolated energy consumption of each entity. The policy is the same as the pro rata division of $E(N)$ considering the isolated or stand-alone energy:

$$\pi_i = \frac{E(\{i\})}{\sum_{j\in N} E(\{j\})} E(N). \tag{4}$$

The energy consumption of a multicore Central Processing Unit (CPU) is shared by the different entities in proportion to their energy usage in isolation (in a single core) [Ryffel et al. 2009]. The same idea has been also proposed to share the cost of heating, ventilation, and air conditioning (HVAC) systems [Tsao et al. 2014].

The most basic version of this policy is proposed for mobile application energy profiling, where the contribution of an application is assumed to be its consumption in isolation $\pi_i = E(\{i\})$ [Mittal et al. 2012; Neugebauer and McAuley 2001; Vergara et al. 2014b]. However, the basic version does not satisfy efficiency since the total energy consumption is not always the sum of the isolated energy consumption of the entities, and thus we only consider the normalised version of the policy.

Policy 3: Marginal contribution: This policy considers that the contribution of an entity to the total consumption is its marginal contribution $E(N) - E(N\backslash\{i\})$, that is, the total consumption when the entity is running minus the consumption when the entity is not running. The policy is normalised using all the marginal contributions:

$$\pi_i = \frac{E(N) - E(N\backslash\{i\})}{\sum_{j \in N} [E(N) - E(N\backslash\{j\})]} E(N). \tag{5}$$

This policy was proposed for operating systems to account for runtime energy consumption of individual system hardware and software entities in multicore systems [Ryffel et al. 2009; Ryffel 2009].

Similarly to Policy 2, a basic version of this policy was proposed in the area of graphical user interfaces [Zhong and Jha 2003], where an entity is prescribed its marginal contribution $\pi_i = E(N) - E(N\backslash\{i\})$. This basic version does not satisfy efficiency since the sum of marginal contributions is not necessarily equal to the total energy consumption.

Policy 4: Isolation energy and remainder: The policy prescribes the allocation of costs as a function of the energy consumption in isolation plus proportionally dividing the additional cost/benefit of non-isolated use:

$$\pi_i = E(\{i\}) + \frac{E(N) - \sum_{j \in N} E(\{j\})}{|N|}. \tag{6}$$

The usage of this policy is originated in the energy-efficient building community [Hay and Rice 2009], where the personal consumption is estimated first and the rest is divided evenly (e.g., base energy load of building elements such as HVAC). Similarly, in the context of wireless sensor networks, the energy cost of messages received by a certain task is prescribed to the task and the rest (e.g., synchronisation or message loss) is shared evenly among the tasks [Kellner 2010].

Policy 5: Shapley value: Given the marginal contribution of i to a coalition $S \subseteq N\backslash\{i\}$ given by $E(S \cup \{i\}) - E(S)$, the *Shapley value* computes the average marginal contribution of i averaging over all the possible sequences through which the grand coalition can be built from the empty coalition [Shapley 1953]:

$$\pi_i = \sum_{S \subseteq N\backslash\{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [E(S \cup \{i\}) - E(S)]. \tag{7}$$

The Shapley value is a well-known single value solution from game theory applied to many cost sharing problems in computer science [Misra et al. 2010; Sereno 2012] as well as in other fields [Fiestras-Janeiro et al. 2011]. Recently, it has been argued to be the ground truth for energy accounting in mobile devices [Dong et al. 2014].

Policy 6: Nucleolus: Given an imputation $\pi = (\pi_1, \pi_2, \dots, \pi_n)$, that is, an efficient apportionment satisfying individual rationality, the *excess* $e(S, \pi) = E(S) - \sum_{i \in S} \pi_i$ measures the satisfaction of a coalition from apportionment π.

Let $\theta(\pi)$ denote a vector that contains the excess for each coalition $S \subseteq N$ in non-decreasing order given an imputation π. The *nucleolus* is the apportionment π that maximises $\theta(\pi)$ over all the other imputations π^* in the imputation set $I(E)$ using the lexicographic ordering ($\geq_L$) [Schmeidler 1969]:

$$\pi \mid \theta(\pi) \geq_L \theta(\pi^*) \text{ for all } \pi, \pi^* \in I(E). \tag{8}$$

The nucleolus is commonly computed by solving a sequence of linear programming (LP) problems [Maschler et al. 1979]. The *prenucleolus* computes an apportionment following the same method for the set of all efficient apportionments, providing an apportionment when the imputation set is empty.

Policy 6 is proposed in this article for energy apportionment due to its interesting properties (Section 5), and, to the best of our knowledge, it has not been proposed as an energy apportionment policy.

Policy 7: Least-squares nucleolus: For a given imputation π the average excess $\bar{e}$ is

$$\bar{e}(\pi) = \frac{1}{2^{|N|} - 1} \sum_{S \subset N} e(S, \pi).$$

The *least-squares nucleolus* (LSN) is the imputation that minimises $\sum_{S \subset N} (e(S, \pi) - \bar{e}(\pi))^2$, that is, an imputation whose associated excesses are closest to the average excess under the least-squares criterion [Ruiz et al. 1996]. The *least-squares prenucleolus* (LSP) is the efficient apportionment that follows the same method.

The computation of the LSN is performed in two steps: (1) LSP π^P is calculated using Equation (9),

$$\pi_i^P = \frac{E(N)}{|N|} + \frac{1}{|N|\, 2^{|N|-2}} \left[|N| \sum_{S : i \in S} E(S) - \sum_{S \subset N} |S|\, E(S) \right], \tag{9}$$

followed by Equation (2), a recursive algorithm that computes the LSN (π) using the LSP apportionment π^P as input. For each step in the algorithm, a 0 is given to the entities with a negative excess in some of the earlier steps, dividing the aggregated negative excess from the previous step evenly among rest of the entities. The algorithm stops when no entity gets a negative excess, which results in an apportionment π that is an imputation.

To the best of our knowledge, LSN has not been proposed for energy apportionment policies.

3.2. Surrogate-Based Policies

This section employs the time model described in Section 2.1.3 to formalise three previously proposed surrogate-based policies. These policies are selected because we believe that they are good representatives of surrogate-based policies.

Policy 8: Proportional to activity: Given two events that signify that an entity becomes active and inactive, respectively, ($V = \{0, 1\}$, where $v = 1$ for active and $v = 0$ for inactive), the policy apportions energy proportionally to the activity time. Let s_i be the signal of events of entity i, the apportionment policy is defined as follows:

$$\pi_i = \frac{a(s_i)}{\sum_{j \in N} a(s_j)} E(N), \tag{10}$$

where $a(s) : S \rightarrow \mathbb{R}$ is a function that provides the total active time of entity i by performing a sum of the periods between events with values 1 and 0. This policy has been proposed to model the occupancy of a building to divide its energy consumption among the occupants [Hay and Rice 2009].

Policy 9: Time slicing: This policy considers that multiple entities may be active simultaneously and there exists idle time. Given the sequence of events $(e_1, e_2, \dots, e_m)$ temporally sorted in $\mathbf{s}$, a sequence of intervals J is computed. An interval $j \in J$ is given by the delimiting events' time points $[t_j, t_{j+1})$, where $e_j = (t_j, v)$, $e_{j+1} = (t_{j+1}, v')$ and $t_j < t_{j+1}$. T is the period between the first and last events.

The Boolean function $a(i, j) : N \times J \rightarrow \{0, 1\}$ denotes whether an entity i is active during interval j or not. Function $f(i, j) : N \times J \rightarrow [0, 1]$ divides an interval among the active entities proportionally or evenly if the entities are inactive,

$$f(i, j) = \begin{cases} \frac{1}{|N|} & \text{if } \sum_{i \in N} a(i, j) = 0 \\ \frac{a(i,j)}{\sum_{i \in N} a(i,j)} & \text{otherwise} \end{cases}.$$

This function is used to define a policy as described in Equation (11):

$$\pi_i = \frac{1}{T} \sum_{j \in J} \left[f(i, j)(t_{j+1} - t_j) \right] E(N). \tag{11}$$

The policy is used to apportion a micro controller's consumption in a scenario of a distributed database over a network of sensors [Kellner 2010] and a building's energy to tenants [Thakur et al. 2014]. Also, this policy is used for apportionment of a single core processor [Flinn and Satyanarayanan 1999b].

Policy 10: Last active: Given the intervals J, the energy consumption for each interval is apportioned among the active entities. During any interval that all entities are inactive, the energy consumed during the interval is assigned to the *last active entity*.

For each interval j the energy consumption is denoted by $E(N, j)$, where $\sum_{j \in J} E(N, j) = E(N)$. The Boolean function $l(i, j) : N \times J \rightarrow \{0, 1\}$ denotes if entity i is the last active entity of interval j by simply keeping track of the latest "become active" event ($v = 0$).

$$g(i, j) = \begin{cases} l(i, j) & \text{if } \sum_{i \in N} a(i, j) = 0 \\ \frac{a(i,j)}{\sum_{i \in N} a(i,j)} & \text{otherwise.} \end{cases}$$

Policy 10 is defined using function $g(i, j)$:

$$\pi_i = \sum_{j \in J} g(i, j) E(N, j). \tag{12}$$

The policy provides incentives to aggregate the active events of the different entities. This policy is used to apportion the energy tails created by inactivity timeouts for different smartphone components [Pathak et al. 2012].

The following sections compare the presented 10 energy apportionment policies based on three criteria: input information, computational complexity, and fairness. Section 4 analyses the input information and computational complexity, whereas in Section 5 we analyse the fairness of the different policies employing five formal criteria.

4. INPUT INFORMATION AND COMPUTATIONAL COMPLEXITY

The presented policies require different amounts of input information to perform the apportionment, and obtaining this information might be harder depending on the domain of the input and the system. In addition, the amount of input information has an impact on the computational complexity of the policies. We analyse and compare the computational complexity of the policies and provide pointers to efficient solutions in the literature for the most complex policies.

Table II summarises the comparison of the policies using the different criteria and we describe the details of the comparisons next.

4.1. Required Input Information

Section 4.1.2 discusses the amount of information needed for each policy, whereas Section 4.1.1 compares the domain of the input information, that is, surrogate-based and energy-based policies.

4.1.1. Information Type. We categorised the policies based on input type: energy consumption or surrogates. Dividing energy consumption based on energy information appears as a natural choice: if the cost function is known, it is natural to use it to apportion the cost. However, while obtaining $E(N)$ is straightforward, measuring or estimating $E(S) \mid S \subset N$ is complex (if not impossible) given the shared nature and lack of visibility of energy consumption in current systems. We observe that obtaining $E(N \backslash \{i\})$ seems qualitatively harder than $E(\{i\})$ since the first already considers the energy resulting from the use of a shared resource, whereas the second one considers the energy in isolation without any interaction with the other entities. Approximations of $E(S)$ range from accurate energy models [Vergara et al. 2014b; Pathak et al. 2012; Mittal et al. 2012] to measuring some of the observable $S \subset N$ and approximating the unobserved values using regression techniques [Dong et al. 2014].

Surrogate-based policies perform energy apportionment by translating the problem to another domain outside energy consumption. This can simplify the apportionment problem and the data acquisition, for example, surrogate information is easily accessible using performance counters in computers [Bertran et al. 2010] or external activity sensors in buildings [Saha et al. 2014; Cheng et al. 2012]. However, the surrogate input does not necessarily reflect how energy is consumed in the system. For example, the amount of data sent does not reflect the energy consumed in the wireless interface, but it is easily accessible [Vergara et al. 2014b].

4.1.2. Information Quantity. Columns 4 and 5 in Table II show the amount of needed information for each policy. Policy 1 employs the minimum input quantity: $E(N)$ and N. Policies 2, 3, and 4 require the energy consumption of $|N|$ subsets. Policies 2 and 4 consider the energy in isolation $E(\{i\})$ and 3 considers the system energy excluding a single entity $E(N \backslash \{i\})$. Policies 5, 6, and 7 present the highest information requirement, requiring the energy consumption of all the subsets of N ($2^{|N|}$) to build the game.

Apart from $E(N)$, the input of the surrogate-based policies is from another domain than energy consumption. Policies 8–10 require at least a signal s_i per entity (n signals) with the events collected from the system. Additionally, Policy 10 requires $E(N)$ per interval.

Thus, depending on the available information for a system, some policies seem more appealing than others. It is worth mentioning that there exist ways to approximate games with incomplete information (e.g., missing information about the consumption of some subsets) [Polkowski and Araszkiewicz 2003; Masuya and Inuiguchi 2009].

Table II. Overview of the Considered Energy Apportionment Policies

Nr.	Name	Type	Information requirement: Input information	Information requirement: $E(S)$	Comp.	Fairness criteria: S	DP	IR	CR	ED	Contextexample
1	Equal division	EB	$E(N), N$	1	$O(1)$	✓	✗	✗	✗	✓	Buildings [Hay and Rice 2009] Wireless sensors [Fonseca et al. 2008]
2	Proportional to isolation	EB	$E(\{1\}), \ldots, E(\{n\})$	$\|N\| + 1$	$O(n)$	✓	✗(0)	✓	✗(1)	✓	Mobile app testing [Mittal et al. 2012] Single core CPU [Ryffel et al. 2009] Buildings [Tsao et al. 2014]
3	Marginal contribution	EB	$E(N\backslash\{1\}), \ldots, E(N\backslash\{n\})$	$\|N\| + 1$	$O(n)$	✓	✗(0)	✗	✗	✓	Operating systems [Ryffel 2009] Multicore CPU [Ryffel et al. 2009] Mobile GUI testing [Zhong and Jha 2003]
4	Isolation energy and remainder	EB	$E(\{1\}), \ldots, E(\{n\})$	$\|N\| + 1$	$O(n)$	✓	✗	✓	✗	✓	Wireless sensors [Kellner 2010] Buildings [Hay and Rice 2009]
5	Shapley value	EB	$E(S) \mid \forall S \subseteq N, S \neq \emptyset$	$2^{\|N\|}$-1	$O(2^n)$	✓	✓	✗(2)	✗(3)	✗(2)	Mobile devices [Dong et al. 2014]
6	Nucleolus	EB	$E(S) \mid \forall S \subseteq N, S \neq \emptyset$	$2^{\|N\|}$-1	$O(4^n)$	✓	✓	✓	✓	✓	In this article
7	Least square nucleolus	EB	$E(S) \mid \forall S \subseteq N, S \neq \emptyset$	$2^{\|N\|}$-1	$O(2^n)$	✓	✗	✓	✗(4)	✗(5)	In this article
8	Proportional to activity	SB	$s_1, \ldots, s_n$	n.a.	$O(m+n)$	✗(6)	✗(0)	✗	✗	✗(6)	Mobile devices [Neugebauer and McAuley 2001] Buildings [Hay and Rice 2009]
9	Time slicing	SB	$s_1, \ldots, s_n$	n.a.	$O(mn)$	✗(6)	✗	✗	✗	✗(6)	Single core CPU [Flinn and Satyanarayanan 1999b] Wireless sensors [Kellner 2010] Buildings [Thakur et al. 2014] Operating systems [Zeng et al. 2002b]
10	Last active	SB	$s_1, \ldots, s_n, E(N, j)$	n.a.	$O(mn)$	✗	✗(0)	✗	✗	✗	Mobile testing [Pathak et al. 2012]

Type: **EB**: Energy-based policy **SB**: Surrogate-based policy
Fairness comparison criteria and simplified interpretation:
S: Symmetry: entities that contribute equally to system's energy consumption get the same energy apportioment
DP: Dummy player: if the contribution of an entity to all coalitions equals its stand alone energy consumption, this is its apportionment
IR: Individual rationality: an entity gets a lower or equal energy apportionment when cooperating than alone
CR: Coalitional rationality: provides stability since no entity can be apportioned less without penalising other entities and no entity is subsidised
ED: Entity desirability: if the contribution to all coalitions of an entity is smaller than the contribution of another entity, the apportionment for the first entity is less than or equal to the other entity
Notes: 0: The policy satisfies only a subset case of the property when the stand-alone contribution equals 0
1: The policy satisfies the property if a certain inequality is satisfied [Drechsel and Kimms 2010]
2: The policy satisfies the property for subadditive games [Rosenbusch 2011]
3: The policy satisfies the property only for concave games [Shapley 1971; Drechsel 2010]
4: It is contained in an approximation of the core and satisfies another property related to stability (explained in Section 5)
5: The policy satisfies an average variation of the property (explained in Section 5) [Ruiz et al. 1996]
6: The policy satisfies the analogous property considering activity instead of energy

4.2. Computational Complexity

This section presents the computational complexity of the policies. We analyse the general complexity and provide pointers for problem specific complexity reduction for the most complex policies. Column 6 in Table II shows the complexity of each policy. The size of the input (information quantity) is a highly influential aspect in the computational complexity of the policies.

4.2.1. General Complexity. Policies 1 to 4 show low complexity. Policy 1 performs a single operation and its complexity is $O(1)$. Policies 2, 3, and 4 are computed in $O(n)$ given the sum of $|N|$ elements. The computational complexity of Policy 5 increases exponentially with the number of entities $O(2^n)$. Policy 6 is the most complex policy, computed by solving a sequence of at most $|N|$ number of LP problems of decreasing dimension [Maschler et al. 1979] or a single large-scale LP problem with $O(4^n)$ constraints [Puerto and Perea 2013]. The complexity of Policy 7 is $O(2^n)$ due to two steps: (1) the calculation of LSP using Equation (9) involves iterating over the 2^n coalitions, and (2) a maximum of $|N|$ operations are performed by the algorithm to compute LSN.

The complexity of Policies 8–10 depends on the number of events m of all the entities and the functions employed. The complexity of Policy 8 is $O(m+n)$ due to analysing m events when using function $a(s)$ and the sum of $|N|$ entities. Policies 9 and 10 employ $m-1$ intervals and for each interval and entity, and the policies compute the apportionment using functions $f(i, j)$ and $g(i, j)$, respectively, leading to a $O(mn)$ complexity.

4.2.2. Optimisations. There exist approximation techniques for the Shapley value (Policy 5) where the computational complexity increases linearly with the number of players [Fatima et al. 2008; Castro et al. 2009]. There are also exact solutions that take advantage of some particular game representation to reduce the complexity (e.g., $O(n)$ and $O(n^2)$ for a tree [Megiddo 1978] and weighted graph representation, respectively [Deng and Papadimitriou 1994]).

Efficient solutions have been found for the nucleolus (Policy 6) for specific game structures, such as a tree structure where it is computed in $O(n^3)$ [Megiddo 1978] or a weighted graph in $O(n^2)$ [Deng and Papadimitriou 1994]. However, NP-hardness has been shown for other specific game representations (e.g., minimum spanning tree game [Faigle et al. 1998]). Reducing the computation requirement for the nucleolus is an ongoing work.

To sum up, the computational complexity is important to consider when performing energy apportionment of systems, especially with a large number of entities. Policies 5–7 exhibit higher complexity than the rest of policies.

5. FAIRNESS IN ENERGY APPORTIONMENT

Except Policy 5, the application of all the presented policies to apportion energy consumption has been promoted as *fair* by the authors using them (see Table II) with no formal justification. We analyse fairness by first introducing a formal notion of fairness that leverages game theory in Section 5.1. Section 5.2 describes in detail the fairness properties satisfied by each policy, which is summarised in Table II.

5.1. Fairness Criteria

In this section, we propose that fairness can be characterised by selecting five different criteria. These criteria capture properties that are important when apportioning energy consumption in an intuitive and rational manner. We introduce and motivate each criteria for energy apportionment:

—**Symmetry:** If $E(S \cup \{i\}) = E(S \cup \{j\})$ for all $S \subseteq N \backslash \{i, j\}$, then $\pi_i = \pi_j$. If the marginal contribution of i and j to all coalitions is the same, then their apportionment is the same. Symmetry is a desirable property for an apportionment policy since two entities that contribute equally to the energy consumption of the system should intuitively receive the same apportionment.

—**Dummy player:** If $E(S \cup \{i\}) - E(S) = E(\{i\})$ for all $S \subseteq N \backslash \{i\}$, then $\pi_i = E(\{i\})$. If the marginal contribution of entity i to all the coalitions is the same as its stand-alone value, then the latter is his or her value.

The property reflects the fact that the consumption of an entity in the system can be independent of the other entities, and thus it is intuitive to assume that if the entity does not impact the energy consumption of the system or the rest of the entities when sharing the resource, this entity should be prescribed its stand-alone cost. The property also includes the case of when $E(\{i\}) = 0$, which should intuitively lead to a prescription of no cost. This basic case of dummy player is also known as null player.

—**Individual rationality:** This property was briefly introduced in Section 2.3. We recall that an apportionment is individually rational if $\pi_i \leq E(\{i\}), \forall i \in N$. An apportionment policy satisfies individual rationality if it results in such an apportionment when it exists. Recall that when such an apportionment exists it is called an imputation, and the set of such imputations is denoted $I(E)$. It is well known that $\sum_{i \in N} E(\{i\}) \geq E(N)$ iff $I(E) \neq \emptyset$. Hence, the property is formally defined as follows: if $I(E) \neq \emptyset$, then $\pi_i \leq E(\{i\}), \forall i \in N$.

We argue that individual rationality is an appropriate fairness criteria since: (1) When all the entities contribute to lower the total energy consumption compared to the sum of entities' stand-alone consumption, it is unintuitive to prescribe an entity more that it would consume alone, and (2) individual rationality provides incentives to cooperate since every entity's apportionment in the system is lower than what its energy consumption would have been in isolation.

—**Coalitional rationality:** An apportionment is coalitionally rational if $\sum_{i \in S} \pi_i \leq E(S)\ \forall S \subseteq N$. An apportionment policy satisfies coalitional rationality if it results in such an apportionment when it exists. In game theory, the set of apportionments that satisfy coalitional rationality is referred to as the *core*[2] of a game denoted by $C(E)$. Then, coalition rationality is formally defined as:
if $C(E) \neq \emptyset$, then $\sum_{i \in S} \pi_i \leq E(S)\ \forall S \subseteq N$.

This property is relevant since it implies that no entity or coalition is subsidised: If one entity would be prescribed less cost in a coalitionally rational apportionment, then another entity would be penalised, that is, the apportionment is Pareto optimal [Drechsel 2010]. Additionally, this property is attractive for energy apportionment in cooperative systems, since selecting a coalitionally rational apportionment leads to the case where no coalition could reach a better result than in the grand coalition. This introduces a stability notion by avoiding any incentive to divide the grand coalition.

—**Entity desirability:** For the final criterion, we adapt the player desirability criterion introduced by Rosenbusch [Rosenbusch 2011]. Entity i is said to be more *desirable* than entity j ($i \succeq j$) if the energy consumption contribution of i to all coalitions that exclude both entities is less than or equal to the one that j would contribute: $E(S \cup \{i\}) \leq E(S \cup \{j\})\ \forall S \subseteq N \backslash \{i, j\}$. The criterion states that a more desirable entity gets an apportionment less than or equal to the apportionment of the less desirable entity: $i \succeq j \Rightarrow \pi_i \leq \pi_j$.

[2]Even though the core presents desired properties, we do not consider the core as an apportionment policy and use it as a comparison criterion since the core is rarely unique and may be empty.

We refer to i and j as *comparable* if $i \succeq j$ or $j \succeq i$. We believe that the criterion is intuitive, since a lower contribution of an entity than another should be reflected in the apportionment.

5.2. Fairness Comparison

The fairness criteria column in Table II shows whether a policy always satisfies the symmetry, dummy player, individual rationality, coalitional rationality, and entity desirability properties. The 21 shaded cells in the table of the 50 relate to known results based on previous extensive analysis from game theory literature. Table III in the appendix provides the references to the existing results tracing each property to each policy.

The other 29 outcomes are our contributions. These results strictly follow from the respective fairness property and policy definitions. The detailed analysis of the policies in this section is structured per property, where we provide the intuition of why a certain policy satisfies or does not satisfy each property. We refer the interested reader to the electronic appendix (attached at the end of this manuscript) where all the proofs can be found. We also refer to each of our proofs in the text.

5.2.1. Symmetry. All policies satisfy symmetry except for the surrogate-based policies (Policies 8, 9, and 10). The proofs for the symmetry property can be found in Section A of the appendix.

Since Equations (3)–(9) of Policies 1 to 7 do not consider any information other than the value of the system energy function E, two entities contributing with the same energy to all coalitions always get the same apportionment. For example, Policy 1 (Equal division) trivially satisfies symmetry since it always apportions the same value to all entities, or Policy 2 (Proportional to isolation) divides the stand-alone energy by the same value for all entities (Theorem A.1). Similarly, Policy 3 (Marginal contribution) satisfies symmetry since the denominator is equal for all entities (Theorem A.2).

Policies 8, 9, and 10 (surrogate-based policies) do not satisfy symmetry since activity and energy are not necessarily correlated. Thus, even if two entities are symmetric in terms of energy, this is not reflected in the apportionment Policies 8, 9, and 10, which consider activity as input (Theorems A.3, A.4, and A.5 respectively).

5.2.2. Dummy Player. Policies 5 and 6 (Shapley value and Nucleolus) are the only policies satisfying the dummy player criterion. The general reason for not satisfying the property is that the policies perform a certain operation that prevents an entity from getting its isolated energy consumption. The proofs for the dummy player property can be found on Section B of the appendix.

The normalisation factor of Equations (4) and (5) is the main reason for not satisfying the dummy player property for Policy 2 (Proportional to isolation) and Policy 3 (Marginal contribution) (Theorems B.1 and B.2, respectively). Even if an entity contributes with its isolated energy consumption $E(\{i\})$ to all coalitions, Policies 2 and 3 normalise the result, and thus the apportionment does not always result in that isolated energy consumption.

Surrogate-based policies (8, 9, and 10) do not always satisfy the criterion even in the case where the activity function is equal to the system energy function. Policy 8 (Proportional to activity) employs a normalisation factor (similarly to Policy 2) (Theorem B.3). Policy 9 (Time slicing) divides the idle energy equally among all entities (Theorem B.4). Policy 10, Last active, prescribes a greater energy consumption to the last active entity (Theorem B.5).

Finally, policies 2, 3, 8, and 10 satisfy the special case when the contribution of an entity is 0 to all coalitions. For surrogate-based policies, it is intuitive that if an entity does not have a surrogate attribute in a scenario (e.g., it does not have any activity), it does not consume energy in isolation and vice versa. Hence, a(i) = 0 iff E({i}) = 0.

It is easy to verify this so-called null player property for Policy 2 (Proportional to isolation), Policy 3 (Marginal contribution), and Policy 8 (Proportional to activity) by looking at Equations (4), (5), and (10): If the contribution of an entity is 0, the entity is prescribed no energy since $E(\{i\})$, $a(i)$, and $E(N) - E(N\backslash\{i\})$ become 0, respectively, and thus $\pi_i = 0$ (Theorems B.6, B.7, and B.8, respectively). For Policy 10 (Last active), if an entity is never active, then it cannot be the last active entity, and thus it is prescribed no energy (Theorem B.10).

5.2.3. Individual Rationality. Policy 2 (Proportional to isolation), Policy 4 (Isolation energy and remainder), Policy 6 (Nucleolus), and Policy 7 (Least-squares nucleolus) always result in an individually rational apportionment when it exists. The proofs for the individual rationality property can be found on Section C of the appendix.

Policies 2 and 4 (Proportional to isolation and Isolation energy and remainer) always satisfy individual rationality since the apportionment is performed by considering the isolated energy consumption $E(\{i\})$ (Theorems C.1 and C.3).

However, Policy 3 (Marginal contribution) does not always satisfy individual rationality mainly due to Equation (5), which does not consider the energy consumption of $E(\{i\})$ (Theorem C.2). Policy 5 (Shapley value) does not always satisfy this property, but if the resulting game based on the energy consumption values $E(S)$ is subadditive, then Policy 5 will satisfy individual rationality.

Similarly, the surrogate-based policies (Policies 8, 9, and 10) do not satisfy the property since the activity of an entity is not necessarily correlated with energy consumption (Theorems C.4, C.5, and C.6).

5.2.4. Coalitional Rationality. Policy 6 (Nucleolus) is the only apportionment policy that satisfies coalitional rationality when such a solution exists. Coalitional rationality implies individual rationality when $|S| = 1$, and thus Policies 1, 3, 8, 9, and 10, which do not always satisfy individual rationality, do not satisfy coalitional rationality either.

Policy 2 (Proportional to isolation) does not always satisfy coalitional rationality, but it will satisfy the criterion if the following restrictive inequality holds [Drechsel and Kimms 2010]:

$$\frac{\sum_{i \in S} E(\{i\})}{\sum_{i \in N} E(\{i\})} \leq \frac{E(S)}{E(N)} \quad \text{for all} \quad S \subseteq N.$$

Policy 4 (Isolation energy and remainder) does not always satisfy coalitional rationality since cost of the coalitions is not considered in Equation (6) (Theorem D.1). Policy 5 (Shapley value) does not always satisfy coalitional rationality. However, if the resulting game based on the energy consumption values $E(S)$ is concave, Policy 5 satisfies the property [Rosenbusch 2011].

Finally, Policy 7 (Least-squares nucleolus) is not always coalitionally rational, but other works have shown that it is always contained in an approximation of the core (*least-core*), which describes approximately stable outcomes [Maschler et al. 1992; Molina and Tejada 2000]. Additionally, Policy 7 is the only solution ensuring that on average all entities are equally treated if each entity evaluates the excess of the coalitions that the entity belongs to $\sum_{S|i \in S} e(S, \pi) = \sum_{S|j \in S} e(S, \pi), \forall i, j \in N$. The egalitarian principle towards coalitions becomes egalitarian towards the entities, too, and thus this solution contributes to stability.

5.2.5. Entity Desirability. Policies 1, 2, 3, 4, and 6 always satisfy entity desirability. Our proofs for entity desirability can be found in Section E of the appendix.

Policy 1 (Equal division) trivially satisfies it since the policy always prescribes $\pi_i = \pi_j$ (Theorem E.1). Policies 2 and 4 (Proportional to isolation and Isolation energy and remainder) always satisfy entity desirability. A more desirable entity has a lower (or at least equal) isolated consumption $E(\{i\})$ than a less desirable entity, and thus Equations (4) and (6) prescribe always a lower (or at least equal) apportionment (Theorems E.2 and E.4, respectively). Policy 3 (Marginal contribution) always satisfies entity desirability since a more desirable entity has a lower marginal contribution (Theorem E.3).

Policy 5 (Shapley value) does not always satisfy the property for the general case, but the policy will always satisfy entity desirability when the game is subadditive [Rosenbusch 2011].

Policy 7 (Least-squares nucleolus) does not always satisfy entity desirability, but instead it satisfies an attractive property: *aggregated entity desirability*.[3] Instead of comparing the marginal contribution of two entities to each coalition, this property compares the total sum of the marginal contributions to all coalitions and prescribes a smaller cost if the sum of an entity's contribution is smaller than or equal to another entity's contribution [Ruiz et al. 1996]:

$$\text{if} \sum_{S \mid i,j \notin S} E(S \cup \{i\}) - E(S) \leq \sum_{S \mid i,j \notin S} E(S \cup \{j\}) - E(S)$$
$$\text{then } \pi_i \leq \pi_j \text{ for all } i, j \in N.$$

The approach of Policy 7 considers all the coalitions equally important.

Finally, the surrogate-based policies (8, 9, and 10) do not always satisfy entity desirability since less activity does not necessarily mean lower energy consumption (Theorems E.5, E.6, and E.7).

5.3. Summary of the Fairness Insights

Table II shows that Policy 6 (Nucleolus) provides the strongest fairness since it satisfies all properties. The intuition is that Policy 6 is defined to minimise the unhappiness of the most unhappy entity in the system.

Policy 5 (Shapley value) is an excellent and fair alternative provided that the system energy function E is always subadditive for the system. If subadditivity does not hold, then the only advantage of this policy compared to the other ones is that it satisfies the dummy player property. Thus, subadditivity is a precondition for a system to employ the full potential of the Shapley value policy.

Policy 1 (Equal division) and Policy 3 (Marginal contribution) satisfy only symmetry and entity desirability, which makes other policies such as Policy 2 (Proportional to isolation) more attractive. Policy 2 is preferred to Policy 4 (Isolation energy and remainder) since Policy 2 additionally satisfies the subset case of dummy player when the contribution of an entity is 0. Policy 2 is an attractive choice given the symmetry, individual rationality, and entity desirability criteria, which provide incentives and a notion of fairness based on comparing two entities in the system.

Policy 7 (Least-squares nucleolus) is very interesting alternative given its more egalitarian approach to fairness and the additional fairness properties that it satisfies compared to Policy 5 or Policy 6. In addition, all coalitions are equally treated by the policy (i.e., all the complaints have the same weight) and it contributes to stability.

The detailed analysis of the above 10 policies pinpoints that surrogate policies do not provide guarantees in terms of fairness from the energy perspective defined as the

[3] This property is called Average Marginal Contribution Monotonicity, but we rename it to keep term consistency in the evaluation.

five criteria. If the antecedents of the fairness properties are defined in terms of the activity function a instead of E, for example, for symmetry $a(S \cup \{i\}) = a(S \cup \{j\})$ for all $S \subseteq N \backslash \{i, j\}$, then Policies 8 (Proportional to activity) and 9 (Time slicing) satisfy activity-based symmetry and entity desirability. Nevertheless, the two criteria providing incentives (individual rationality and coalitional rationality) inherently require energy-related concepts, and thus activity needs to be correlated to energy, otherwise surrogate-based policies can provide misleading incentives.

6. SUMMARY OF THE ANALYSIS AND DISCUSSION

In this section, we provide an overview of the presented analysis considering the different evaluation criteria and the identified tradeoffs. Section 6.1 summarises the major tradeoffs between fairness and the other evaluation criteria. Section 6.2 discusses the importance of analysing the system energy function E, and, finally, Section 6.3 provides some high-level guidelines to select an energy apportionment policy given the purpose of the apportionment.

6.1. General Tradeoffs

The major tradeoffs identified in the analysis are the following:

—**Fairness vs. information type:** We have distinguished between energy-based and surrogate-based energy apportionment policies based on the type of input information. Surrogate information (e.g., resource access time or data sent) is in general easily accessible compared to energy-based information, which usually needs to be estimated.

However, we have shown that some fairness properties (dummy player, individual rationality, and coalitional rationality) cannot be satisfied by a surrogate-based policy. In addition, if the surrogate information is not highly correlated with the energy information, the surrogate policies will not always satisfy the other fairness properties (symmetry and entity desirability). Table II reflects this issue showing that none of the surrogate-based policies satisfies any of the five fairness properties in the general case.

—**Fairness vs. computational complexity:** The amount of input information is tightly related to the computational complexity of a policy, which leads to a tradeoff between fairness and computational complexity.

Policy 1 (Equal division) provides the lowest complexity but also weak fairness. Policies 2, 3, and 4 present low information requirement (a single value per entity), leading to a low computational complexity, but none of these policies satisfies the 5 fairness properties. We identified Policy 2 (Proportional to isolation) as an interesting policy in the tradeoff between fairness and computational complexity: It provides reasonable fairness (providing symmetry and entity desirability) as well as incentives (individual rationality) in linear time.

Surrogate-based policies (Policies 8, 9, and 10) are not very attractive since they do not satisfy any fairness property and the computation requirement is still similar to Policies 2, 3, and 4.

Satisfying the additional fairness properties requires more input information (e.g., to compare coalitions), leading to higher computational complexity. Policy 6 (Nucleolus) satisfies all fairness properties, and Policies 5 and 7 (Shapley value and Least-squares nucleolus) have a well-defined notion of fairness. However, the size of the input grows exponentially with the number of entities for the game-theoretic policies. Moreover, coalitional rationality (satisfied by Policy 6) relates to an inherently hard problem. Determining whether there exists a set of coalitionally rational apportionments is NP-complete [Conitzer and Sandholm 2003, 2006].

Thus, the computational cost of satisfying all fairness properties is high, and lowering the computational cost requires selecting a smaller set of fairness properties. Alternatively, one could apply a high complexity approach to a problem with a small set of inputs. Moreover, as pointed out in Section 4.2.2, there are efficient algorithms to compute the game-theoretic policies for restricted classes of games.

To sum up, fairness comes at at cost of having to measure or estimate energy consumption as well as computational complexity.

6.2. Properties of the System Energy Function

The system energy function E models the energy consumption of the system and its values are either measured or estimated. In Section 2.3 we introduced some common properties used to classify E (e.g., subadditive or concave). Knowing the class of E is highly valuable for Policy 5 (Shapley value) since it satisfies relevant fairness properties only for some of the E categories.

While subadditivity is often an assumption for some problems in the game theory literature [Shapley 1953; Engevall et al. 1998], it is difficult to say how likely it is that the system function E is subadditive for energy systems. If E is subadditive, then the energy consumption of two coalitions (respectively, entities) together is not higher than the sum of their isolated consumption. This implies that being together in the system is always advantageous, which implies that the interaction of the entities in the system is efficient. While analysing whether an instance of E is subadditive is intuitively system dependent, we provide some illustrative examples that result in a subadditive E and a non-subadditive E, respectively:

—**Subadditive:** Consider a system similar to the one presented in the introduction section, where major contribution to the energy consumption is due to an inactivity timer triggered after performing some activity. For example, a secure digital card (SD card) consumes energy for some seconds after each write/read operation [Pathak et al. 2012]. If two applications use the SD card concurrently, then the total energy is smaller than sum of the stand-alone energy of the applications (overlapping energy tails). If the applications use it separately, then the consumption is simply the sum of the stand-alone energy. Thus, the system is subadditive. Other examples following a similar intuition are a water kettle or a TV [Reinhardt et al. 2012], where the usage overlap by several users intuitively results in a subadditive E.

—**Non-subadditive:** Consider the 3G wireless interface that is often characterised by two main consuming states depending on the utilisation [Vergara et al. 2014b]. The drivers of some WiFi interfaces use a similar mechanism [Pyles et al. 2012]. Small amounts of data are sent in a low consuming state, but when the data sent for a period of time is higher than a given threshold, the interface is moved to a high-energy state. The consumption of two processes sending small amounts of data together can lead to the high-energy state, whereas sending data separately would result in a lower consumption due to using the low-energy state separately. Thus, this system is not always subadditive.

Consider another simple example from the energy-efficient building area, a refrigerator, whose compressor is activated periodically in order to maintain the desired temperature in the hysteresis region [Cheng et al. 2012]. Opening the refrigerator increases the need of starting the compressor earlier. For this system, the result of two people opening the refrigerator separately can easily result in a non-subadditive E since the entities alone would consume less than together.

Concavity implies subadditivity, and, hence, from the discussion above, it follows that some real-world examples of E may not satisfy concavity.

To sum up, we believe that further work is needed to analyse the properties of E for different types of systems. It is also interesting to analyse the properties of E for complex systems composed of different subsystems (e.g., a mobile device has an SD card as well as some wireless interfaces).

6.3. Selection Guidelines

In this section, we outline the most significant criteria that impact the selection of a policy for energy apportionment. Based on our analysis and the previous discussion, the following criteria serve as guidelines in order to select a policy:

—**Surrogate- vs. energy-based policy:** Where applicable, we believe that energy-based policies are more appropriate than surrogate-based ones. Highly accurate energy models and measurement techniques are already available for many systems and obtaining better estimates of the E values is an ongoing work. This will facilitate the information acquisition for the apportionment.

—**Number of entities and real-time requirements:** The number of entities in the system plays an important role on the computational complexity, and this can drastically impact the timing requirements for energy apportionment. In general, complex algorithms are not fit for large numbers of entities, especially in real-time applications of apportionment. However, they may make sense in offline analysis settings. Analysing the running time of the apportionment policy implementations in different systems as well as their energy cost is an interesting future direction.

—**The purpose of apportionment:** We classify the purpose of apportionment into three main categories: informing, profiling, and incentivising.

  —**Informing:** When the purpose of the energy apportionment is simply to inform about the overall energy consumption of the entities (e.g., rank the most consuming entities), there is no strict need for a complex policy. We believe that Policy 2 (Proportional to isolation) is a good fit for this purpose since it includes a fairness notion where each entity is compared in a one-to-one fashion (symmetry, null player, and entity desirability).

  —**Profiling:** This applies when the purpose of the apportionment is to accurately profile the energy usage (via testing) by entities in a system, for example, for improving the efficiency of applications or software optimisation. Symmetry and entity desirability are necessary properties for this purpose, and the dummy player property is desirable. Table II shows that Policy 5 (Shapley value) and Policy 6 (Nucleolus) are good choices for this purpose.

  —**Incentivising:** Energy apportionment can provide incentives to reduce the energy consumption, especially when the entities of the system can influence their energy consumption. Basic incentives can be realised by selecting a policy that satisfies individual rationality, such as the simple Policy 2 (Proportional to isolation). It provides incentives to reduce the stand-alone consumption of an entity, since this will be apportioned less than its stand-alone consumption when the system is efficient. For example, it can incentivise application developers to reduce the stand-alone energy of their applications.

  Satisfying coalitional rationality is recommended when the purpose is to incentivise entities to efficiently cooperate in the system in addition to reduce their stand-alone consumption. For example, two applications can be incentivised to reduce their consumption together (e.g., by synchronising transmission patterns) because they are apportioned no more than their stand-alone consumption but also no more than their consumption together. Coalitional rationality is desirable in systems where multiple dynamic entities can join or leave the system (e.g., device-to-device clusters in 5G [Bianzino et al. 2014; Boccardi et al. 2014]), where incentives are the key to cooperation.

Thus, a system can benefit from a strong incentive structure of Policy 6 (Nucleolus), which satisfies individual and coalitional rationality, at the cost of computational complexity. A more egalitarian fairness notion towards all coalitions of Policy 7 (Least-squares nucleolus) is also an interesting alternative when users are involved (e.g., energy-efficient buildings).

Finally, if the subadditivity and concavity of E can be shown, Policy 5 (Shapley value) can also provide incentives.

Thus, depending on the purpose of energy apportionment and the system specifics, some policies can be a better fit than others. We have identified that Policy 2 (Proportional to isolation) provides an interesting tradeoff between computational complexity and fairness while still providing basic incentives. Policy 6 (Nucleolus) and Policy 7 (Least-squares nucleolus) are the most interesting properties in order to provide stronger incentives, and Policy 5 (Shapley value) and Policy 6 (Nucleolus) are a good fit to profile systems. Finally, it is clear that analysing E is crucial for employing Policy 5 (Shapley value).

7. RELATED WORKS

Our work has studied the energy apportionment problem from a perspective not adopted before. In Section 7.1 we describe how different works have practically approached energy apportionment and describe their context as well as the implemented policy. In addition, Section 7.2 provides a brief overview of the area of energy management and how energy apportionment plays a highly relevant role on it.

7.1. Energy Accounting

This section is devoted to describe the application and implementation of the presented energy apportionment policies. We briefly explain each approach and analyse and report on which policies they employ. The works are categorised in four different contexts: processing and operating systems, wireless sensor networks, energy-efficient buildings, and mobile devices.

7.1.1. Processing and Operating Systems. PowerScope [Flinn and Satyanarayanan 1999b] maps energy consumption to program structure by employing physical power measurements combined with kernel-level system activity information. The work studies a single-core CPU and thus attributes a process the energy consumed when the process is being executed. Since a single process can be executed simultaneously, the process allocates its consumption based on the time slice when it was scheduled (similar to Policy 9, Time slicing). This work does not consider cases where the activity of entities in the system is not instantaneously reflected in the power trace (e.g., energy tails).

A follow-up work by Bellosa [2000] proposes Joule Watcher, a similar event-driven resource profiler to attribute energy consumption of CPU and memory to individual threads. Even though the work does not adress the problem of multi-entity apportionment, it refers to surrogate information such as performance-monitoring information.

Neugebauer and McAuley [2001] argue that energy accounting of shared resources is a requirement for energy management. They discuss the difficulties in estimating each entity's consumption due to asynchronous energy overheads (e.g., energy tails). They suggest that the apportionment should be done in proportion to utilisation-based surrogates, that is, Policy 8 (Proportional to activity). Examples of surrogates include the access time to a resource, the number of pixels used in a display, or the amount of data exchanged with the network.

ECOSystem [Zeng et al. 2002b] is an energy-centric operating system that employs energy accounting for the CPU, disk, and network card. The running time of the process is used to apportion the CPU consumption assuming that the CPU consumes a fixed amount of power (Policy 9). The active access of the disk is computed using surrogates when write or read operations occur. The consumption due to asynchronous operation (e.g., spin-up, spinning, and spin-down) is proportionally divided by the tasks that employed it between the spin-up and spin-down events. This is similar to Policy 9 (Time slicing) considering the spin-up and spin-down as events. Finally, the energy consumption of the network interface is attributed based the amount of data transmitted and received.

Ryffel et al. [2009] and LEA^{2}P [Ryffel 2009] extend the idea of PowerScope to multi-core system and propose Policy 3 (Marginal contribution) as fair without any formal justification. The system employs power measurements of different hardware elements (e.g., CPU, Graphics Processing Unit (GPU), or hard drive) combined with indirect information based on performance counters. However, they mention that obtaining the energy consumption values for Policy 3 is not feasible for some system components such as CPU since processes run simultaneously. Estimating the energy consumption of a process in isolation (in a single CPU) is proposed to overcome the limitation, and thus Policy 2 (Proportional to isolation) is employed to estimate the apportionment of each process for the multi-core processor.

Joulemeter [Kansal et al. 2010] is a system to attribute the energy consumption of a cloud server to the running virtual machines (VMs). The energy consumption of each VM is estimated based on measurement-based energy models. It is argued that the apportionment is correct if the sum of the estimated energy is equal to the total measured energy (i.e., efficiency), which is not always the case. They consider the energy in isolation as the apportionment for each VM (similar to Policy 2).

7.1.2. Wireless Sensor Networks. Quanto [Fonseca et al. 2008] is an energy profiler for network nodes that combines energy metering hardware and program activity information to quantify the energy consumption for user-defined activities. They use the notion of activity as a logical set of operations whose resource usage is grouped together. An entity can adopt the same notion in our work.

Quanto estimates the energy breakdown of the different components and then system quantifies the resource consumption of the activities. A single-activity device is considered, leading to a straightforward apportionment, but the authors mention that the apportionment for a multiple-activity device is a policy decision. They preliminarily select to divide the energy consumption equally, that is, in a similar way to Policy 1.

Kellner [2010] considers the context of dynamic sensor networks where various applications can query a sensor network at any point in time. The work proposes an online energy accounting system employing measurement-based finite-state machine (FSM) models to attribute energy consumption to tasks. Since energy consumption not only depends on the active usage (e.g., energy tails or between uses), the author identifies several ways to divide the remainder energy. Policy 4 (Isolation energy and remainder) is applied to message reception where the energy consumption of the messages that cannot be attributed directly to any task is equally shared. Policy 9 (Time slicing) is used to divide the energy consumption of the micro controller considering active or inactive tasks.

7.1.3. Energy-Efficient Buildings. A first and influential work by Hay and Rice [Hay and Rice 2009] formulate the problem of attributing the energy consumption of a building or organisation to individual users. They argue that it can provide incentives to reduce the total consumption and identify two requirements that every policy should satisfy: completeness and accountability. The first one refers to efficiency, whereas the second states that actions of a user should have a maximal effect on their own share and a minimal on the rest of users. The entity desirability property is related to accountability, where a more consuming entity will get prescribed a higher energy consumption.

Their work proposes Policy 1 (Equal division) as the most basic policy. This is further extended using occupancy time as a surrogate of energy resulting in a version of Policy 8 (Proportional to activity). Policy 4 (Isolation energy and remainder) is then proposed by first calculating the personal load of a user and then dividing the rest evenly. Their efforts focus on accurately estimating the personal load employing sensors.

Similarly, WattShare [Thakur et al. 2014] seeks to attribute the energy consumption of a shared building to individual occupants (*personal apportionment*) using Policy 9 (Time slicing). Considering only the total energy consumption measured at the smart meter, their work proposes to employ information from smartphone sensors (e.g., WiFi signal strength or microphone data) carried by the occupants to perform the apportionment.

Tsao et al. [2014] investigate how to attribute the energy consumption of centralised heating, ventilation, and air conditioning system to different rooms in a building. They propose to employ Policy 2 (Proportional to isolation) and argue that it satisfies efficiency and accountability. There is no evidence showing that their apportionment approach satisfies the two properties.

7.1.4. Mobile Devices. Pushed by short battery lifetimes energy profiling has attracted a great attention in mobile devices to improve software energy efficiency. Our previous work, EnergyBox [Vergara et al. 2014b], develops an FSM-based energy model for wireless interfaces. By isolating the traffic of an application, the energy efficiency of the applications is studied in isolation using EnergyBox [Vergara et al. 2014a; Almquist et al. 2015]. Other studies perform a similar simplifying assumption to profile the energy consumption due to communication [Qian et al. 2011; Rice and Hay 2010], which is similar to Policy 2 (Proportional to isolation).

Similarly, WattsOn [Mittal et al. 2012] is a system to estimate the energy consumption by an application at the development environment that reuses and extends previously developed FSM-based energy models. This work considers that the energy consumption of the application is the total energy consumption of the system when only the application is running (i.e., similar to Policy 2). Zhang et al. [2010] develop activity-based power models (FSM and linear functions) using measurements in order to attribute the energy consumption of each component to the different processes in the system. Their work does not consider the asynchronous energy consumption such as energy tails.

Bugu [Li et al. 2014] is an application-level service that aims to profile the applications running in client devices by sending consumption statistics to a server. The considered apportionment policy is that an application's consumption is the energy consumption in isolation ($E(\{i\})$) based on their power model (similarly to Policy 2), which does not always satisfy efficiency.

The characterisation of graphical user interfaces and display energy consumption from hardware, software, and application perspective is performed with a focus on optimising the energy consumption of Graphical User Interface (GUI) applications and GUI platforms [Zhong and Jha 2003]. The work uses Policy 3 (Marginal contribution) to obtain the energy contribution of GUIs.

Eprof [Pathak et al. 2012] is an energy profiler for smartphones to support efficient application development. It employs system call-based resource usage information and FSM-based models to attribute the energy consumption to applications (and activities within the application). They discuss different alternatives to attribute the energy consumption of energy tails such as equally (Policy 1) or proportionally dividing the energy consumption among the consuming entities (Policy 2). Finally, they propose the *last-trigger policy* (Policy 10, Last active in our work) in order to incentivise the developer to batch energy events. In our work, we formalised their policy proposal to analyse its properties.

The recent work by Dong et al. [2014] is the first work considering the application of cooperative game theory in the context of mobile energy accounting. They argue that the Shapley value (Policy 5) is the ground truth for the apportionment problem in mobile systems and provides an approximation of it by approximating the game. Using physical power measurements, they define every 10ms interval as a game where the energy consumption needs to be attributed. By considering these short intervals, they attempt to observe the consumption of different coalitions (i.e., observing intervals where only some entities are active) and employ a combination of the observed coalitions with hardware states (for WiFi and CPU) to reduce the variance of the observations. The unobserved coalitions are statistically estimated from historical data. Their results show that other accounting methods differ significantly from their approach.

Recent work has also focused on identifying energy-hungry applications by collaboratively collecting usage information from many devices and employing statistical techniques [Oliner et al. 2013; Chandra et al. 2013]. These works provide a limited level of energy accounting due to the coarse granularity of the approach that reduces the possibilities of using them in energy management.

Compared to the previous works, we focus on analysing the fundamental aspects of energy apportionment policies rather than application aspects of them. Our work provides an overview of the energy apportionment problem in different research areas in order to unify efforts towards energy efficiency.

We show that fairness is a relevant criterion and pinpoint that, as in cooperative game theory, there is no best solution in general. Additionally, we propose the usage of two apportionment policies with strong fairness notions that have not previously been used in the described areas. Finally, we provide guidelines to select a policy based on the information requirement, computational complexity, and fairness properties and suggest that the analysing structure of the E function for a system is decisive.

7.2. Energy Management

Efficiently managing system resources in terms of energy requires u to control and determine the energy expenditure of the system and the entities. Energy accounting, and, thus, energy apportionment, too, are key elements to determine the energy consumption of the entities. This section describes approaches considering energy allocation to competing entities in the system, mostly from the operating system and mobile device literature. These rely on energy accounting and apportionment policies.

The first energy-aware operating system concepts and attempts date from the late 1990s [Ellis 1999; Neugebauer and McAuley 2001]. Ellis [1999] advocates that energy consumption should be considered a first-class resource while still meeting performance requirements. Since then, the main approach in energy management is to devise different strategies to control the energy consumption of the system by limiting the energy usage of entities in the system (e.g., applications).

Budget-based energy management (BBEM) allocates an energy budget to entities in the system (e.g., applications), and these will not run if they run out of budget. Setting a target lifetime is also a recurring trend. Nemesis [Neugebauer and McAuley 2001], ECOSystem [Ellis 1999; Zeng et al. 2002b, 2003], and Odyssey [Flinn and Satyanarayanan 1999a, 2004] are examples of early operating systems that propose adaptive applications that can trade off Quality of Service (QoS) for lower consumption. These works also pinpoint the need for energy accounting. For example, Odyssey uses PowerScope [Flinn and Satyanarayanan 1999b] to build a system's power model and employs Policy 8 as apportionment policy. More recently, BBEM has been adopted and extended by several works to extend the battery lifetime of mobile devices (e.g., Cinder [Rumble et al. 2009; Roy et al. 2011]). The complexity of BBEM lies on devising a strategy to define the budgets and the replenishment rate to overcome the potential QoS and Quality of Experience (QoE) burden on the user when applications run out of budget. Prioritising some applications [Cho et al. 2014], limiting background applications [Chandra et al. 2013], or exploiting context information (e.g., user location) [Chu et al. 2011; Vallina-Rodriguez and Crowcroft 2011] are some examples of recent proposals for energy management.

Finally, the theoretical work by Dong [2013] formulates energy management as a utility optimisation problem: Schedule the different processes to maximise the aggregate utility of all system processes under the energy capacity constraint. BBEM is compared to a novel energy management strategy that dominates by optimal utility value.

To summarise, energy is a vital resource, and efficiently managing system resources from an energy perspective is an ongoing activity. Selecting an appropriate energy apportionment policy for a system is important for efficient energy accounting and energy management.

8. CONCLUSIONS AND FUTURE WORK

In this article, we categorised, compared, and analysed a total of 10 different energy apportionment policies in terms of required information, computational complexity, and fairness. Among the 10 policies, we formalised 3 policies not described mathematically earlier and proposed two novel policies based on cooperative game theory.

Our work shows that cooperative game theory provides an excellent framework and solutions for addressing the energy apportionment problem and that selecting an appropriate policy plays a role in providing incentives for energy efficiency to the entities in the system.

Unfortunately, there is no known ground truth for the general apportionment problem, and, therefore, energy-based policies are no better than surrogate-based ones. However, we have pinpointed that providing incentives and stability requires energy-related concepts and that an energy apportionment policy needs to reflect the inner workings of the system leading to energy consumption. Thus, employing the system energy function in the apportionment seems a more reasonable alternative.

Our results demonstrate that there is a tradeoff between fairness and the other evaluation criteria. The strongly fair policies based on game theory have the highest information requirement and computational complexity, which can be less practical in some contexts (e.g., high number of entities in the system or timing requirements). Depending on the context, a simpler policy like Policy 2 (Proportional to isolation) can achieve a satisfactory compromise by sacrificing some fairness properties but drastically reducing complexity and still providing basic incentives. Further work can be done to identify how to satisfy a set of fairness properties with minimum complexity

Finally, the characteristics and properties of the system energy function should be analysed since some policies assume certain properties (e.g., Shapley value), and sometimes different policies can lead to identical results. In the latter case, selecting the simplest policy among the equivalent ones is recommended.

The future work directions include applying the above insights in systems from different contexts and empirically studying the impact of the incentives on energy efficiency. Developing means to measure or better approximate the system function for different systems is a key capability that can provide insights about the structure of the system function.

ELECTRONIC APPENDIX

The electronic appendix for this article can be accessed in the ACM Digital Library.

ACKNOWLEDGMENT

This work was supported by the Swedish national graduate school in computer science (CUGS). The authors thank Holger I. Meinhardt and Artus Ph. Rosenbusch for their insightful comments about game theory.

REFERENCES

Mathias Almquist, Viktor Almquist, Ekhiotz Jon Vergara, and Simin Nadjm-Tehrani. 2015. Communication energy overhead of mobiles games. In *Proceedings of the 2nd Workshop on Mobile Gaming (MobiGames'15)*. ACM, 1–6. DOI:http://dx.doi.org/10.1145/2751496.2751498

Kumaripaba Athukorala, Eemil Lagerspetz, Maria von Kügelgen, Antti Jylhä, Adam J. Oliner, Sasu Tarkoma, and Giulio Jacucci. 2014. How carat affects user behavior: Implications for mobile battery awareness applications. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'14)*. ACM, 1029–1038. DOI:http://dx.doi.org/10.1145/2556288.2557271

Niranjan Balasubramanian, Aruna Balasubramanian, and Arun Venkataramani. 2009. Energy consumption in mobile phones: A measurement study and implications for network applications. In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference (IMC'09)*. ACM, 280–293. DOI:http://dx.doi.org/10.1145/1644893.1644927

Frank Bellosa. 2000. The benefits of event-driven energy accounting in power-sensitive systems. In *Proceedings of the 9th Workshop on ACM SIGOPS European Workshop: Beyond the PC: New Challenges for the Operating System (EW 9)*. ACM, 37–42. DOI:http://dx.doi.org/10.1145/566726.566736

R. Bertran, Y. Becerra, D. Carrera, V. Beltran, M. Gonzalez Tallada, X. Martorell, J. Torres, and E. Ayguade. 2010. Accurate energy accounting for shared virtualized environments using PMC-based power modeling techniques. In *Proceedings of the IEEE/ACM International Conference on Grid Computing*. 1–8. DOI:http://dx.doi.org/10.1109/GRID.2010.5697889

Aruna Prem Bianzino, Mikael Asplund, Ekhiotz Jon Vergara, and Simin Nadjm-Tehrani. 2014. Cooperative proxies: Optimally trading energy and quality of service in mobile devices. *Comput. Netw.* 75 (2014), 297–312. DOI:http://dx.doi.org/10.1016/j.comnet.2014.10.013

F. Boccardi, R. W. Heath, A. Lozano, T. L. Marzetta, and P. Popovski. 2014. Five disruptive technology directions for 5G. *IEEE Commun. Mag.* 52, 2 (Feb. 2014), 74–80. DOI:http://dx.doi.org/10.1109/MCOM.2014.6736746

Javier Castro, Daniel Gómez, and Juan Tejada. 2009. Polynomial calculation of the shapley value based on sampling. *Comput. Oper. Res.* 36, 5 (May 2009), 1726–1730. DOI:http://dx.doi.org/10.1016/j.cor.2008.04.004

Ranveer Chandra, Omid Fatemieh, Parya Moinzadeh, Chandramohan A. Thekkath, and Yinglian Xie. 2013. *End-to-End Energy Management of Mobile Devices*. Technical Report MSR-TR-2013-69. Microsoft Research.

Yun Cheng, Kaifei Chen, Ben Zhang, Chieh-Jan Mike Liang, Xiaofan Jiang, and Feng Zhao. 2012. Accurate real-time occupant energy-footprinting in commercial buildings. In *Proceedings of the 4th ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings (BuildSys'12)*. ACM, 115–122. DOI:http://dx.doi.org/10.1145/2422531.2422552

Jungwook Cho, Youngjoo Woo, Suntae Kim, and Euiseong Seo. 2014. A battery lifetime guarantee scheme for selective applications in smart mobile devices. *IEEE Trans. Cons. Electron.* 60, 1 (February 2014), 155–163. DOI:http://dx.doi.org/10.1109/TCE.2014.6780938

David Chu, Aman Kansal, Jie Liu, and Feng Zhao. 2011. Mobile apps: It's time to move up to CondOS. In *Proceedings of the 13th USENIX Conference on Hot Topics in Operating Systems (HotOS'13)*. USENIX Association, Berkeley, CA, 16.

Vincent Conitzer and Tuomas Sandholm. 2003. Complexity of determining nonemptiness of the core. In *Proceedings of the 4th ACM Conference on Electronic Commerce (EC'03)*. ACM, 230–231. DOI:http://dx.doi.org/10.1145/779928.779973

Vincent Conitzer and Tuomas Sandholm. 2006. Complexity of constructing solutions in the core based on synergies among coalitions. *Artif. Intell.* 170, 67 (2006), 607–619. DOI:http://dx.doi.org/10.1016/j.artint.2006.01.005

Xiaotie Deng and Christos H. Papadimitriou. 1994. On the complexity of cooperative solution concepts. *Math. Oper. Res.* 19, 2 (May 1994), 257–266, INFORMS. DOI:http://dx.doi.org/10.1287/moor.19.2.257

Ming Dong. 2013. *Energy Accounting and Optimization for Mobile Systems*. Ph.D. Dissertation. Rice University.

Mian Dong, Tian Lan, and Lin Zhong. 2014. Rethink energy accounting with cooperative game theory. In *Proceedings of the 20th Annual International Conference on Mobile Computing and Networking (MobiCom'14)*. ACM, 531–542. DOI:http://dx.doi.org/10.1145/2639108.2639128

Julia Drechsel. 2010. Selected topics in cooperative game theory. In *Cooperative Lot Sizing Games in Supply Chains*. Lecture Notes in Economics and Mathematical Systems, Vol. 644. 5–39, Springer. DOI:http://dx.doi.org/10.1007/978-3-642-13725-9_2

J. Drechsel and A. Kimms. 2010. Computing core allocations in cooperative games with an application to cooperative procurement. *Int. J. Prod. Econ.* 128, 1 (2010), 310–321, Elsevier. DOI:http://dx.doi.org/10.1016/j.ijpe.2010.07.027

C. S. Ellis. 1999. The case for higher-level power management. In *Proceedings of the Seventh Workshop on Hot Topics in Operating Systems, 1999*. IEEE, 162–167. DOI:http://dx.doi.org/10.1109/HOTOS.1999.798394

Randall P. Ellis and Thomas G. McGuire. 1986. Provider behavior under prospective reimbursement. *J. Health Econ.* 5, 2 (1986), 129–151, Elsevier. DOI:http://dx.doi.org/10.1016/0167-6296(86)90002-0

Stefan Engevall, Maud Göthe-Lundgren, and Peter Värbrand. 1998. The traveling salesman game: An application of cost allocation in a gas and oil company. *Ann. Operat. Res.* 82, 0 (1998), 203–218, Springer. DOI:http://dx.doi.org/10.1023/A:1018935324969

Ulrich Faigle, Walter Kern, and Jeroen Kuipers. 1998. Computing the nucleolus of min-cost spanning tree games is NP-hard. *Int. J. Game Theory* 27, 3 (Dec. 1998), 443–450. DOI:http://dx.doi.org/10.1007/s001820050083

Shaheen S. Fatima, Michael Wooldridge, and Nicholas R. Jennings. 2008. A linear approximation method for the shapley value. *Artif. Intell.* 172, 14 (Sept. 2008), 1673–1699. DOI:http://dx.doi.org/10.1016/j.artint.2008.05.003

M. Gloria Fiestras-Janeiro, Ignacio García-Jurado, and Manuel A. Mosquera. 2011. Cooperative games and cost allocation problems. *TOP* 19, 1 (2011), 1–22. DOI:http://dx.doi.org/10.1007/s11750-011-0200-1

Jason Flinn and M. Satyanarayanan. 1999a. Energy-aware adaptation for mobile applications. *SIGOPS Oper. Syst. Rev.* 33, 5 (Dec. 1999), 48–63, ACM. DOI:http://dx.doi.org/10.1145/319344.319155

Jason Flinn and M. Satyanarayanan. 1999b. PowerScope: A tool for profiling the energy usage of mobile applications. In *Proceedings of the 2nd IEEE Workshop on Mobile Computer Systems and Applications (WMCSA'99)*. IEEE.

Jason Flinn and M. Satyanarayanan. 2004. Managing battery lifetime with energy-aware adaptation. *ACM Trans. Comput. Syst.* 22, 2 (May 2004), 137–179. DOI:http://dx.doi.org/10.1145/986533.986534

Rodrigo Fonseca, Prabal Dutta, Philip Levis, and Ion Stoica. 2008. Quanto: Tracking energy in networked embedded systems. In *Proceedings of the 8th USENIX Conference on Operating Systems Design and Implementation (OSDI'08)*. USENIX Association, Berkeley, CA, 323–338.

Debabrata Ghosh and Janat Shah. 2015. Supply chain analysis under green sensitive consumer demand and cost sharing contract. *Int. J. Prod. Econ.* 164 (2015), 319–329, Elsevier. DOI:http://dx.doi.org/10.1016/j.ijpe.2014.11.005

Simon Hay and Andrew Rice. 2009. The case for apportionment. In *Proceedings of the 1st ACM Workshop on Embedded Sensing Systems for Energy-Efficiency in Buildings (BuildSys'09)*. ACM, 13–18. DOI:http://dx.doi.org/10.1145/1810279.1810283

Aman Kansal, Feng Zhao, Jie Liu, Nupur Kothari, and Arka A. Bhattacharya. 2010. Virtual machine power metering and provisioning. In *Proceedings of the 1st ACM Symposium on Cloud Computing (SoCC'10)*. ACM, 39–50. DOI:http://dx.doi.org/10.1145/1807128.1807136

Simon Kellner. 2010. Flexible online energy accounting in TinyOS. In *Proceedings of the 4th International Workshop on Real-World Wireless Sensor Networks (LNCS)*, Vol. 6511. Springer, 62–73. DOI:http://dx.doi.org/10.1007/978-3-642-17520-6_6

Edward A. Lee. 1999. Modeling concurrent real-time processes using discrete events. *Ann. Softw. Eng.* 7, 1–4 (June 1999), 25–45. DOI:http://dx.doi.org/10.1023/A:1018998524196

Jean Lemaire. 1984. An application of game theory: Cost allocation. *ASTIN Bull.* 14, 1 (4 1984), 61–81. DOI:http://dx.doi.org/10.1017/S0515036100004815

Youhuizi Li, Hui Chen, and Weisong Shi. 2014. Power behavior analysis of mobile applications using bugu. *Sust. Comput.: Inform. Syst.* 4, 3 (2014), 183–195. DOI:http://dx.doi.org/10.1016/j.suscom.2014.07.002

M. Maschler, B. Peleg, and L. S. Shapley. 1979. Geometric properties of the kernel, nucleolus, and related solution concepts. *Math. Oper. Res.* 4, 4 (1979), 303–338.

M. Maschler, J. A. M. Potters, and S. H. Tijs. 1992. The general nucleolus and the reduced game property. *Int. J. Game Theory* 21, 1 (1992), 85–106. DOI:http://dx.doi.org/10.1007/BF01240254

Satoshi Masuya and Masahiro Inuiguchi. 2009. Toward the theory of cooperative games under incomplete information. In *Proceedings of the 6th International Conference on Modeling Decisions for Artificial Intelligence (MDAI'09)*. Springer-Verlag, 102–113. DOI:http://dx.doi.org/10.1007/978-3-642-04820-3_10

Roger A. McCain. 2013. *Value Solutions in Cooperative Games*. World Scientific Publishing Company.

Nimrod Megiddo. 1978. Computational complexity of the game theory approach to cost allocation for a tree. *Math. Oper. Res.* 3, 3 (1978), 189–196. DOI:http://dx.doi.org/10.1287/moor.3.3.189

Vishal Misra, Stratis Ioannidis, Augustin Chaintreau, and Laurent Massoulié. 2010. Incentivizing peer-assisted services: A fluid shapley value approach. In *Proceedings of the ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS'10)*. ACM, 215–226. DOI:http://dx.doi.org/10.1145/1811039.1811064

Radhika Mittal, Aman Kansal, and Ranveer Chandra. 2012. Empowering developers to estimate app energy consumption. In *Proceedings of the 18th Annual International Conference on Mobile Computing and Networking (Mobicom'12)*. ACM, 317–328. DOI:http://dx.doi.org/10.1145/2348543.2348583

Elisenda Molina and Juan Tejada. 2000. The least square nucleolus is a general nucleolus. *Int. J. Game Theory* 29, 1 (2000), 139–142. DOI:http://dx.doi.org/10.1007/s001829900028

Stefano Moretti and Fioravante Patrone. 2008. Transversality of the shapley value. *TOP* 16, 1 (2008), 1–41. DOI:http://dx.doi.org/10.1007/s11750-008-0044-5

Hervé Moulin and Scott Shenker. 2001. Strategyproof sharing of submodular costs: Budget balance versus efficiency. *Econ. Theory* 18, 3 (2001), 511–533. DOI:http://dx.doi.org/10.1007/PL00004200

Rolf Neugebauer and Derek McAuley. 2001. Energy is just another resource: Energy accounting and energy pricing in the nemesis OS. In *Proceedings of the 8th Workshop on Hot Topics in Operating Systems (HOTOS'01)*. IEEE.

Adam J. Oliner, Anand P. Iyer, Ion Stoica, Eemil Lagerspetz, and Sasu Tarkoma. 2013. Carat: Collaborative energy diagnosis for mobile devices. In *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems (SenSys'13)*. ACM, Article 10, 14 pages. DOI:http://dx.doi.org/10.1145/2517351.2517354

A. E. Papathanasiou and M. L. Scott. 2003. Energy efficiency through burstiness. In *Proceedings of the 5th IEEE Workshop on Mobile Computing Systems and Applications*. 44–53. DOI:http://dx.doi.org/10.1109/MCSA.2003.1240766

Abhinav Pathak, Y. Charlie Hu, and Ming Zhang. 2012. Where is the energy spent inside my app? Fine grained energy accounting on smartphones with Eprof. In *Proceedings of the 7th ACM European Conference on Computer Systems (EuroSys'12)*. ACM, 29–42. DOI:http://dx.doi.org/10.1145/2168836.2168841

J. E. Petersen, V. Shunturov, K. Janda, G. Platt, and K. Weinberger. 2007. Dormitory residents reduce electricity consumption when exposed to real-time visual feedback and incentives. *Int. J. Sust. Higher Educ.* 8 (2007), 16–33.

Lech Polkowski and Boleslaw Araszkiewicz. 2003. A rough set approach to estimating the game value and the shapley value from data. *Electron. Notes Theor. Comput. Sci.* 82, 4 (2003), 219–227. DOI:http://dx.doi.org/10.1016/S1571-0661(04)80720-5

Justo Puerto and Federico Perea. 2013. Finding the nucleolus of any n-person cooperative game by a single linear program. *Comput. Oper. Res.* 40, 10 (2013), 2308–2313. DOI:http://dx.doi.org/10.1016/j.cor.2013.03.011

Andrew J. Pyles, Xin Qi, Gang Zhou, Matthew Keally, and Xue Liu. 2012. SAPSM: Smart adaptive 802.11 PSM for smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing (UbiComp'12)*. ACM, 11–20. DOI:http://dx.doi.org/10.1145/2370216.2370219

Feng Qian, Zhaoguang Wang, Alexandre Gerber, Zhuoqing Mao, Subhabrata Sen, and Oliver Spatscheck. 2011. Profiling resource usage for mobile applications: A cross-layer approach. In *Proceedings of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys'11)*. ACM, 321–334. DOI:http://dx.doi.org/10.1145/1999995.2000026

A. Reinhardt, P. Baumann, D. Burgstahler, M. Hollick, H. Chonov, M. Werner, and R. Steinmetz. 2012. On the accuracy of appliance identification based on distributed load metering data. In *Sustainable Internet and ICT for Sustainability (SustainIT), 2012*. IEEE, 1–9.

Andrew Rice and Simon Hay. Dec. 2010. Measuring mobile phone energy consumption for 802.11 wireless networking. *Perv. Mobile Comput.* 6, 6 (Dec. 2010), 593–606. DOI:http://dx.doi.org/10.1016/j.pmcj.2010.07.005

Artus Ph. Rosenbusch. 2011. *Fairness Considerations in Cooperative Games*. Ph.D. Dissertation. TU Darmstadt/Mathematik.

Arjun Roy, Stephen M. Rumble, Ryan Stutsman, Philip Levis, David Mazières, and Nickolai Zeldovich. 2011. Energy management in mobile devices with the cinder operating system. In *Proceedings of the 6th Conference on Computer Systems (EuroSys'11)*. ACM, 139–152. DOI:http://dx.doi.org/10.1145/1966445.1966459

Luis M. Ruiz, F. Valenciano, and Jose M. Zarzuelo. 1996. The least square prenucleolus and the least square nucleolus. Two values for TU games based on the excess vector. *Int. J. Game Theory* 25 (1996), 113–134.

Stephen M. Rumble, Ryan Stutsman, Philip Levis, David Mazières, and Nickolai Zeldovich. 2009. Apprehending Joule Thieves with cinder. In *Proceedings of the 1st ACM Workshop on Networking, Systems, and Applications for Mobile Handhelds (MobiHeld'09)*. ACM, 49–54. DOI:http://dx.doi.org/10.1145/1592606.1592618

Sebastian Ryffel. 2009. *LEA^{2}P: The Linux Energy Attribution and Accounting Platform*. Master's thesis. Swiss Federal Institute of Technology.

Sebi Ryffel, Thanos Stathopoulos, Dustin McIntire, William Kaiser, and Lothar Thiele. 2009. *Accurate Energy Attribution and Accounting for Multi-core Systems*. Technical Report. Center for Embedded Network Sensing, University of California. Retrieved from https://escholarship.org/uc/item/81s2s0t2.

Manaswi Saha, Shailja Thakur, Amarjeet Singh, and Yuvraj Agarwal. 2014. EnergyLens: Combining smartphones with electricity meter for accurate activity detection and user annotation. In *Proceedings of the 5th International Conference on Future Energy Systems (e-Energy'14)*. ACM, 289–300. DOI: http://dx.doi.org/10.1145/2602044.2602058

David Schmeidler. 1969. The nucleolus of a characteristic function game. *SIAM J. Appl. Math.* 17, 6 (1969), 1163–1170.

Matteo Sereno. 2012. Cooperative game theory framework for energy efficient policies in wireless networks. In *Proceedings of the 3rd International Conference on Future Energy Systems: Where Energy, Computing and Communication Meet (e-Energy'12)*. ACM, Article 17, 9 pages. DOI: http://dx.doi.org/10.1145/2208828.2208845

L. S. Shapley. 1953. A value for n-person games. *Contrib. Theory Games* 2 (1953), 307–317.

Lloyd S. Shapley. 1971. Cores of convex games. *Int. J. Game Theory* 1, 1 (1971), 11–26. DOI: http://dx.doi.org/10.1007/BF01753431

André Sieber and Jörg Nolte. 2013. Online device-level energy accounting for wireless sensor nodes. In *Proceedings of the 10th European Conference on Wireless Sensor Networks (EWSN'13)*. Springer-Verlag, 149–164. DOI: http://dx.doi.org/10.1007/978-3-642-36672-7_10

Shailja Thakur, Manaswi Saha, Amarjeet Singh, and Yuvraj Agarwal. 2014. WattShare: Detailed energy apportionment in shared living spaces within commercial buildings. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings (BuildSys'14)*. ACM, 30–39. DOI: http://dx.doi.org/10.1145/2674061.2674069

Yi-ting Tsao, Chiao-Ching Huang, and Jane Yung-jen Hsu. 2014. Demand-based apportionment on electricity payment of HVAC systems. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems (AAMAS'14)*. International Foundation for Autonomous Agents and Multiagent Systems, 1601–1602, ACM. Retrieved from http://dl.acm.org/citation.cfm?id=2615731.2616082

Narseo Vallina-Rodriguez and Jon Crowcroft. 2011. ErdOS: Achieving energy savings in mobile OS. In *Proceedings of the 6th International Workshop on MobiArch (MobiArch'11)*. ACM, 37–42. DOI: http://dx.doi.org/10.1145/1999916.1999926

Ekhiotz Jon Vergara, Simon Andersson, and Simin Nadjm-Tehrani. 2014a. When mice consume like elephants: Instant messaging applications. In *Proceedings of the 5th International Conference on Future Energy Systems (e-Energy'14)*. ACM, 97–107. DOI: http://dx.doi.org/10.1145/2602044.2602054

Ekhiotz Jon Vergara, Simin Nadjm-Tehrani, and Mihails Prihodko. 2014b. EnergyBox: Disclosing the wireless transmission energy cost for mobile devices. *Sust. Comput.: Inform. Syst.* 4, 2 (2014), 118–135. DOI: http://dx.doi.org/10.1016/j.suscom.2014.03.008

Heng Zeng, Carla S. Ellis, Alvin R. Lebeck, and Amin Vahdat. 2002a. ECOSystem: Managing energy as a first class operating system resource. *SIGOPS Oper. Syst. Rev.* 36, 5 (Oct. 2002), 123–132, ACM. DOI: http://dx.doi.org/10.1145/635508.605411

Heng Zeng, Carla S. Ellis, Alvin R. Lebeck, and Amin Vahdat. 2002b. ECOSystem: Managing energy as a first class operating system resource. *SIGARCH Comput. Archit. News* 30, 5 (Oct. 2002), 123–132, ACM. DOI: http://dx.doi.org/10.1145/635506.605411

Heng Zeng, Carla S. Ellis, Alvin R. Lebeck, and Amin Vahdat. 2003. Currentcy: A unifying abstraction for expressing energy management policies. In *Proceedings of the Annual Conference on USENIX Annual Technical Conference (ATEC'03)*. USENIX Association, 4–4.

Lide Zhang, Birjodh Tiwana, Zhiyun Qian, Zhaoguang Wang, Robert P. Dick, Zhuoqing Morley Mao, and Lei Yang. 2010. Accurate online power estimation and automatic battery behavior based power model generation for smartphones. In *Proceedings of the 8th IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis (CODES/ISSS'10)*. ACM, 105–114. DOI: http://dx.doi.org/10.1145/1878961.1878982

Lin Zhong and Niraj K. Jha. 2003. Graphical user interface energy characterization for handheld computers. In *Proceedings of the 2003 International Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES'03)*. ACM, 232–242. DOI: http://dx.doi.org/10.1145/951710.951742

Received September 2015; revised June 2016; accepted July 2016