Deferrable loads in an energy market: coordination under congestion constraints

Donatello Materassi, Saverio Bolognani, Mardavij Roozbehani, Munther A. Dahleh

***Abstract*— We consider a scenario where price-responsive energy consumers are allowed to optimize their individual utilities via mechanisms of load-shifting in a distribution network subject to capacity constraints. The uncoordinated selfish behavior of the consumers would lead, in general, to requests that could not be served by the distribution network because of such constraints. Thus, a centralized or hierarchically coordination mechanism is required. We derive algorithms and methods to determine in real-time the largest set of consumers' decisions that are compatible with the physical constraints of the network and capable of avoiding congestion phenomena in the future.**

These methods are shown to be applicable to the design of coordination mechanisms with the aim of providing a large number of degrees of freedom to the users while guaranteeing the integrity of the system.

I. Introduction

The introduction of mechanisms of price-response in the power grid is expected to happen via the design, implementation, and deployment, of a large number of individual decision makers corresponding to different participants in the energy market, each one optimizing its own independent utility function [1], [2], [3], [4]. The behavior of the individual user necessarily depends on its specific utility of consumption, which, typically, is not just a static function of the delivered electric power. Indeed, in many noticeable cases – Electric vehicles (EV), thermostatic loads, heating, ventilation, and air conditioning (HVAC) systems, manufacturing processes, and residential loads – the user utility is described by quite complex individual models that might include the possibility of deferring the load, a penalty for postponing consumption and soft or hard deadlines.

The impact of uncoordinated response of flexible loads to a price signal has been analyzed, for example in [5], [6], [7]. In particular, [7] studies the effect of a widespread adoption of Electric Vehicles (EVs) on a large scale, while [6] considers the impact of EVs in terms of power losses at the level of the distribution network. These studies suggest that, in order to guarantee the reliability of the grid, individual users cannot be allowed to participate freely in the retail energy market, and some level of supervision is needed. This is the main focus of this article, where we enlarge the problem perspective from the determination of optimal consumption policies of a single energy user, e.g. [8], to the scenario where multiple energy users need to be coordinated to meet the physical constraints imposed by the system.

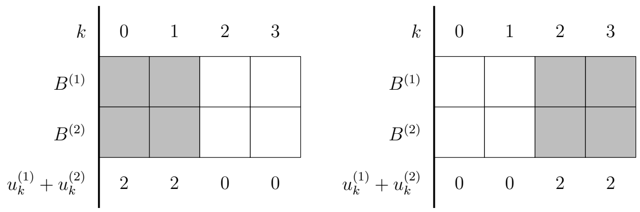
In particular, we consider a specific class of congestion phenomena in the power network, consisting in overstepping capacity limits on the maximum aggregate power that can be delivered to a pool of users at the same time. Indeed, there are many practical scenarios where the operational constraints have this form - for example large fleet of electric vehicles connected to the same charging station, where each vehicle has to satisfy its energy demand (full battery charge) within some firm deadline. It is also reasonable to expect that the limit on the maximum power at the distribution substation would correspond to constraints of the same type.

This type of constraints have already been analyzed in [9], for example, where however no stochastic prices are considered. Heuristic scheduling strategies have been proposed in [10], in order to schedule deferrable loads in a energy market, for better exploitation of renewable sources and of reserve capacity. A decentralized protocol for day-ahead load scheduling has been proposed in [11], aiming at flat aggregate consumption profiles, while both [12] and [13] consider the issue of power losses minimization in their centralized load control. In all these examples, loads are dispatched by a (possibly distributed) scheduler, to which the user have to reveal information about their utility function. Our formulation, instead, intends to provide a framework where individual policies of the users in response to stochastic energy prices can be incorporated seamlessly, so that the aggregator has to guarantee only the satisfaction of operational constraints of the grid, while users are allowed to participate to the market according to their own strategy and interests, without even disclosing them.

We can use a simple example to describe the mechanism that make congestion to arise when different users are allowed to pursue their own individual utility with no supervision. Consider the problem of charging two identical batteries, $B^{(1)}$ and $B^{(2)}$. Assume that each battery requires a total of $E^{(1)} = E^{(2)} = 2$ units of energy by the time $T = 4$. At each time $k = 0, 1, 2, 3,$, the i-th battery can be charged by $u_k^{(i)} \geq 0$ units of energy with a maximum rate of $\overline{u}^{(i)} = 1$ per time unit, for $i = 1, 2$. In other terms, we have that $0 \leq u_k^{(i)} \leq \overline{u}^{(i)}$ for $k = 0, 1, 2, 3$ and $i = 1, 2$. In addition, assume that the two batteries are subject to the capacity constraint $\overline{U}$ of their charging station: $u_k^{(1)} + u_k^{(2)} \leq \overline{U} = 1$ for $k = 0, 1, 2, 3$ and, being identical, are going to follow the same decision strategy - for example the optimal cost-minimizing strategy proposed in [8].

The authors are with the Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge MA 02139, USA. {donnie13, saverio, mardavij, dahleh}@mit.edu

The research leading to these results has received support from Draper Laboratories (URAD project), Siemens AG, and National Science Foundation (CPS-1135843).



(a) (b)

Fig. 1. Example of uncoordinated response of two batteries in a regime of dynamic-pricing. In (a) we see the response associated to to *low* energy prices for $k = 0, 1$: both batteries decide to consume. In (b) we see the response to *low* prices for $k = 0, 1$: both batteries postpone consumption, activating too late when they cannot consume at the same time.

According to such a strategy, in the case of a "low" energy price during the time intervals $k = 0, 1$ (see Figure 1a) both users would decide to consume at their maximum rate, overstepping the global constraint given by $u_k^{(1)} + u_k^{(2)} \leq \overline{U} = 1$. Via simple contention mechanisms, it might be possible to split the total rate $\overline{U}$ fairly between the two batteries, so that both users would have their demand satisfied by their deadlines. However, a more complex case of congestion happens in the case of "high" prices during the time intervals $k = 0, 1$ (see Figure 1b). In this case both batteries would postpone their consumption, expecting lower prices in the future. However, as soon as they both decide not to consume at some time k, they end up in a condition where their demands can not be satisfied any more within their deadlines.

Our results suggest that a possible architecture capable of avoiding congestion phenomena consists of an *aggregator* that has the duty of collecting bids from the individual batteries and clearing a auction. "Aggregators" that guarantee the correct behavior of the power grid have already been proposed (see [5] and [7] and references therein). However, the functions of these entities are often described in a qualitative way, without providing a formal characterization of what their behavior or their implementation should be.

Our main contribution is to precisely characterize the acceptable decisions from the the users that guarantee a feasible consumption and the satisfaction at the specified deadlines, without jeopardizing the schedulability of the charging problem in the future consumption horizon. In particular, we show that, the space of admissible consumption patterns can be algorithmically computed, and has the structure of a Partially Ordered set (POset). Such structure can be incorporated in the energy auction, in the form of a "constrained multi-object auctions" similar to those that have been studied in [14], [15], [16].

The paper has the following structure. We first provide a mathematical formulation of the problem (Section II), describe the solution in the simplified scenario in which all users have the same deadline (Section III), and then we extend the solution to the general case (Section IV). In Section V we illustrate one possible way to use the proposed methodology in the design of a constrained auction for the energy market. Due to space constraints, the proofs have been omitted, and are available online [17].

II. Problem formulation

Each deferrable load (e.g. battery) is modeled as a *task*, defined as follows.[1]

Definition 1 (Task): A task is a 3-tuple $(E, T, \overline{u}) \in \mathbb{R}^3$ where $E > 0$ defines the demand, T is the deadline, i.e. the time left to satisfy such demand, and $\overline{u} > 0$ is the maximum rate at which the task can be served.

In line with many practical applications, and with minimal loss of generality, we introduce a discretization of time and of the parameters that define the tasks. We assume a unitary sampling time and introduce the following definition.

Definition 2 (Integer task): A task $(E, T, \overline{u})$ is *integer* if $T \in \mathbb{N}$ and if $\frac{E}{\overline{u}} \in \mathbb{N}$.

We can therefore introduce the following discrete time model to describe the dynamic process of task completion (or battery charging)

$$\begin{aligned} x_{k+1} &= x_k + u_k \\ x_0 &= -E \end{aligned}$$

where $x_k \leq 0$ is the state of completion of the task (the state of charge), and $u_k \geq 0$ is the rate at which the task is served (in a piece-wise constant way) during the k-th interval, namely between time k and $k + 1$.

The constraint on the maximum rate, and the deadline for task completion, corresponds respectively to the inequalities

$$0 \leq u_k \leq \overline{u}, \quad \text{for } k = 0, \dots, T - 1,$$

and the equality

$$x_T = 0.$$

We now consider the case of multiple users, whose behaviors are coupled by the fact that, at any time, the aggregate service rate (the sum of the rates at which different tasks are served) cannot exceed a given limit $\overline{U}$.

We therefore consider N tasks $(E^{(i)}, T^{(i)}, \overline{u}^{(i)})$, where $i = 1, ..., N$, and we consider the following problem.

Definition 3 (Task scheduling problem): We define the *task scheduling problem* as the problem of deciding the variables $u_k^{(i)}$ for $k = 0, ..., T := \max_{1 \leq i \leq N} T^{(i)}$ such that for any task i

$$\begin{aligned} x_{k+1}^{(i)} &= x_k^{(i)} + u_k^{(i)} \\ x_0^{(i)} &= -E^{(i)} \\ x_{T^{(i)}}^{(i)} &= 0 \end{aligned} \tag{1}$$

where $x_k^{(i)} \leq 0$ is the state of task i at time k, and such that, at any time k,

$$\begin{aligned} &0 \leq u_k^{(i)} \leq \overline{u}^{(i)} \quad \text{for } i = 1, ..., N \\ &\sum_{i=1}^{N} u_k^{(i)} \leq \overline{U}. \end{aligned} \tag{2}$$

We refer to T as the horizon of the problem.

[1]The generality of the terminology introduced in this section indicates how the same model can effectively describe other equivalent scenarios, e.g. real-time scheduling in multiprocessor machines [18].

We introduce the following definitions.

Definition 4 (Action): We define the *action* $\mathbf{u}_k$ at time k as the vector in $\mathbb{R}^N$ obtained by stacking the rates $u_k^{(i)}$ of the different tasks.

Definition 5 (Feasible schedule): Given the tasks $(E^{(i)}, T^{(i)}, \overline{u}^{(i)})$ for $i = 1, ..., N$, and an aggregate limit $\overline{U}$, a schedule (i.e. a series of actions) $\{\mathbf{u}_k\}_{k=0}^{T-1}$ is *feasible* if it solves the task scheduling problem, i.e. it satisfies the constraints (1) and (2).

Definition 6 (Schedulable tasks): Given an aggregate limit $\overline{U}$, the tasks $(E^{(i)}, T^{(i)}, \overline{u}^{(i)})$ for $i = 1, ..., N$, are *schedulable* if a feasible schedule for the corresponding task scheduling problem exists.

As explained in the introduction, the goal of the analysis presented in this paper is to give necessary and sufficient conditions so that the actions of the users meet the constraints of the task scheduling problem, and do not compromise the possibility of generating a feasible schedule in the future, given the current state of completion of the tasks.

Our analysis is performed at time $k = 0$ and provides guarantees that, if the tasks are schedulable to begin with, then a certain action $\mathbf{u}_0$ leaves the tasks in a schedulable configuration over the horizon $[1, \max_{1 \le i \le N} T^{(i)}]$. Then, the same approach can be iteratively applied.

An effective characterization of the set of all (and only) the admissible actions allows to derive coordination strategies among the users, possibly in the form of an aggregator, as explained in the Introduction.

We therefore introduce the following formal definitions.

Definition 7 (Admissible action): Given the tasks $(E^{(i)}, T^{(i)}, \overline{u}^{(i)})$ for $i = 1, ..., N$, and an aggregate limit $\overline{U}$, an action

$$\mathbf{u}_0 = \begin{bmatrix} u_0^{(1)} \\ \vdots \\ u_0^{(N)} \end{bmatrix}$$

is *admissible* if:

- it satisfies

$$0 \le u_0^{(i)} \le \overline{u}^{(i)},\ i = 1, \ldots, N, \quad \text{and} \quad \sum_{i=1}^{N} u_0^{(i)} \le \overline{U},$$

- the tasks $(E^{(i)} - u_0^{(i)}, T^{(i)} - 1, \overline{u}^{(i)})$ are schedulable with the global constraint $\overline{U}$.

We define by $\mathcal{U}_0$ the set of all admissible actions.

We now make a technical assumption, with minimal loss of generality.

Assumption 8: The tasks $(E^{(i)}, T^{(i)}, \overline{u}^{(i)})$, $i = 1, ..., N$, are all integer tasks. Moreover, there exist a common $\overline{u}$, that we denote as the *unit rate*, such that

$$\frac{E^{(i)}}{\overline{u}} \in \mathbb{N}, \quad \frac{\overline{u}^{(i)}}{\overline{u}} \in \mathbb{N}, \quad \frac{\overline{U}}{\overline{u}} \in \mathbb{N},$$

for all $i = 1, ..., N$.

Based on this, we introduce the following definition.

Definition 9 (Integer schedule): Given a task scheduling problem, a schedule is integer if all its elements are integer multiples of a unit rate $\overline{u}$.

In the following, we will focus only on *integer schedules*. This choice is reasonable in the common practice, where most of the time tasks can be served at some quantized rate of service (i.e. according to the number of processors assigned to the thread in a multiprocessor system [18], or according to some of the EV battery charging standards where different levels of charging speed are available).

It is worth remarking that, even if the problem of finding a solution to the scheduling problem (1)-(2) becomes easier if the integer constraints are relaxed (and reduces to the application of linear programming to load scheduling [19], [20]), a practical characterization of all the feasible actions remains a difficult problem. On the other hand, the following result shows that the restriction to integer schedules has no effect on the schedulability analysis of the task scheduling problem.

Theorem 10: Consider a set of N tasks $(E^{(i)}, T^{(i)}, \overline{u}^{(i)})$, with an aggregate bound $\overline{U}$, and let Assumption 8 hold. The tasks are schedulable if and only if an integer schedule exists.

Moreover, there is no loss of generality by assuming that all tasks have the same rate constraint $\overline{u}^{(i)} = 1$, as the general case can always be reformulated in this way, by "splitting" each task into smaller tasks with homogeneous rate constraints [17]. Therefore, the choice of considering only integer schedules corresponds to having schedules with binary decision variables, i.e.

$$u_k^{(i)} \in \{0, 1\}, \quad \text{for } k =, ..., T-1;\ i = 1, ..., N.$$

Definition 11 (Aggregate service rate): Given an action $\mathbf{u}_0$, we indicate the *aggregate service rate* as

$$\|\mathbf{u}_0\|_1 = \sum_{i=1}^{N} u_0^{(i)}. \tag{3}$$

We also borrow the notion of slack from the area of multi-processor scheduling [18].

Definition 12 (Slack): We define the *slack* of a task as

$$s^{(i)} := T^{(i)} - E^{(i)} \tag{4}$$

The slack represents the maximum idle time that the task can wait, before having to be served, in order to meet its deadline (remember that $\overline{u}^{(i)} = 1$). Based on this definition, we introduce a function of the action $\mathbf{u}_0$ that returns the slacks of the tasks that are served according to that action:

$$\begin{aligned} &\mathbf{s} : \{0,1\}^N \to \mathbb{N}^{\|\mathbf{u}_0\|_1} \\ &\mathbf{s}(\mathbf{u}_0) = \left[s^{(i)} \right]_{i: u_0^{(i)} = 1}. \end{aligned} \tag{5}$$

As an example, in the case of an action serving tasks 2 and 3 among 5 tasks, we have that

$$\mathbf{s}\left(\begin{bmatrix} 0 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \right) = \begin{bmatrix} s^{(2)} \\ s^{(3)} \end{bmatrix}.$$

It is possible to define a *partial ordering* relation between these vectors via the following definition.

Definition 13 (Slack partial ordering): Let $\mathbf{s}'$ and $\mathbf{s}''$ be two vectors of the dimensions d' and d'', respectively. We say that

$$\mathbf{s}' \preccurlyeq \mathbf{s}''$$

if $d' \geq d''$ and the sorted permutation of the d'' smallest elements of $\mathbf{s}'$ is element-wise smaller than the sorted permutation of $\mathbf{s}''$.

As an example, we have that $\left[\begin{smallmatrix}1\\2\end{smallmatrix}\right] \preccurlyeq \left[\begin{smallmatrix}3\\5\end{smallmatrix}\right]$, and also $\begin{bmatrix}1\\2\\8\end{bmatrix} \preccurlyeq \left[\begin{smallmatrix}3\\5\end{smallmatrix}\right]$, while $\left[\begin{smallmatrix}1\\4\end{smallmatrix}\right]$ and $\left[\begin{smallmatrix}2\\3\end{smallmatrix}\right]$ are not comparable, and neither $\begin{bmatrix}1\\4\\8\end{bmatrix}$ and $\left[\begin{smallmatrix}2\\3\end{smallmatrix}\right]$ are.

III. Characterization of admissible policies (homogeneous deadlines)

In this section we consider the case where all the tasks have the same deadline, namely $T^{(i)} = T$, for $i = 1, ..., N$. The following result characterize the schedulability of a set of tasks, for generic maximum rates $\overline{u}^{(i)}$.

Lemma 14: Consider a set of N tasks $(E^{(i)}, T, \overline{u}^{(i)})$, for $i = 1, ..., N$, sharing the same deadline T, and let Assumption 8 hold. The tasks are schedulable if and only if the two following conditions are met

$$\sum_{i=1}^{N} \frac{E^{(i)}}{\overline{U}} \leq T, \qquad \max_{1 \leq i \leq N} \frac{E^{(i)}}{\overline{u}^{(i)}} \leq T. \tag{6}$$

The characterization of schedulability proposed in Lemma 14 is the key tool that allows to give a complete characterization of the set $\mathcal{U}_0$ of admissible decision.

We can therefore state the main result for the case of homogeneous deadlines.

Theorem 15: Consider a set of N schedulable tasks $(E^{(i)}, T, 1)$, for $i = 1, ..., N$, sharing the same deadline T, and let Assumption 8 hold. Then the set of admissible integer decisions at time 0 is

$$\mathcal{U}_0 = \bigcup_{y=\max\{\underline{U}, z\}}^{\overline{U}} \left\{ \mathbf{u}_0 : \|\mathbf{u}_0\|_1 = y \text{ and } \mathbf{s}(\mathbf{u}_0) \preccurlyeq \mathbf{0}_z \right\},$$

where z is the number of tasks with slack zero

$$z = |\{i : s^{(i)} = 0\}|,$$

the symbol $\mathbf{0}_z$ indicates a vector of size z in which all elements are 0, and $\underline{U}$ is defined as

$$\underline{U} = \sum_{i=1}^{N} E^{(i)} - (T-1)\overline{U}. \tag{7}$$

Notice that the characterization of the set of admissible decisions $\mathcal{U}_0$ can be translated in the two following necessary and sufficient conditions:

- the aggregate service rate $\|\mathbf{u}_0\|_1$ of all the tasks cannot be smaller than $\underline{U}$, as defined in (7);
- tasks with slack $s^{(i)}$ equal to zero must be served.

A. The two-battery problem revisited

Let us consider again the problem of charging two indentical batteries that was presented in the introduction and let us analyze it in light of the results obtained in this section. We can formalize it as a task scheduling problem where $(E^{(1)}, T^{(1)}, \overline{u}^{(1)}) = (E^{(2)}, T^{(1)}, \overline{u}^{(1)}) = (2, 4, 1)$ with an aggregate limit $\overline{U} = 1$. From Equation (7), we can compute $\underline{U} = 1$, while from Equation (4) we can compute $s^{(1)} = s^{(2)} = 2$. As both the slacks are greater than zero, from Theorem 15 we know that any decision such that $\|\mathbf{u}_0\|_1 = 1$ is admissible, and thus

$$\mathcal{U}_0 := \{(1,0)^T, (0,1)^T\}.$$

In other words, at least one of the two batteries has to consume at $k = 0$, otherwise feasibility of the problem will be irreversibly lost at $k = 1$. We showed in the Introduction how, in the case of a high price at time $k = 0$, neither of the two batteries would consume if they are following a selfish optimal strategy for cost minimization. Our result suggests that the existence of an Aggregator monitoring the situation and implementing, for example, a constrained multi-object auction to assign a minimal amount of energy when needed, is a viable solution to the congestion problem.

IV. Characterization of admissible policies (heterogeneous deadlines)

In this section, we study the set $\mathcal{U}_0$ of admissible decisions in the case of batteries with different deadlines.

We first present an algorithm which allows to test the schedulability of a given set of tasks.

Minimum Effort Algorithm

INPUT

- tasks: $(E^{(i)}, T^{(i)}, 1)$, $i = 1, ..., N$
- aggregate limit: $\overline{U}$

EXECUTE

1) INITIALIZE
   - $T := \max_{1 \leq i \leq N} T^{(i)}$
   - $u_k^{(i)} = 0$ for $0 \leq k < T$, $1 \leq i \leq N$
2) FOR $k = T-1 : -1 : 0$
   - Create **ActiveTasks**:= list of all tasks with deadline $T^{(i)} > k$ and demand $E^{(i)} > 0$
   - Compute N_A := **#** of elements in **ActiveTasks**
   - Order **ActiveTasks** according to their *reverse* slack $r^{(i)} := k + 1 - E^{(i)}$
   - For the first $\min(N_A, \overline{U})$ tasks with least reverse slack $r^{(i)}$ in **ActiveTasks**, decrease $E^{(i)}$ by 1 and assign $u_k^{(i)} := 1$

   OUTPUT:
   - schedulability: if $E^{(i)} = 0$ for all $i = 1, ..., N$ return **TRUE** otherwise return **FALSE**
   - schedule: $u_k^{(i)} = 0$ for $0 \leq k < T$, $1 \leq i \leq N$
   - effort: $\underline{U} := \sum_{i=1}^{N} u_0^{(i)}$

The algorithm is clearly polynomial in the number of tasks and in the length of the time horizon (the longest deadline). An example of application of the Minimum Effort Algorithm is presented in Figure 2 for the completion of 7 tasks with heterogeneous deadlines.

The following result shows how the proposed Minimum Effort Algorithm is effective in finding a feasible schedule, if such a schedule exists.

Theorem 16: Consider the tasks $(E^{(i)}, T^{(i)}, 1)$ for $i = 1, ..., N$, together with the aggregate constraint $\overline{U}$, and let

Assumption 8 hold. The tasks are schedulable if and only if the Minimum Effort Algorithm returns **TRUE**. Also, if the tasks are schedulable the Minimum Effort Algorithm returns a feasible integer schedule.

The scheduling generated by the algorithm exhibit one specific property that will be useful in the characterization of the set $\mathcal{U}_0$ of admissible decisions, and gives the name to the algorithm.

Lemma 17: Consider the tasks $(E^{(i)}, T^{(i)}, 1)$ for $i = 1, ..., N$, together with the aggregate constraint $\overline{U}$, and let Assumption 8 hold. Let $\underline{U}$ be the effort computed by the Minimum Effort Algorithm applied to such task scheduling problem. Then for any integer admissible decision $\mathbf{u}_0 \in \mathcal{U}_0$, we necessarily have that

$$\|\mathbf{u}_0\|_1 \geq \underline{U}.$$

Lemma 17 states that the Minimum Effort Algorithm returns a lower bound $\underline{U}$ on the aggregate consumption that must happen at time 0, in order to maintain schedulability of the problem at the next time step. Also, the Minimum Effort Algorithm guarantees the existence of a feasible schedule with aggregate consumption exactly equal to $\underline{U}$ at time $k = 0$. However, an aggregate consumption larger or equal to $\underline{U}$ at time $k = 0$ is not a sufficient condition. Indeed, consider for example the task scheduling problem represented in Figure 2. The Minimum Effort Algorithm returned $\underline{U} = 3$, therefore via Lemma 17 we know that the aggregate consumption at time 0 must be at least 3, for any feasible scheduling. However, one can easily check that the decision of serving tasks $B^{(1)}$, $B^{(4)}$, and $B^{(7)}$ is not admissible: in fact, the system was scheduled to work at full aggregate rate $\overline{U}$ up to time $k = 4$, and therefore there is no way to serve task $B^{(7)}$ at time 0, as there is no other task that can be served at time 5 or later.

Conversely, it is also clear that the decision computed by the algorithm is not the only admissible one. In the same example, it is easy to check that the decision of serving tasks $B^{(1)}$, $B^{(3)}$, and $B^{(6)}$ is also admissible.

The following result clarifies what is the mathematical structure of the set $\mathcal{U}_0$ of admissible decisions, via the partial ordering relation introduced in Section II.

Theorem 18: Consider the tasks $(E^{(i)}, T^{(i)}, 1)$ for $i = 1, ..., N$, together with the aggregate constraint $\overline{U}$, and let Assumption 8 hold. Let $\mathbf{u}_0$ be an admissible decision. Then any decision $\mathbf{v}_0$ is also admissible if

$$\|\mathbf{v}_0\|_1 \leq \overline{U} \quad \text{and} \quad \mathbf{s}(\mathbf{v}_0) \preccurlyeq \mathbf{s}(\mathbf{u}_0).$$

In other words, Theorem 18 states that if a decision is admissible, then a decision smaller in slack, in the sense provided by the relation "$\preccurlyeq$", is also admissible.

The characterization given by Theorem 18 is exemplified in Figure 3. It represents the lattice of all possible choices of $\underline{U}$ tasks to be served among the set of 7 tasks already described in Figure 2. The label for each node represents the slacks associated with the tasks that are served. The Hasse diagram [21] highlights the POset structure defined by the relation "$\preccurlyeq$". In red (and on the right of the same figure) we have reported all the admissible decisions, illustrating the relation of partial ordering described by Theorem 18.

Battery	$B^{(1)}$	$B^{(2)}$	$B^{(3)}$	$B^{(4)}$	$B^{(5)}$	$B^{(6)}$	$B^{(7)}$
Demand $E^{(i)}$	3	2	4	3	1	5	1
Deadline $T^{(i)}$	3	3	5	5	5	8	8
Slack $s^{(i)}$	0	1	1	2	4	3	7

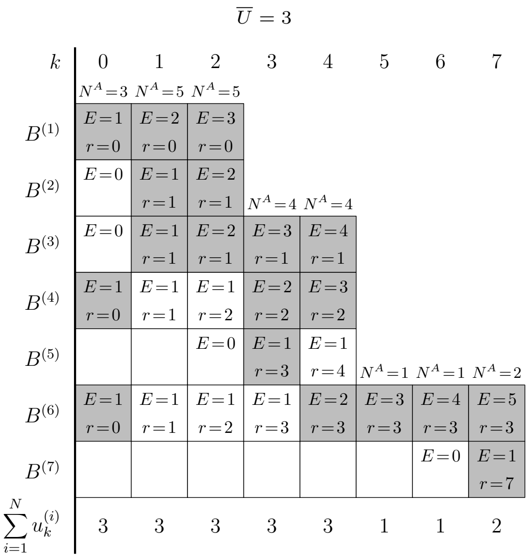


Fig. 2. A set of 7 integer tasks with heterogeneous deadlines and the schedule computed by the Minimum Effort Algorithm. The aggregate bound is $\overline{U} = 3$. A gray box indicates that at a certain time k the task i is served ($u_k^{(i)} = 1$) while a white box indicates that the task is idle ($u_k^{(i)} = 0$). Each box also contains the numerical values of $E^{(i)}$ and of the reverse slack $r^{(i)}$ as computed by the Minimum Effort Algorithm during its execution. As all the demands are served ($E^{(i)} = 0$ for all tasks after the iteration at time $k = 0$), the problem is schedulable. The aggregate rate at time $k = 0$ represents the minimum effort ($\underline{U} = 3$).

V. Example: an auction with minimum-effort least-slack constraints

The analysis in Section IV poses a lower bound on the aggregate service rate $\|\mathbf{u}_0\|_1$ of the admissible decisions, and shows that the set $\mathcal{U}_0$. has a poset structure. In the scenario that motivated this study, at every time the Aggregator has to guarantee that the action $\mathbf{u}_0$ of the users belongs to $\mathcal{U}_0$. We propose here an example of one possible and very practical way to enforce such constraint, which we denote *Minimum-effort Least-slack* (MELS) policy, and which is based on the following result.

Corollary 19: Let $\underline{U}$ be the minimal aggregate consumption as returned by the Minimum Effort Algorithm. Let $i_1, \ldots, i_{\underline{U}}$ be the $\underline{U}$ tasks with the smallest slacks (resolving ties arbitrarily). Then any decision $\mathbf{u}_0$ in

$$\mathcal{U}_0^{\text{MELS}} = \left\{ \mathbf{u}_0 : \|\mathbf{u}_0\|_1 \leq \overline{U} \text{ and } u_0^{(i)} = 1 \text{ for } i = i_1, \ldots, i_{\underline{U}} \right\}$$

is admissible, being $\mathcal{U}_0^{\text{MELS}} \subseteq \mathcal{U}_0$.

The proposed MELS policy consists in a regulated energy market described by the following steps.

1) In order to participate to the market, users declare their deadline and demand.

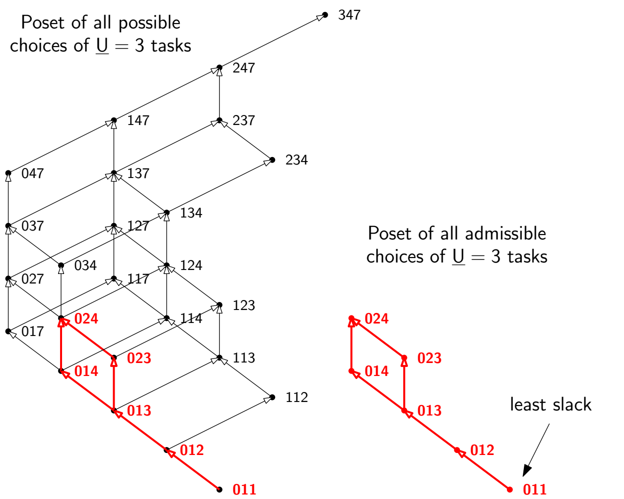


Fig. 3. The Hasse diagram on the left represents the poset structure of all possible way of choosing $\underline{U}$ batteries that consume (at full rate) at time 0. If two decisions x, y satisfy $x \preccurlyeq y$ (and therefore a directed path goes from x to y in the diagram) then x is guaranteed to be an admissible decision if y is admissible. In this example, [024] is the maximal element (the admissible decision with the largest slack), and all and only the decisions with smaller slack are admissible. The diagram on the right represents the subset of all admissible decisions. Clearly, the decision with the least slack is always admissible.

2) User are allowed to join only if their demand is feasible. Such test can be performed via the Minimum Effort Algorithm, and once a load is admitted, the aggregator guarantees that its demand will be served (in a form of commitment similar to the one in [22]).
3) At every time step, all users declare a *bid*, i.e. the price that they are willing to pay in order to consume at full rate, according to their own selfish optimization policy (like for example the one derived in [8]).
4) Based on these date, an aggregator computes the lower bound $\underline{U}$ via the Minimum Effort Algorithm.
5) The aggregator clears a constrained multi-object auction with the following constraints:
   - the $\underline{U}$ loads with the least slack must be served;
   - no more than $\overline{U}$ loads are served, in total.

This example shows how a precise characterization of the set of admissible decisions allows the users to participate in the energy market with their specific, selfish, and possibly undisclosed, policy (i.e. via their bids), while ensuring that operational constraints and deadlines are satisfied.

VI. Conclusions

In an energy market with dynamic pricing, consumers can optimize their individual utilities adopting several mechanisms among which load-shifting is one of most effective ones. However, the response of a large number of uncoordinated price-responsive consumers might lead to local or global congestion of the power distribution system.

The goal of this paper is to suggest a possible methodology to enable a large number of price-responsive consumers to participate to the market without compromising the reliability of the network. This methodology is based on the computation of the set of admissible actions that are guaranteed to preserve the integrity of the system. It is shown that this set admits a precise mathematical characterization defined by a relation of partial ordering. This set of admissible actions can be used along with a constrained multi-object auction mechanism to establish a verified decision protocol providing both the efficiency of a distributed dynamic pricing system and the reliability of a centralized approach.

References

[1] A. Papavasiliou and S. Oren, "Supplying renewable energy to deferrable loads: Algorithms and economic analysis," in *Power and Energy Society General Meeting, 2010 IEEE*. IEEE, 2010, pp. 1–8.
[2] K. Afridi, A. M. Farid, J. M. Grochow, W. W. Hogan, H. D. Jacoby, J. L. Kirtley, H. G. Michaels, I. Perez-Arriaga, D. J. Perreault, N. L. Rose, and G. L. Wilson, *The Future of the Electric Grid*. MIT Press, 2011.
[3] D. Caves, K. Eakin, and A. Faruqui, "Mitigating price spikes in wholesale markets through market-based pricing in retail markets," *The Electricity Journal*, vol. 13, no. 3, pp. 13–23, 2000.
[4] D. Kirschen, "Demand-side view of electricity markets," *Power Systems, IEEE Transactions on*, vol. 18, no. 2, pp. 520–527, 2003.
[5] M. Caramanis and J. M. Foster, "Management of electric vehicle charging to mitigate renewable generation intermittency and distribution network congestion," in *Proceedings of the 48th IEEE Conference on Decision and Control*, Dec. 2009, pp. 4717–4722.
[6] K. Clement-Nyns, E. Haesen, and J. Driesen, "The impact of charging plug-in hybrid electric vehicles on a residential distribution grid," *IEEE Trans. on Power Systems*, vol. 25, no. 1, pp. 371–380, 2010.
[7] J. A. P. Lopes, F. J. Soares, and P. M. R. Almeida, "Integration of electric vehicles in the electric power system," *Proceedings of the IEEE*, vol. 99, no. 1, pp. 168–183, 2011.
[8] D. Materassi, S. Bolognani, M. Roozbehani, and M. Dahleh, "Deferrable loads in an energy market: optimal consumption policy," in *Proc. of the American Control Conference*, 2014.
[9] O. Sundström and C. Binding, "Flexible charging optimization for electric vehicles considering distribution grid constraints," *IEEE Transactions on Smart Grid*, vol. 3, no. 1, pp. 26–37, Mar. 2012.
[10] A. Subramanian, M. Garcia, A. Dominiguez-Garcia, D. Callaway, K. Poolla, and P. Varaiya, "Real-time scheduling of deferrable electric loads," in *Proc. of the American Control Conference*, 2012.
[11] L. Gan, U. Topcu, and S. H. Low, "Optimal decentralized protocol for electric vehicle charging," *IEEE Transactions on Power Sytems*, vol. 28, no. 2, pp. 940–951, May 2013.
[12] E. Sortomme, M. M. Hindi, S. D. J. MacPherson, and S. S. Venkata, "Coordinated charging of plug-in hybrid electric vehicles to minimize distribution system losses," *IEEE Transactions on Smart Grid*, vol. 2, no. 1, pp. 198–205, Mar. 2011.
[13] K. Clement, E. Haesen, and J. Driesen, "Coordinated charging of multiple plug-in hybrid electric vehicles in residential distribution grids," in *Proc. of the Power Systems Conference and Exposition*, 2009.
[14] M. Penn and M. Tennenholtz, "Constrained multi-object auctions and b-matching," *Inform. Process. Letters*, vol. 75, no. 1, pp. 29–34, 2000.
[15] M. Armstrong, "Optimal multi-object auctions," *The Review of Economic Studies*, vol. 67, no. 3, pp. 455–481, 2000.
[16] V. Krishna, *Auction theory*. Academic press, 2009.
[17] D. Materassi, S. Bolognani, M. Roozbehani, and M. A. Dahleh, "Deferrable loads in an energy market: coordination under congestion constraints (extended version)," 2014. [Online]. Available: http://web.mit.edu/saverio/www/papers/MED14.pdf
[18] M. Pinedo, *Scheduling: theory, algorithms, and systems*. Springer, 2012.
[19] J. Chen, F. N. Lee, A. M. Breipohl, and R. Adapa, "Scheduling direct load control to minimize system opoperation cost," *IEEE Transactions on Power Systems*, vol. 10, no. 4, pp. 1994–2001, Nov. 1995.
[20] K.-H. Ng and G. B. Sheble, "Direct load control - a profit-based load management using linear programming," *IEEE Transactions on Power Systems*, vol. 13, no. 2, pp. 688–695, May 1998.
[21] R. Brèuggemann and G. P. Patil, *Ranking and Prioritization for Multi-indicator Systems: Introduction to Partial Order Applications*. Springer, 2011, vol. 5.
[22] S. Chen, L. Tong, and T. He, "Optimal deadline scheduling with commitment," in *Proc. of the 49h Annual Allerton Conf. on Communication Control and Computing*, 2011.