Noname manuscript No.
(will be inserted by the editor)

Easy Knapsacks and the Complexity of Energy Allocation Problems in the Smart Grid

Theo Karaboghossian · Michele Zito

Received: date / Accepted: date

Abstract Motivated by the growing interest in the smart grid and intelligent energy management mechanisms we study two classes of domestic energy allocation problems. In the first case we work with a system that is tasked with scheduling the work on a number of appliances over a given time window. In the second one a collection of air conditioning appliances is used to control the temperature of a given domestic environment. Our framework for this case includes a simplified mechanism for modelling the heat exchange between the interior and the exterior of the given environment. We present various polynomial time algorithms and NP-hardness proofs. In particular the main result of the paper is a proof that although it is NP-hard to schedule the operation of a single air-conditioning (AC) unit, working at various temperature levels in a variable energy price regime, there is a polynomial time algorithm for controlling one such device working at a single temperature level, for houses with low thermal inertia. The algorithm analysis hinges on the properties of a polynomial time variant of the minimisation version of the knapsack problem which may be of independent interest.

Keywords Complexity · Optimization · Dynamic Programming · Algorithms

1 Introduction

The last ten years have seen the rapid development of the concept of smart grid, and significant work in the area of demand side load management [1,2]. With the increasingly large range of (renewable) energy sources and tariffs, optimization in energy management is more and more important. Koutsopoulos and Tassiulas [16] studied a type of optimization problem that is typical in this setting: an energy grid operator receives consumer requests with different power requirements, durations, and deadlines. The objective of the operator is to

F. Author
first address
Tel.: +123-45-678910
Fax: +123-45-678910
E-mail: fauthor@example.com

S. Author
second address

devise a scheduling policy that minimizes the grid operational cost, seen as a convex function of the total load, over a given time horizon. The authors showed that if power demands can be served preemptively then the allocation problem can be solved effectively, and if that is not the case then the problem is NP-hard. Arikiez *et al.* [5] studied a *Micro-Grid* scenario that generalizes Koutsopoulos *et al*'s model. A set of houses and a set of (renewable) energy generators is given, fully connected to each other and connected to a national electricity generator (NEG). Each house has a set of appliances which must be run a certain number of times in specified time periods (*must-use* constraints). Additionally some appliances may be used to control the temperature in (part of) the house (*comfort* constraints). The time horizon is finite and subdivided in constant length time intervals. The collection of such intervals is identified with an initial segment $\mathcal{T} = \{1, 2, \ldots, \tau\}$ of the set of positive natural numbers ("0" often refers to a generic moment in time before the process of interest starts). The system state is only allowed to change, instantaneously, between successive time intervals. Hence we talk equivalently of "time (steps)" or "time intervals". Some of the system settings are described by finite sequences defined over $\mathcal{T}$. For instance each generator r has an *available energy* function: $P_r(t)$ indicates how much electricity this generator can deliver during the t-th time (interval) and we assume that this stays constant during such interval. Also, a cost function $\gamma_{r,h}(t)$ is defined for each house-generator pair (h, r) which indicates the unit price at which house h can buy electricity from generator r at time t. The NEG is represented by a cost function $\lambda_h(t)$ which indicates the unit price at which any house can buy electricity from NEG at time t. To avoid trivialities we assume that $\gamma_{r,h}(t) < \lambda_h(t)$ for all r, h, and t.

The main purpose of this paper is to study the complexity of classes of optimization problems which can be defined within Arikiez's et al. framework. For this reason, in the rest of the paper, we restrict to systems formed by a single house, connected to NEG and to a single generator (hence we drop indices h and r from our notations). In this setting P1 (equivalent to the non-preemptive case studied by Koutsopoulos and Tassiulas) is the problem of finding an appliance schedule compatible with a given set of must-use constraints which minimizes the energy cost for the house, assuming no temperature controlling appliance is present. After describing a few polynomial time cases, we show that P1 is actually NP-hard in the strong sense even if a certain amount of renewable energy is available free of charge. The main contribution of this paper, though, is the first theoretical analysis of a second, much richer, variant which we call P2. Here there is no limit on how many times each appliance is run, but the house has internal temperature constraints and appliances are air-conditioning (AC) units used to keep the internal temperature within such constraints. In the forthcoming sections we investigate the effect of energy cost variability as well as the type and distribution of the AC units on the problem complexity. If there are many units the problem is NP-hard, in the variable cost scenario. Therefore the most interesting case is that of an apartment with a single unit. If the device can operate at many different temperature states the problem is NP-hard. However if the AC unit only has a single operating mode then the problem can be solved efficiently. This is simple if the energy cost is fixed, however, if the cost varies, we found that the existence of an efficient algorithm solution depends crucially on the house thermal inertia. Our main algorithmic result on the problem at hand involves the study of the properties of a minimization version of KNAPSACK, which may be of independent interest.

Variants of the allocation problem considered here have been studied before [11, 14, 17, 7]. In fact research in domestic energy management, technologies for the smart grid, and home automation is thriving with hundreds of papers published every year and dozens of conferences and journals devoted to these topics. However it seems that most of the effort concentrated on finding feasible allocation heuristics, while relatively little [3] attention has been given to the scalability of such heuristics or the problems complexity. Some of these problems are related to machine scheduling [6] and bin packing [16]. But the way in which the appliances are used, their arrangement and different price strategies are specific features of the smart grid setting. Problem P2 is related to the minimum cost resource allocation studied in [9] and the capacitated covering problems of [8]. However our hardness proofs hold in simpler cases than those considered in the cited papers, and nobody seems to have considered exact polynomial time algorithms for non-trivial special cases.

In the next Section we focus on P1. We define the problem and discuss its complexity. In Section 3 we work on P2. We start by providing all relevant definitions. Then we analyze the problem's complexity first in the case of many appliances (Section 3.1) then looking at single room single appliance systems (Section 3.2 and 3.3). The main result of this paper is the design of a polynomial time strategy for the optimization problem P2 in the case of a single AC unit, with a single operating state, in a poorly isolated house. In all our complexity results if Π_1 and Π_2 are two computational problems then $\Pi_1 \leq \Pi_2$ will stand for the statement "Π_1 is polynomial time reducible to Π_2" in the sense that there is a polynomial time algorithm translating instances of Π_1 into instances of Π_2 that preserve solvability (the reader is referred to [13] for all basic complexity theoretic definitions and notations).

2 Allocating "Must Use" Appliances

In this section we focus on problem P1. The given house contains n appliances, identified by the integers $1, \ldots, n$. The model presented in [5] is quite general but to simplify our presentation we restrict ourselves mainly to so called *uniphase interruptible* appliances: at each time step t each unit i is either "OFF" or it is "ON" and if it is "ON" it uses an amount of power equal to α_i. Each appliance's state can be changed freely at any time. Given τ consecutive time steps the goal is to run each appliance exactly once for a single time step, minimizing the total energy cost for the house. The problem admits a natural Linear Program formulation. The total amount of electricity needed at time t is $\sum_{1 \leq i \leq n} \alpha_i \cdot x_i(t)$ where, for each $i \in \{1, \ldots, n\}$, $x_i(t) = 1$ (resp. $x_i(t) = 0$) if appliance i is ON (resp. OFF) at time t. Electricity may come either from the NEG or from the local renewable power generator. Let $G(t)$ (resp. $L(t)$) denote the amount of power taken from the generator (resp. from the NEG) at time t. Problem P1 is then described as follows (bold typeface symbols denote vectors with τ components, so, for instance, $\mathbf{P} = (P(1), \ldots, P(\tau))$, etc and notations like "$\mathbf{x} \cdot \mathbf{y}$" have the usual algebraic interpretation):

$$\begin{array}{rll} \min & \boldsymbol{\lambda}^{\mathrm{T}} \cdot \mathbf{L} + \boldsymbol{\gamma}^{\mathrm{T}} \cdot \mathbf{G} & \text{s.t.} \\ \mathbf{1}^{\mathrm{T}} \cdot \mathbf{x}_i = 1 & & \forall i \in \{1, \ldots, n\} \\ \mathbf{G} \leq \mathbf{P} & & \\ \mathbf{L} + \mathbf{G} = \sum_{1 \leq i \leq n} \alpha_i \cdot \mathbf{x}_i & & \\ \mathbf{x}_i \in \{0, 1\}^{(\tau)} & & \forall i \in \{1, \ldots, n\} \end{array}$$

where the vector $\mathbf{x}_i$ describes the state of appliance i at each time step, the first constraint forces $x_i(t)$ to be one at a single point t, the second one forces the required amount of renewable power not to be larger than the total renewable power available, the third constraint is an energy balance one, and the last one restricts the range of the vectors $\mathbf{x}_i$.

Note that if the number of appliances is fixed the set of feasible solutions for instances of P1 can be enumerated in time polynomial in τ. In fact this is true also if we required each

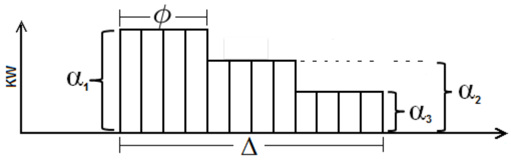

Fig. 1 Multiphase appliance, with three phases each running for four time steps, using power α_1, α_2, and α_3 respectively.

appliance to be used an arbitrary, but fixed, number of times, or if the appliances had any of the more complex energy usage patterns defined in [5] (Figure 1 gives an example of the energy requirements for a *multiphase non-interruptible* appliance.) Also, if the system has no renewable power generator then, again, the allocation is easy as we can simply allocate everything at a time for which $\lambda(t)$ is minimal. Conversely, if the number of appliances is large the problem is NP-hard, even for $\tau = 2$ [3,5]. Here we strengthen such result. Arikiez *et al.* [5] evaluate the performances of an exact algorithm for P1 and observe that they degrade rapidly if n or τ become large. However it is not clear whether a so called pseudo-polynomial [13, Chapter 4] algorithm may exist that allow P1 to be solved in time polynomial in the magnitude of the numbers involved in the problem instance (this has proved to be beneficial in practical contexts [4]). Our next result makes this rather unlikely.

The 3-PARTITION problem is defined as follows (see [13]):

DATA: $a_1, \ldots, a_{3m}$ positive integers adding up to mB such that each a_i satisfies $B/4 < a_i < B/2$.

SOLUTION: A partition of the given set of numbers into m blocks such that the sum of the elements in each block is equal to B.

It is well-known that 3-PARTITION is NP-hard in the strong sense (see [13, p.99]) and this in turns rules out the possibility of a pseudo-polynomial time algorithm for this problem (unless P=NP). We describe a reduction from 3-PARTITION to the decision version of P1 that preserves strong NP-hardness. This leads to the following result.

Theorem 1 *Problem* P1 *is NP-hard in the strong sense.*

Proof We show that a generic instance of 3-PARTITION can be reduced to the decision version of P1 with a pseudo polynomial transformation (as defined in [12]). Let $a_1, \ldots, a_{3m}$ and $B > 0$ be an instance of 3-PARTITION. Define an instance of P1 by taking $\tau = m$, and using $3m$ appliances with $\alpha_i = a_i$ for each $i \in \{1, \ldots, 3m\}$. Assume that there is a single renewable power generator. Let $\gamma(t) = 0$, and $P(t) = B$, for all $t \in \mathcal{T}$. Set $\lambda(t)$ to be some arbitrary fixed positive value.

The transformation preserves strong NP-hardness since the largest numerical value of the resulting instance is $B < 4 \cdot a_1$. It is easy to see that if there is partition in m blocks then there is an appliance allocation over $\tau (= m)$ time steps that uses all renewable power available and costs nothing. Conversely, if there is no good partition then there must be a time step t in which we need more than $P(t)$ energy (and hence we must pay for it, buying it from NEG). □

3 Controlled Temperature Environments

So far we have looked at energy optimisation in a rather isolated environment: a house connected to several energy sources, has a number of appliances that consume energy and need to be scheduled in a given time window. In this section we change this in two ways. First, appliances do not correspond to tasks that must be executed at all costs. Second, the house sits in an environment that exchange heat with the building. In this context appliances should be thought of as air conditioning (AC) units used to control the interior temperature. This second framework is again inspired by the work in [5] but as in Section 2, our goal is to understand the problem features that affect its complexity and for this reason the model is presented in a somewhat simplified fashion.

The given house is split into a set $\mathcal{S}$ of rooms, each having a thermostat for measuring the room temperature. The external environment affects the house in two ways. First, like before, the house can use the renewable energy generated by a single local micro-generation plant. Second, the house sits in an environment whose temperature $\mathbf{T}_{out}$ is known in advance throughout the time window of interest. Each room $s \in \mathcal{S}$ contains $n_s > 0$ AC units. Each unit can either be OFF or making a certain contribution to the temperature of the room it is in. Assume that unit i in room s has a finite set of allowed temperatures contributions $\mathit{\Delta} T_{s,i} = \{T_1^{s,i}, \ \ldots, \ T_{n_{s,i}}^{s,i}\}$. Positive (negative) elements of $\mathit{\Delta} T_{s,i}$ correspond to the appliance being used as a heater (resp. cooler). The goal is to keep each room's temperature in a predefined comfort interval $[T^s_{min}(t), T^s_{max}(t)]$ for all $t \in \mathcal{T}$. Following [15], for each room s, we assume that the room temperature at time t, denoted by $T_s(t)$, is linked to the outside temperature and the room's units behaviour by the equation:

$$\forall t \in \mathcal{T} \qquad T_s(t) = \epsilon \cdot T_s(t-1) + (1-\epsilon)(T_{out}(t) + x_s(t)), \tag{1}$$

where $x_s(t)$ is the average of the contributions of all appliances in the room that are ON at time t and $\epsilon \in [0,1]$ is an *inertia* factor. A discussion on the validity of (1) is beyond the scope of this paper (the interested reader is referred to [15]). However it is instructive to understand how the formula works. The room temperature at time t is viewed as a linear combination of its temperature at time $t-1$ and contributions from external sources, and that of the air conditioning appliances it contains. The coefficients of this combination depend on physical properties of the house, like its volume or the materials it is made of. Note that, if $\epsilon = 1$, the internal environment is perfectly isolated from the outside and, in fact, we also have no way to control the internal temperature. On the other hand, if $\epsilon = 0$, then there is no isolation and the system has no memory. At every moment in time the internal temperature is equal to the external one plus a value dependent on the contribution of the AC units. Therefore, to avoid trivialities, from now on we further assume that $0 < \epsilon < 1$. The problem of interest admits a natural mixed integer linear programming formulation:

$$\begin{array}{rll}
\min & \boldsymbol{\lambda}^{\mathrm{T}} \cdot \mathbf{L} + \boldsymbol{\gamma}^{\mathrm{T}} \cdot \mathbf{G} & \text{s.t.} \\
\mathbf{G} \leq & \mathbf{P} & \\
\mathbf{L} + \mathbf{G} = & \sum_{s \in \mathcal{S}} \sum_{i=1}^{n_s} \frac{1}{\eta_i^s} \sum_{j=1}^{n_{s,i}} |T_j^{s,i}| \mathbf{y}_j^{s,i} & \\
\mathbf{x}_s = & \frac{1}{n_s} \sum_{i=1}^{n_s} \sum_{j=1}^{n_{s,i}} T_j^{s,i} \mathbf{y}_j^{s,i} & \forall s \in \mathcal{S} \\
\sum_{j=1}^{n_{s,i}} \mathbf{y}_j^{s,i} \leq & \mathbf{1} & \forall s \in \mathcal{S}, i \in \{1, \ldots, n_s\} \\
\mathbf{y}_j^{s,i} \in & \{0,1\}^{(\tau)} & \forall s \in \mathcal{S}, i \in \{1, \ldots, n_s\}, j \in \{1, \ldots, n_{s,i}\} \\
\mathbf{E}_\epsilon \cdot \mathbf{x}_s \geq & \mathbf{T}^s_{min} & \forall s \in \mathcal{S} \\
\mathbf{E}_\epsilon \cdot \mathbf{x}_s \leq & \mathbf{T}^s_{max} & \forall s \in \mathcal{S}
\end{array}$$

where $y_j^{s,i}(t)$ is one (resp. zero) if the ith appliance in room s is in state j at time t. Binary variables $y_j^{s,i}(t)$ are use to model the fact that each AC unit can only be in one of its $n_{s,i}$ states at any given time step. Discrete variables $x_s(t)$ model the temperature change in room s at time t. We are after an assignment to these variables that minimize the cost of the house appliances and keeps the temperature of room s in the comfort ranges $[T^s_{min}(t), T^s_{max}(t)]$ for every $t \in \mathcal{T}$ and $s \in \mathcal{S}$. The first four sets of constraints model the energy allocation process. The right-hand side of the second one is a vector describing the total amount of energy used by the house. Here $\eta_i^s > 0$ is the efficiency of unit i in room s, thus if the unit contributes $T_j^{s,i}$ in a particular time step, then $|T_j^{s,i}|/\eta_i^s$ is the amount of power needed by the unit during that step. The last three sets of constraints restrict the appliance choices at any given time to those that keep the room temperatures within the prescribed limits. Notice that the recursive contraints (1) have been solved and the internal temperature variables $\mathbf{T}_s$ replaced by their values $(1-\epsilon)\mathbf{F}_\epsilon[\mathbf{T}_{out} + \mathbf{x}_s] + \mathbf{T}_\epsilon$. Here $\mathbf{E}_\epsilon$ and $\mathbf{F}_\epsilon$ are the following $\tau \times \tau$ matrices:

$$\mathbf{E}_\epsilon = \begin{pmatrix} \epsilon^{\tau-1} & 0 & \dots & \dots & 0 \\ \epsilon^{\tau-1} & \epsilon^{\tau-2} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \epsilon^{\tau-1} & \epsilon^{\tau-2} & \dots & \epsilon & 0 \\ \epsilon^{\tau-1} & \epsilon^{\tau-2} & \dots & \epsilon & 1. \end{pmatrix} \qquad \mathbf{F}_\epsilon = \begin{pmatrix} 1 & 0 & \dots & 0 \\ \epsilon & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \epsilon^{\tau-1} & \dots & \epsilon & 1 \end{pmatrix}.$$

Also, the bounds on the products $\mathbf{E}_\epsilon \cdot \mathbf{x}_s$, which, abusing notations, we still call $\mathbf{T}^s_{min}$ and $\mathbf{T}^s_{max}$, are, in fact, the only part of this model that depends on the initial internal temperature, the comfort limits, and the outside temperature. They stand for, respectively

$$\boldsymbol{\epsilon}^\tau\left(\frac{1}{1-\epsilon}(\mathbf{T}^s_{min} - \mathbf{T}^s_\epsilon) - \mathbf{F}_\epsilon \cdot \mathbf{T}_{out}\right) \quad \text{and} \quad \boldsymbol{\epsilon}^\tau\left(\frac{1}{1-\epsilon}(\mathbf{T}^s_{max} - \mathbf{T}^s_\epsilon) - \mathbf{F}_\epsilon \cdot \mathbf{T}_{out}\right)$$

with

$$(\boldsymbol{\epsilon}^\tau)^{\mathrm{T}} = (\epsilon^{\tau-1}, \epsilon^{\tau-2}, \dots, 1), \tag{2}$$

and $\mathbf{T}_\epsilon^{\mathrm{T}} = T_s(0) \cdot (\epsilon, \epsilon^2, \dots, \epsilon^\tau)$.

3.1 Many Heathers

We first consider the case of a house containing many AC units. First assume that they are all part of a single room. Starting from [10], various authors have defined a natural minimization version of the classical KNAPSACK problem. For future reference denote by MINSACK$(\mathbf{w}, \mathbf{p}, M)$ an instance of such problem involving item weights $\mathbf{w}$, profits $\mathbf{p}$ and knapsack bound M. This is exactly the problem we are interested in here where items correspond to appliances, and the allocation is over a single time step. Next, we will show that many single state heaters make P2 NP-hard even if the heaters are installed in different rooms in a variable energy price regime. In what follows PARTITION is the well-known NP-hard [13, Problem SP12] computational problem defined as follows:

DATA: $a_1, ..., a_n$ positive integers.
SOLUTION: a subset $I \subseteq \{1, \dots, n\}$ such that $\sum_{i \in I} a_i = \sum_{i \in \{1,\dots,n\} \setminus I} a_i$.

Theorem 2 PARTITION $\leq$ P2.

Proof Each instance of PARTITION is translated into an instance of P2 having $\tau = 2$, n rooms, each equipped with a single heater having a single "ON" state. Value a_s becomes the amount of energy needed by the unit in room s to run in a single time step. Thus for each $s \in \{1, \dots, n\}$, $T^s = \eta_s \cdot a_s$ (the house and the rooms thermostatic parameters ϵ and η_s can be chosen arbitrarily). For each $s \in \{1, \dots, n\}$ we set $T^s_{min}(1)$ is an arbitrary negative number and choose $T^s_{min}(2)$ in the interval $(0, \epsilon\, T^s)$, Similarly we choose $T^s_{max}(1) > \epsilon\, T^i$ and $T^s_{max}(2) \in (T^s, (1+\epsilon)\, T^s)$. $P(1) = P(2) = \frac{1}{2} \sum a_s$ and we assume that renewable energy costs nothing, whereas the energy from the grid has an arbitrary price λ.

To complete the proof we need to show that if $a_1, \dots, a_n$ is a "YES" instance of PARTITION then the resulting instance of P2 can be solved with an allocation that has zero cost and if $a_1, \dots, a_n$ is a "NO" instance of PARTITION any solution of P2 will have a positive cost. This will follow from a key property shared by all solutions to the instances of P2 under consideration: each solution requires each of the n heaters to be switched on exactly once. To prove this notice that the temperature constraints simplify to

$$\mathbf{T}^s_{min}/T^i \le \begin{pmatrix} \epsilon & 0 \\ \epsilon & 1 \end{pmatrix} \cdot \mathbf{y}^s \le \mathbf{T}^s_{max}/T^i$$

and the product in the middle, depending on the value $\mathbf{y}$, is equal to $\begin{pmatrix} 0 \\ 0 \end{pmatrix}$, $\begin{pmatrix} \epsilon \\ \epsilon \end{pmatrix}$, $\begin{pmatrix} 0 \\ 1 \end{pmatrix}$, or $\begin{pmatrix} \epsilon \\ 1+\epsilon \end{pmatrix}$. But only the two middle values (corresponding to $\mathbf{y}^s$ having only one of the two components equal to one) satisfy the stated inequalities.

Suppose now that the given instance of PARTITION is a "YES" instance and let I be one of its solutions. Then scheduling all units in I to run at the same time step will result in a feasible solution for P2 with cost zero. Conversely if the given instance of PARTITION is a "NO" instance then any solution of P2 must have a positive cost. □

The upshot of the analysis in this section is that P2 does not seem very easy to solve if the system contains a large number of AC units. In the rest of the paper we will concentrate on increasingly restricted versions of this problem.

3.2 One Heater, A Hard Case

From now on we focus on a further restriction, which we call PS, obtained by assuming that the house only contains one room, that there is always enough renewable power, and that the single AC unit can be used in n different states, all of them providing a positive temperature contribution. We start our analysis by showing that PS is still NP-hard if the electricity price varies and n is large. The SUBSETSUM problem [13, Problem SP13] is defined as follows:

DATA: $a_1, ..., a_n$ and M positive integers.
SOLUTION: a subset $I \subseteq \{1, \dots, n\}$ such that $\sum_{i \in I} a_i = M$.

Theorem 3 SUBSETSUM $\le$ PS.

Here PS denotes the decision version of the energy allocation problem at hand. The reduction translates each instance of SUBSETSUM to an instance of PS involving a single heater with many different (positive) temperature contributions. Let $a_1, a_2, \dots, a_n$ and M define an instance of SUBSET SUM. We set $\tau = n$. Furthermore set

- $\epsilon = \frac{1}{2} \min \left(\frac{1}{\tau \max\{a_i\}}, \frac{1}{M} \right)$;

- the energy prices $\gamma(t) = \epsilon^{\tau - t}$ for all $t \in \mathcal{T}$,
- the heater temperature contributions $T_j = a_j/\epsilon^{\tau-j}$ for all $j \in \{1, \ldots, n\}$, and $\eta = 1$, and
- finally set $T_{min}(t) = 0$ for all $t \in \mathcal{T} \setminus \{\tau\}$ and $T_{min}(\tau) = M$.

We argue that the SUBSETSUM instance is a "YES" instance if and only if the instance of PS admits a solution of cost M. First notice that with these choices we can easily see that the first $\tau - 1$ temperature constraints are always verified since expressions

$$\sum_{1 \le i \le t} \sum_{1 \le j \le n} T_j \cdot y_j(i) \cdot \epsilon^{\tau - i} \qquad t \in \mathcal{T} \setminus \{\tau\}$$

are non-negative. Denote by Ψ the problem's objective function. The last temperature constraint actually implies that a "YES" instance of PS must have $\Psi = M$. The next lemma states a property that can be used to complete the proof of Theorem 3.

Lemma 1 *If $\mathbf{y}$ is a solution of our reduced problem then for all $t \in \mathcal{T}$ we have $x(t) = T_t$ or $x(t) = 0$.*

Proof We will show that for all $t \in \mathcal{T}$ we have $x(t) \notin \{T_1, \ldots, T_{t-1}\}$ and then that $x(t) \notin \{T_{t+1}, \ldots, T_\tau\}$. The first step just comes from the fact that if we have a t such as $x(t) = T_l$ with $l < t$ we would have $\epsilon^{\tau-t} \cdot x(t) = \epsilon^{\tau-t} \cdot T_l = \epsilon^{\tau-t} \cdot a_l/\epsilon^{\tau-l} = a_l/\epsilon^{t-l} \ge a_l/\epsilon$ and since $\epsilon < 1/M$ this value is greater than M. So the objective function would also be greater that M. Now to the second step. Suppose we have found a solution such that $\Psi = M$ and that for some $t \in \{1, \ldots, n\}$ we have $x(t) = T_l$ with $l > t$. Let A be the set of these t, we decompose Ψ in two sums: $\sum_{t \notin A} \epsilon^{\tau-t} \cdot x(t) + \sum_{t \in A} \epsilon^{\tau-t} \cdot x(t) = M$. Note that since we already proved that for all $t \in \mathcal{T}$ we have $x(t) \notin \{T_1, \ldots, T_{t-1}\}$ what we sum in the first sum are either 0s or a_ts and the first sum is an integer. Let's take a closer look at the second sum: for all $t \in A$ we have a l_t such that $x(t) = T_{l_t}$ with $l_t > t$ by definition, so we have:

$$\begin{aligned}
\sum_{t \in A} \epsilon^{\tau-t} \cdot x(t) &= \sum_{t \in A} \epsilon^{\tau-t} \cdot \frac{a_{l_t}}{\epsilon^{\tau-l_t}} && \text{by the choice of } x(t) \\
&\le \sum_{t \in A} \epsilon \cdot a_{l_t} && \text{since } 0 < \epsilon < 1 \text{ and } l_t, t \text{ are integers with } l_t > t \\
&\le |A| \cdot \epsilon \cdot \max\{a_j\} && \text{by defition of max} \\
&\le \tau \cdot \epsilon \cdot \max\{a_j\} && \text{by defition of } A
\end{aligned}$$

hence, since $\epsilon < 1/\tau \max\{a_j\}$, this sum is less than 1, thus it is not an integer, and this in turns contradicts the fact that M and the first sum are integers. □

3.3 Polynomial Time Algorithms

So far we have discovered that instances of PS involving variable energy costs *and* heaters with many temperature levels may be hard to solve. We complete this section by describing a variant of PS that can be solved in polynomial time. Consider the following problem:

$$\begin{aligned}
\min \quad & \gamma^{\mathrm{T}} \cdot \mathbf{y} \quad \text{s.t.} \\
\mathbf{y} &\in \{0,1\}^{(\tau)} \\
\mathbf{E}_\epsilon \cdot \mathbf{y} &\ge \mathbf{T}_{min}
\end{aligned}$$

Algorithm Greedy

```
Require: $\mathbf{T}_{min}$, $\epsilon$
 1: $\mathbf{y} = \mathbf{0}$
 2: for $t := 1$ to $\tau$ do
 3:     $i := 0$
 4:     while $T_{min}(t) > \sum_{1 \le x \le t} y(x) \cdot \epsilon^{\tau - x}$ and $i < t$ do
 5:         $y(t - i) := 1$
 6:         increment $i$
 7:     end while
 8:     if $i = t$ and $T_{min}(t) > \sum_{1 \le x \le t} y(x) \cdot \epsilon^{\tau - x}$ then exit /* ”no solution” */
 9:     end if
10: end for
11: return $\mathbf{y}$
```

If the electricity price is fixed (ie wlog $\boldsymbol{\gamma} = \mathbf{1}$), there is a single heater which can either be “OFF” or in a single “ON” state contributing some value T to the house temperature, then we claim that Algorithm Greedy can be used to find an optimal solution for the problem above in polynomial time. The idea is to solve the τ constraints $\mathbf{E}_\epsilon \cdot \mathbf{y} \geq \mathbf{T}_{min}$ iteratively, one by one. The algorithm **while** loop has the task of trying to build a vector $\mathbf{y}$ that satisfy the temperature constraints. It is easy to argue (formally by induction on t) that after the t-th iteration of the main **for** loop, if we did not enter the **if** on line 8, then $y(1), \ldots, y(t)$ is a minimal solution of the problem. Note that the left-hand side of the t-th inequality, $\sum_{1 \le x \le t} y(x) \cdot \epsilon^{\tau - x}$, is just $y(t) \cdot \epsilon^{\tau - t}$ plus the left-hand side of the $t-1$-th inequality. Hence if the process reached stage t either $\sum_{1 \le x \le t-1} y(x) \cdot \epsilon^{\tau - x} \geq T_{min}(t)$, in which case the same assignment will be picked up in the **while** loop in stage t or $\sum_{1 \le x \le t-1} y(x) \cdot \epsilon^{\tau - x} < T_{min}(t)$, in which case a different assignment will be computed. The new assignment will satisfy all previous constraints because of the order in which variables $y(x)$ are set (starting from the ones multiplying the largest monomials $\epsilon^{\tau - x}$).

The analysis so far leaves the variable energy cost case open. While we are not able to answer this in full in the reminder of this section we present a polynomial time strategy to solve the problem above provided ϵ is a positive constant smaller than $1/2$. In what follows we denote this problem as PS($\frac{1}{2}$). The result hinges on a particular property of the sequence $\boldsymbol{\epsilon}^\tau$, as defined in (2), and on the computational feasibility of a class of KNAPSACK instances involving sequences of this type. The main result of this section is the following:

Theorem 4 PS($\frac{1}{2}$) *can be solved in polynomial time.*

A sequence of non-negative real numbers $\mathbf{w} = (w(1), \ldots, w(a))$ is *left independent* if for all positive integers $j \leq a$ we have $\sum_{i=1}^{j-1} w(i) < w(j)$. Note that if $\mathbf{w}$ is left independent, then so is any of its prefixes $\mathbf{w}_i = (w(1), \ldots, w(i))$, for $i < a$. Also, it is easy to see that the sequence $\boldsymbol{\epsilon}^\tau$ is left independent, provided $\epsilon \in (0, 1/2)$. Left independent sequences also have another important property.

Lemma 2 MINSACK$(\mathbf{w}, \mathbf{p}, M)$ *can be solved in polynomial time, for any profit vector* $\mathbf{p}$ *and bound* M*, provided* $\mathbf{w}$ *is left independent.*

Proof Algorithm EKP solves MINSACK$(\mathbf{w}, \mathbf{p}, M)$ in polynomial time if $\mathbf{w}$ is left independent. The process returns the empty set if $M \leq 0$, otherwise it searches for the greatest

Algorithm EKP

```
Require: $\mathbf{w}$, $\mathbf{p}$, and $M$
i := a
while $i > 0$ and $w(i) \geq M$ do decrement $i$
end while
if $i = 0$ then
    if $M > 0$ then return $\mathrm{argmin}_{1 \leq j \leq a}(p(j))$
    else return $\emptyset$
    end if
else
    $S$ := EKP($\mathbf{w}_{i-1}, \mathbf{p}_{i-1}, M - w(i)$)
    if $\min_{i<j\leq a}(p(j)) < \sum_{j \in S} p(j) + p(i)$ then return $\mathrm{argmin}_{i<j\leq a}(p(j))$
    else return $S \cup \{i\}$
    end if
end if
```

$w(i)$ for $i \leq a$ lower than M and run recursively on i and $M - w(i)$. It then compares $\sum_{j \in S} p(j) + p(i)$ with the lowest $p(j)$ with $j > i$ and return $S \cup \{i\}$ if the former was lower and $\{j\}$ otherwise.

We prove that EKP works as promised by induction on a. The case $a = 1$ is simple: if $w(1) \geq M > 0$ we return $\{1\}$ and if $M \leq 0$ we return $\emptyset$. Suppose now the algorithm works for 1, 2, ..., $a - 1$. The first case of the main **if** doesn't really need any comment: if $M \leq 0$ then $\emptyset$ is the best solution and else we choose the j minimizing $p(j)$ since all the $w(j)$ work. To show that the algorithm works in the second case we just have to prove that $S \cup \{i\}$ is the smallest solution without a $j > i$. It's quite simple to show: since we have $\sum_{1 \leq j \leq i-1} w(j) < w(i) < M$ if we don't take a $j > i$ then we are forced to take i and so we just have to find the minimal solution for $\textsc{Minsack}(\mathbf{w}_{i-1}, \mathbf{p}_{i-1}, M - w(i))$ which is exactly S by induction hypothesis. □

In what follows if $S \subset \mathcal{T}$ let $\boldsymbol{\chi}_S \in \{0,1\}^\tau$ be the vector of size τ such that $\chi_S(t) = 1 \iff t \in S$. Also, $\min S$ (resp. $\max S$) is the smallest (largest) element of S and if $t_1 \leq t_2$, then $[t_1..t_2]$ denotes the set $\{t_1, t_1 + 1, \ldots, t_2\}$. Finally for any $S \subset \mathcal{T}$ let $v(S) = \sum_{t \in S} \epsilon^{\tau - t}$ and for any vector $\mathbf{p}$ of size τ denote $\mathbf{p}[S]$ the expression $\sum_{t \in S} p(t)$.

Lemma 3 *Let $\mathbf{p}$ be a vector of size τ and S_1, $S_2 \subset \mathcal{T}$.*

1. *If S_1, S_2 are disjoint then $v(S_1 \cup S_2) = v(S_1) + v(S_2)$ and $\mathbf{p}[S_1 \cup S_2] = \mathbf{p}[S_1] + \mathbf{p}[S_2]$.*
2. *The function v is injective.*
3. *$v(S_1) < v(S_2)$ if and only if there exist $t \in \mathcal{T} \setminus S_1$ and $S \subset [1..t-1]$ such that $S_2 = (S_1 \setminus [1..t]) \cup \{t\} \cup S$.*

Proof The first claim is trivial. Suppose $S_1 \neq S_2$, we will show that $v(S_1) \neq v(S_2)$. Let $t = \max S_1 \Delta S_2$, where $S_1 \Delta S_2$ is the symmetric difference of S_1 and S_2. Without loss of generality we may assume that $t \in S_2$. Then by the left independence of $\boldsymbol{\epsilon}^\tau$ we have $v(S_1 \cap [1..t]) < \epsilon^{\tau - t} \leq v(S_2 \cap [1..t])$. This in turns implies that $v(S_1 \cap [1..t]) + v(S_1 \setminus [1..t]) < v(S_2 \cap [1..t]) + v(S_2 \setminus [1..t])$, also using $S_1 \setminus [1..t] = S_2 \setminus [1..t]$. The result now follows by the first part of the statement as $v(S_1) < v(S_2)$.

We now argue about the final claim. If there exists a t as stated such that $S_2 = (S_1 \setminus [1..t]) \cup \{t\} \cup S$ then by left independence we have $v(S_1 \cap [1..t]) < \epsilon^{\tau - t}$ since $t \notin S_1$.

Therefore:

$$\begin{aligned} v(S_1) &= v(S_1 \cap [1..t]) + v(S_1 \setminus [1..t]) \quad \text{by the first statement} \\ &< \epsilon^{\tau-t} + v(S_1 \setminus [1..t]) \\ &\leq v(S) + \epsilon^{\tau-t} + v(S_1 \setminus [1..t]) \quad \text{for any } S \subset [1..t-1] \\ &= v(S_2) \quad \text{again by the first statement} \end{aligned}$$

Conversely, let $t = \max S_1 \Delta S_2$. Then it must be $t \in S_2$ since otherwise by definition of the symmetric difference we would have $S_1 \setminus [1..t] = S_2 \setminus [1..t]$ and so, by the same argument used to prove the other implication, we would have

$$\begin{aligned} v(S_2) &= v(S_2 \cap [1..t]) + v(S_2 \setminus [1..t]) = v(S_2 \cap [1..t-1]) + v(S_2 \setminus [1..t]) \\ &< \epsilon^{\tau-t} + v(S_2 \setminus [1..t]) = \epsilon^{\tau-t} + v(S_1 \setminus [1..t]) \\ &< v(S_1) \end{aligned}$$

which is not possible by hypothesis. Hence $S_2 = (S_1 \setminus [1..t]) \cup \{t\} \cup S$ with $S = S_2 \cap [1..t-1] \subset [1..t-1]$ □

If M is a positive real number, we call *ϵ-decomposition of M* the subset $S_\epsilon(M)$ of $\mathcal{T}$ such that:

$$v(S_\epsilon(M)) = \min\{v(I) \mid I \subset \mathcal{T} \text{ and } v(I) \geq M\}.$$

Thus $v(S_\epsilon(M))$ is the smallest number greater than M which can be written as a sum of powers of ϵ^ℓ for $\ell \in \mathcal{T}$. Note that $S_\epsilon(M)$ can be computed as $\textsc{Minsack}(\boldsymbol{\epsilon}^\tau, \boldsymbol{\epsilon}^\tau, M)$.

We are now ready to complete the proof of Theorem 4. We claim that algorithm Main takes $\mathbf{T}_{min}$, the cost function γ and ϵ as arguments and returns an optimal solution to the given instance of PS($\frac{1}{2}$) in time polynomial in τ. The process starts by creating the ϵ-decompositions of the numbers $T_{min}(t)$ and stores them in the array Sol (at the same time checking whether the given problem is trivially unfeasible). The second loop is the main part of the algorithm. In the tth iteration we focus on $\mathsf{Sol}[t]$ (assuming it is not empty). Thanks to the left independence of the rows of $\mathbf{E}_\epsilon$, instead of searching the minimal combination satisfying the first t constraints among all the possible combinations, we can concentrate on a set of less than τ possibilities. Then we update the rest of Sol in order to simplify the search of solutions in the next iterations.

Note that all the set operations, computations of $\gamma[S]$, $v(S)$, and $S_\epsilon(T_{min}(t))$ can be done (using algorithm EKP as a subroutine) in time $O(\tau)$; all loops have at most τ iterations and we have a maximum of two interlocked loops. Hence Algorithm Main runs in $O(\tau^3)$.

Next we argue that Algorithm Main returns a solution if and only if the given instance $\mathcal{I}$ of PS($\frac{1}{2}$) is feasible. This can be seen through the following chain of equivalences:

$$\begin{aligned} \text{there is a solution} &\iff \mathcal{T} \text{ is a solution} \\ &\iff \forall t \in \mathcal{T} \quad T_{min}(t) \leq \sum_{k \in [1..t]} \epsilon^{\tau-k} \\ &\iff \forall t \in \mathcal{T} \quad v(S_\epsilon(T_{min}(t))) \leq \sum_{k \in [1..t]} \epsilon^{\tau-k} \\ &\qquad\qquad\qquad\qquad\qquad\qquad \text{(by definition of } \epsilon\text{-decomposition)} \\ &\iff \text{Main returns a solution} \qquad \text{(by left independence),} \end{aligned}$$

Algorithm Main

```
Require: $\mathbf{T}_{min}$, $\gamma$, $\epsilon$
 1: let Sol be an array of sets all initialized to $\emptyset$
 2: let Aux := $\emptyset$
 3: for $t = 1$ to $\tau$ do
 4:     let Sol[$t$] be the $\epsilon$-decomposition of $T_{min}(t)$
 5:     if Sol[$t$] $\cap$ [$t + 1..\tau$] $\neq \emptyset$ then print ”no solution” and exit
 6:     end if
 7: end for
 8: for $t := 1$ to $\tau$ do
 9:     if Sol[$t$] $\neq \emptyset$ then
10:         Aux := Sol[min Sol[$t$] $- 1$] $\cup$ Sol[$t$]
11:         for $k \in$ [min Sol[$t$].. $t$]$\setminus$Sol[$t$] do
12:             if $\gamma$[Aux] $> \gamma$[Sol[$k - 1$] $\cup \{k\} \cup$ (Sol[$t$] $\setminus$ [1..$k$])] then
13:                 Aux := Sol[$k - 1$] $\cup \{k\} \cup$ (Sol[$t$] $\setminus$ [1..$k$])
14:             end if
15:         end for
16:         Sol[t] := Aux
17:         for $j := t + 1$ to $\tau$ do
18:             if $v$(Sol[$j$] $\cap$ [1..$t$]) $\leq v$(Sol[$t$]) then Sol[$j$] := Sol[$j$] $\setminus$ [1..$t$]
19:             end if
20:         end for
21:     end if
22: end for
23: return Sol[$\tau$]
```

To complete the proof of the Theorem we need to argue that if S is returned by Main it is an optimal solution to $\mathcal{I}$. In what follows, given an instance $\mathcal{I}$ of PS($\frac{1}{2}$), for $t \in \mathcal{T}$, denote by $\mathcal{I}_t$ the sub-problem of obtained using $(\mathbf{E}_\epsilon)_t$, the sub-matrix of $\mathbf{E}_\epsilon$ formed by the first t rows and columns, and the prefixes $(\mathbf{T}_{min})_t$ and γ_t. Note that $\mathcal{I}$ coincides with $\mathcal{I}_\tau$. Also if $1 \leq k < t$ and S is such that $(\mathbf{T}_{min})_t \leq (\mathbf{E}_\epsilon)_t \cdot \boldsymbol{\chi}_S$ then $(\mathbf{T}_{min})_k \leq (\mathbf{E}_\epsilon)_k \cdot \boldsymbol{\chi}_{S \cap [1..k]}$, since, for each $j \in \{1, \ldots, k\}$, only the first j elements of the jth row of $\mathbf{E}_\epsilon$ are non-zero.

Assume that we ran Main and that the algorithm returned a solutions. We will show by induction on t that $\mathsf{Sol}[t]$ (seen as a set in $[1..t]$) is a solution to the sub-problem of size t. The case $t = 1$ is obvious. We suppose now that $\mathsf{Sol}[k]$ is a solution to the sub-problem of size k for $1 \leq k < t$ and we want to show that $\mathsf{Sol}[t]$ is a solution to the sub-problem of size t.

First show that we have $(\mathbf{T}_{min})_t \leq (\mathbf{E}_\epsilon)_t \cdot \boldsymbol{\chi}_{\mathsf{Sol}[t]}$. By construction we have $\mathsf{Sol}[t] = \mathsf{Sol}[k-1] \cup \{k\} \cup (S_\epsilon(T_{min}(t)) \setminus [1..k])$ for $m \leq k \leq t$ with m being the minimum of $\mathsf{Sol}[t]$ at the begining of the tth iteration. Therefore, by induction hypothesis $(\mathbf{T}_{min})_{k-1} \leq (\mathbf{E}_\epsilon)_{k-1} \cdot \boldsymbol{\chi}_{\mathsf{Sol}[k-1]} = (\mathbf{E}_\epsilon)_{k-1} \cdot \boldsymbol{\chi}_{\mathsf{Sol}[t] \cap [1..k-1]}$. Let's now consider $k \leq j < t$. We must have $v(\mathsf{Sol}[j]) < v(S_\epsilon(T_{min}(t)) \cap [m..j])$ since otherwise we would have $j < m$ because of the for loop on lines 17 to 20 in Algorithm Main, which isn't possible since $m \leq k \leq j$. So, by induction hypothesis, $T_{min}(j) \leq v(\mathsf{Sol}[j]) < v(S_\epsilon(T_{min}(t)) \cap [m..j]) \leq v(S_\epsilon(T_{min}(t)) \cap [1..j]) < v(\mathsf{Sol}[t] \cap [1..j]$ by Lemma 3, since $k \notin S_\epsilon(T_{min}(t))$, and this for all $j \in [k..t-1]$. Finally by Lemma 3 we also have $T_{min}(t) \leq v(S_\epsilon(T_{min}(t))) \leq v(\mathsf{Sol}[t])$.

Now take another feasible solution $S \subset [1..t]$. We will show that $\gamma[\mathsf{Sol}[t]] \leq \gamma[S]$. Since $(\mathbf{T}_{min})_t \leq (\mathbf{E}_\epsilon)_t \cdot \chi_S$, we also have that $v(S_\epsilon(T_{min}(t))) \leq v(S)$ and so by Lemma 3 there exist $j \in [1..t] \setminus S_\epsilon(T_{min}(t)) \cup \min S_\epsilon(T_{min}(t))$ and $S' \in [1..j-1]$ such that $S = (S_\epsilon(T_{min}(t)) \setminus [1..j]) \cup \{j\} \cup S'$ (with $j = \min S_\epsilon(T_{min}(t))$) in case of equality since v is injective). If $j < m$ then we have $S \setminus [1..m-1] = S_\epsilon(T_{min}(t)) \setminus [1..m-1]$ and so by induction hypothesis we have:

$$\begin{aligned} \gamma[\mathsf{Sol}[m-1]] &\leq \gamma[S \cap [1..m-1]] \\ \gamma[\mathsf{Sol}[m-1]] + \gamma[S_\epsilon(T_{min}(t)) \setminus [1..m-1]] &\leq \gamma[S \cap [1..m-1]] + \gamma[S \setminus [1..m-1]] \\ \gamma[\mathsf{Sol}[m-1] \cup \{m\} \cup (S_\epsilon(T_{min}(t)) \setminus [1..m])] &\leq \gamma[S] \end{aligned}$$

where the last inequality follows from Lemma 3 and since $m \in S_\epsilon(T_{min}(t))$. This gives $\gamma[\mathsf{Sol}[t]] \leq \gamma[S]$ since $\gamma[\mathsf{Sol}[t]] \leq \gamma[\mathsf{Sol}[m-1]] \cup \{m\} \cup (S_\epsilon(T_{min}(t)) \setminus [1..m]))$ by construction. Now if $j \geq m$ similarly we have $S \setminus [1..j-1] = \{j\} \cup (S_\epsilon(T_{min}(t)) \setminus [1..j])$ and so:

$$\begin{aligned} \gamma[\mathsf{Sol}[j-1]] &\leq \gamma[S \cap [1..j-1]] \\ \gamma[\mathsf{Sol}[j-1]] + \gamma[\{j\} \cup S_\epsilon(T_{min}(t)) \setminus [1..j]] &\leq \gamma[S \cap [1..m-1]] + \gamma[S \setminus [1..j-1]] \\ \gamma[\mathsf{Sol}[j-1] \cup \{j\} \cup (S_\epsilon(T_{min}(t)) \setminus [1..j])] &\leq \gamma[S], \end{aligned}$$

which gives us again $\gamma[\mathsf{Sol}[t]] \leq \gamma[S]$.

4 Conclusion

We studied a number of energy allocation optimization problems which may occur in domestic buildings. Two broad cases were considered: a "must-use" scenario where a set of appliances must be scheduled over a given time horizon, and a "comfort-aware" scenario where the appliances help to satisfy a predefined environment comfort level. In all cases we were interested in minimal energy cost solutions. Our main goal was to investigate the computational complexity of the relevant problems and characterize the border between polynomial-time tractability and NP-hardness. We studied the effect of the number of appliances on the complexity of problems of the first type, and that of the type and distribution of the AC units, as well as the energy price and the thermal properties of the given environment on problems of the second type. The main result of the paper is a proof that although it is NP-hard to schedule the operation of a single air-conditioning (AC) unit, working at various temperature levels in a variable energy price regime, there is a polynomial time algorithm for controlling one such device working at a single temperature level, for houses with low thermal inertia. The proof of such result uses the algorithmic properties of a variant of the well-known KNAPSACK problem.

References

1. European SmartGrids Technology Platform. Directorate-General for Research. Sustainable Energy Systems. Report EUR 22040, 2006. Available from `https://ec.europa.eu/research/energy/pdf/smartgrids_en.pdf`. (July 2016).
2. What is the Smart Grid? U.S. Department of Energy, 2016. URL: `https://www.smartgrid.gov/the_smart_grid/` [July 2016].

3. Agnetis, A., de Pascale, G., Detti, P., Vicino, A.: Load scheduling for household energy consumption optimization. *IEEE Transactions on Smart Grid*, 4, 2364–2373 (2013)
4. Arikiez, M. K., Gatens, P., Grasso, F., Zito, M. Smart domestic renewable energy management using knapsack. In *IEEE PES ISGT Europe*, 2013.
5. Arikiez, M., Grasso, F., Zito, M.: Heuristic algorithm for coordinating smart house in microgrid. In *IEEE International Conference on Smart Grid Communications*, (2015)
6. Azar, Y. On-line load balancing. In A. Fiat, editor, *Online Algorithms*, volume 1442 of *LNCS*, pages 178–195. Springer-Verlag, 1996.
7. Burcea, M., Hon, W.-K., Liu, H.-H., Wong, P.W.H., Yau, D.K.Y. Scheduling for electricity cost in smart grid. In P. Widmayer, Y. Xu, and B. Zhu, editors, *Computing and Combinatorics; 19th Annual International Conference, COCOON 2013*, volume 8287 of *LNCS*, pages 306–317. Springer-Verlag, 2013.
8. T. Carnes and D. Shmoys. Primal-dual schema for capacitated covering problems. In A. Lodi, A. Panconesi, and G. Rinaldi, editors, *Integer Programming and Combinatorial Optimization: 13th International Conference, IPCO 2008*, volume 5035 of *Lecture Notes in Computer Science*, pages 288–302. Springer-Verlag, 2008.
9. V. T. Chakaravarthy, A. Kumar, G. R. Parija, S. Roy, and Y. Sabharwal. Minimum cost resource allocation for meeting job requirements. In *2011 IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 14–23. CPS, 2011.
10. J. Csirik, J. B. G. Frenk, M. Labbé, and S. Zhang. Heuristics for the 0-1 min-knapsack problem. *Acta Cybernetica*, 10(1–2):15–20, 1991.
11. Dounis, A. I., Caraiscos, C.: Advanced control systems engineering for energy and comfort management in a building environment a review. *Renewable and Sustainable Energy Reviews*, 13, 1246–1261 (2009)
12. Garey, M., Johnson, D.: Strong NP-completeness results: Motivation, examples, and implications. *Journal of the ACM* 25, 499–508 (1978)
13. Garey, M., Johnson, D.: *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company (1979)
14. Huang, Z., Zhu, T., Gu, Y., Irwin, D., Mishra, A., Shenoy, P.: Minimizing electricity costs by sharing energy in sustainable microgrids. In *Proceedings of the 1st ACM Conference on Embedded Systems for Energy-Efficient Buildings*, 120–129 (2014)
15. Hubert, T., Grijalva, S.: Modeling for residential electricity optimization in dynamic pricing environments. *IEEE Transactions on Smart Grid*, 3, 2224–2231 (2012)
16. Koutsopoulos, I., Tassiulas., L. Control and optimization meet the smart power grid: scheduling of power demands for optimal energy management. In *Proceedings of the 2nd International Conference on Energy-Efficient Computing and Networking*, pages 41–50, 2011.
17. Tsui, K. M., Chan, S. C.: Demand response optimization for smart home scheduling under real-time pricing. *IEEE Transactions on Smart Grid*, 3, 1812–1821, (2012)