Real-Time Scheduling for Peak Load Reduction in a Large Set of HVAC Loads

Marco L. Della Vedova and Tullio Facchinetti
Dpt. of Industrial and Information Engineering
University of Pavia, Italy
E-mail: marco.dellavedova@unipv.it, tullio.facchinetti@unipv.it

***Abstract*—This paper presents a technique to predictably coordinate the activation of Heating, Ventilation and Air Conditioning systems (HVACs) in order to limit the overall peak load of power consumption (peak shaving). The proposed solution represents a viable approach to the Demand-Side Management in the context of a smart grid for this type of loads. The coordination method performs a load shifting based on the discipline of real-time scheduling traditionally studied in the field of computing systems. With this approach, individual constraints on the temperature associated with the activation of each HVAC can be satisfied. The main advantage of the proposed technique is its low computational complexity, which allows to manage large sets of loads. A specific approach is proposed and evaluated to deal with large sets of loads by properly partitioning the load set into subsets (scheduling groups) that are scheduled independently from each other. Simulation results based on realistic parameters show that the peak load can be reduced by 35% in normal working conditions, and up to 60% with respect to worst case situations, without affecting the comfort achieved by each HVAC.**

Keywords- Real-time systems; Power system control; Scheduling; Demand-Side Management; Load shifting.

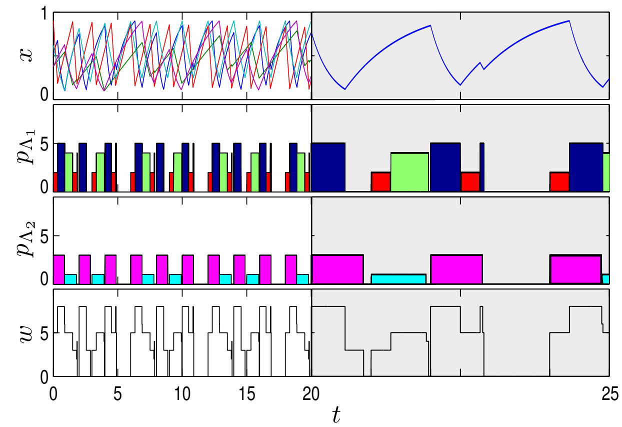

Fig. 1. Example of schedule of several HVACs. The time-scale is enlarged from $t = 20$ to $t = 25$ to better depict the system evolution. In the top chart shows the evolutions of temperatures normalized in their working ranges. The second and third top charts show the consumed power by two independently scheduled groups of HVACs. Active loads are indicated by their color. The bottom chart shows the overall power demand.

I. Introduction

The Demand-Side Management (DSM) is one of the key features enabled by the implementation of a smart power grid. The DSM is commonly implemented to improve energy system performance and reliability by limiting the peak load [1].

This paper proposes a technique to achieve predictable Demand-Side Management actions targeted to the reduction of the peak load of power consumption in a smart energy environment. Our approach is based on load shifting to avoid unnecessary simultaneous activations of a set of HVACs to reduce the peak load that is caused by the absence of coordination. A typical goal of a HVAC system is to keep the room temperature within the desired range. Therefore, heating or cooling is provided depending on the actual room temperature, which is affected by the temperature of the external environment. For a given external temperature the activation pattern of a HVAC can be suitably approximated by a periodic activity [2].

The modeling and control approach is derived from the domain of Real-Time scheduling studied in computing systems [3]. The timing behavior of load activations is modeled using parameters traditionally adopted by real-time processing systems, such as periods and activation times. The distinguishing features of our approach are: i) to provide guarantees in worst case conditions; ii) to achieve the user constraints on the temperature controlled by each HVAC; iii) to address the scalability issue in order to deal with large sets of loads.

This paper addresses the load management in worst case conditions. While it can be pessimistic in the average scenario, it allows a predictable load balancing to limit the impact of low-probability (but still possible) conditions which are behind potentially catastrophic consequences such as power disruption due to overloads [4]. The performance of real-time scheduling applied to electric loads in average conditions is studied in [5], where real-time load scheduling was compared with the case of absence of load control.

The control action is based on the specification of the requirements that need to be enforced on the temperature, i.e., the desired working range for each HVAC. The control policy will meet such requirements, while limiting the peak load. To model and meet the requirements on the desired working conditions, an approach based on the existing real-time scheduling theory has been used, namely Real-Time Physical Systems (RTPS) [6]. RTPSs are a special case of switching hybrid system where the switching signal corresponds to a real-time schedule. A physical variable (the controlled temperature, in our case) changes its value according to the activation the HVAC, and such activations are driven by a real-time scheduler. In practice, the dynamics of the physical process changes when its state is switched due to scheduling decisions. Moreover, the activity of a load may be interrupted and resumed later to account for higher priority loads. The priority of a load is automatically determined by the scheduling algorithm on the basis of the timing constraints.

The focus of the proposed approach is on the scalability to large systems. For this purpose, the discussed method has low computational complexity, making it applicable to large sets of loads. The idea is to organize the loads into distinct groups that are scheduled independently from each others. This commonly adopted approach is known in the field multiprocessor real-time computing systems as *partitioned scheduling*. Despite it may produce a sub-optimal solution, the partitioned scheduling reduces the complexity to a linear or polynomial problem, and it allows the use of simpler scheduling algorithms having a number of known useful properties.

Fig. 1 shows an example of system behavior where different on/off HVACs are partitioned into two independent groups and scheduled to limit the overall peak load. The figure also shows the behavior of the temperature associated with each load, which depends on the activation state of the load.

The reminder of the paper is organized as follow. In Section II is presented a brief analysis of the state of the art in DSM literature. Section III describes the system model in both dynamics and timing terms. Section IV states the properties of the system and shows how is possible to get adequate timing parameters for the switching signal. Section V tackles the problem of the large set of loads: a partitioning method is proposed to achieve both good results in peak load reduction and small complexity in algorithms that allows the method to scale well with the number of loads. Section VI quantifies the performances of the proposed approach with simulated examples. Final remarks are in Section VII.

II. Related works

The literature on power systems addressing DSM approaches is wide. A recent overview and categorization of DSM approaches is available in [1]. Some works focuses on the modeling aspects, without proposing a control method based on those models [7]. Optimization methods are often leveraged to minimize the peak load. However, the adopted solutions are based on off-line algorithms that can not cope with the dynamic and heterogeneous nature of large systems [8], [9]. Approaches based on artificial intelligence are also available, such as [10], where fuzzy logic is used to control a set of thermal loads. However, the properties of these methods (and predictability in particular) are not formally proved. In [11] the authors present a priority-based approach to the load management. It clearly shows different types of loads and their working constraints. The weak point of the approach is the manual assignment of priorities. Our approach, instead, while being inherently based on the assignment of priorities, provides an automatic assignment of priorities on the basis of timing parameters. In fact, the scheduler dynamically activates the load having the highest priority. This is a distinguishing feature of real-time scheduling methods.

RTPSs have been firstly presented in [6], dealing with affine dynamic systems. In [6] a partitioning-based method to manage large load sets is mentioned as a possibility and not explicitly integrated in the model. Before the formalization of RTPSs, the use of real-time scheduling for the management of electric loads was proposed in [12] and [13]. In [12] the focus is on the optimal partitioning of loads to manage large sets of loads. However, no physical variables are associated with the loads. In [13] such association is addressed, but the issue of large systems is not considered. Moreover, in [13] state variables are characterized by integrator dynamics, while affine dynamics are considered in this paper. The application of real-time techniques to the load scheduling is investigated in [14] considering constraints on state variable variations and modeling errors. Errors are characterized by a statistical distribution, and they are compensated using a feedback technique based on the measurement of the state variable value in correspondence to request times.

III. System model

This section describes the system model, including the adopted model for HVACs and its modeling terms of real-time parameters.

A. Physical system

The power system considered in this paper is modeled as a set of n electric on/off HVACs. HVACs are independent from each others. A simple but accurate model for a HVAC system is proposed in [9], and it is recalled in the following. The adopted model describes a first order dynamic system, which has been proven to capture the behavior of HVAC loads accurately.

$$\frac{\mathrm{d}X(t)}{\mathrm{d}t} = \frac{X_o(t) - X(t) - X_g s(t)}{\tau} \tag{1}$$

In (1), $X(t)$ is the internal air temperature of the room, $X_o(t)$ is the outside air temperature, X_g is the temperature gain of the air-conditioner, τ is the actual time constant of the room expressed in minutes, and $s(t) : \mathbb{R} \to \mathbb{B}$ is the current state of activation of the load: $s(t) = 0$ if the load is not active at time t and $s(t) = 1$ otherwise. The above model characterizes the behavior of a single HVAC. The controller must guarantee that the inside room temperature lays within a comfort range:

$$X(t) \in \left[X^{\min}, X^{\max}\right], \quad \forall t \tag{2}$$

In [15] this type of loads is called *controllable load*, since it can be shed to achieve the peak load reduction. Preferably, in this specific case a feedback control scheme should be integrated to regulate the temperature upon external temperature variations. A suitable method for this control is proposed in [16].

B. Real-time modeling of the switching signal

Load activations are driven by a switching signal. Such a signal is generated by a centralized controller, called *scheduler*. Considering the entire system, the *schedule* is defined as the function $s : \mathbb{R}^+ \to \mathbb{B}^n = [s_1 \, \ldots \, s_n]$. The distinguishing point of the proposed approach is that the switching signal is generated by a real-time scheduling algorithm, such as the Earliest Deadline First algorithm (EDF) [3].

The total power demand in any given time instant t is the sum of the power consumed by all active loads at time t. On the other hand, the scheduling action performed by the scheduler produces the effect to reduce the unnecessary simultaneous activations of loads. As a result, the scheduling action is able to limit the peak load by avoiding unnecessary simultaneous activations.

Considering the above observations, the modeling and control problem translates to the assignment of proper values to timing parameters and constraints associated with the HVACs. For this purpose, a set of real-time parameters are associated to each HVAC. The adopted model derives from the periodic task model studied in real-time computing systems [17]. The generic i-th load is associated with the tuple $\lambda_i \doteq (T_i, C_i, P_i)$, where:

- $T_i \in \mathbb{R}^+$: it is the minimum time interval between two consecutive request times, or *period*; a *request time* $r_{i,k}$ is defined as the k-th request for activating the load; it holds $r_{i,k+1} - r_{i,k} = T_i$, $k \in \mathbb{N}$;
- $C_i \in \mathbb{R}^+ : C_i \leq T_i$ represents the required duration of the load activation time within each period T_i;
- $P_i \in \mathbb{R}^+$ is the nominal power of the i-th load.

Real-time parameters are used by the scheduling algorithm to generate the switching signal, i.e., the schedule. The values of timing parameters will be bounded to enable the schedulability analysis in the worst case. For this purpose, the following definition is introduced:

Definition 1 (Valid schedule)**.** *A schedule s is said to be* valid *if it assigns to each load an amount of activity time equal to C_i within each time interval $[r_{i,k}, r_{i,k+1}]$. Formally, it holds:*

$$\forall i, \forall k \quad \int_{r_{i,k}}^{r_{i,k+1}} s_i(t)\,\mathrm{d}t = C_i \tag{3}$$

Note that the definition of valid schedule is slightly different from the one applicable to traditional real-time systems. In particular, in traditional real-time systems a less-then-equal relation is allowed, since C_i refers to the Worst-Case Execution Time (WCET). The WCET is the longest possible execution time of a real-time computing task. The WCET is used to perform the schedulability analysis in the worst case, while during the system behavior the actual duration of a task can be less than WCET. An equality is instead formulated in (3). This is required to achieve the requirements on the state variable variation.

To derive the results in following sections, two common figures used in real-time systems are introduced: the load utilization $U_i \doteq C_i/T_i$ and the total utilization $U^{\text{tot}} \doteq \sum_{i=1}^{n} U_i$. While $U_i \leq 1$ is the fraction of time in which the i-th load is active, U^{tot} represents the total fraction of activity time of the whole load set. The total utilization U^{tot} is particularly useful. In fact, it is used to perform a test (the so-called *schedulability test*) to determine whether a given scheduling algorithm can successfully schedule the load set [3]. Under proper assumptions, a scheduling algorithm $\mathcal{A}$ is able to schedule a load set if $U^{\text{tot}} \leq U^{\text{lub}}(\mathcal{A})$, where $U^{\text{lub}}(\mathcal{A})$ is the least upper bound on the total utilization that guarantees the schedulability when using the $\mathcal{A}$ scheduling algorithm.

C. The feasibility problem

As from previous definitions, the considered system model is composed by a dynamic system, the desired working range, the real-time parameters and a scheduling algorithm. While the dynamic system and the working range are related with the underlying physical process, i.e., the controlled HVAC, real-time parameters and the scheduling algorithm can be selected by the system designer. The selection should be made in order to obtain a feasible RTPS, according with the following definition:

Definition 2 (Feasibility)**.** *Given the timing parameters describing the load set, a RTPS is said to be* feasible *if and only if user requirements are satisfied by every valid schedule.*

Equation (3) characterizes a class of switching signals within the set of all possible scheduling patterns. The RTPS *feasibility problem* concerns the identification of the class of valid switching signals such that the requirements on the controlled temperature are guaranteed. This problem translates to the identification of suitable values for C_i and T_i to drive the evolution of physical variables in compliance with user requirements.

The analysis is based on the observation that the scheduler generates a valid switching signal among all the possible valid signals. Therefore, the analysis is performed considering the worst case signal, i.e., the signal that brings to the worst possible situation in terms of user requirements violation. This allows to assess the behavior of all other “less critical” valid switching signals.

D. Peak load minimization

The application of RTPSs proposed in this paper is to limit the peak load of power demanded from a set of HVACs, while meeting requirements on the controlled temperature. The activity of an HVAC is controlled by the scheduler, which generates the schedule s_i for the i-th load. The i-th HVAC consumes either an amount of electric power $P_i \in \mathbb{R}^+$ when active, no power otherwise. Hence, the power consumption over time can be modeled with the function $p : \mathbb{R}^+ \to \mathbb{R}^+$ defined in (4).

$$p_i(t) = P_i s_i(t) \tag{4}$$

Transient phases between active and inactive states are not considered in this paper.

The overall instantaneous electric power absorbed at time t is the sum of the power consumed by all the HVACs, as stated in (5).

$$w(t) = \sum_{i=1}^{n} p_i(t). \tag{5}$$

The *peak load* is the maximum value taken by $w(t)$ during the considered timespan.

The peak load minimization can be optimally achieved by a RTPS scheduler when a uni-processor scheduling algorithm, such as EDF, is able to schedule the load set. In this case, the algorithm achieves that only one load is active at any given time, and the peak load is equal to the nominal power of the most power-consuming load. The schedulability test can be used to determine whether there exists a feasible schedule, provided that user requirements are also met.

On the other hand, if simultaneous activations can not be avoided, i.e. when a uni-processor scheduling algorithm is not able to schedule the load set, then the minimization of the peak load becomes more complex. In this case, a RTPS scheduler generates a schedule that approximates the optimal solution. Therefore, the RTPS method represents an efficient heuristic to this problem. In fact, a scheduling algorithm as EDF has complexity $O(n \log n)$ required to sort a queue upon a load activation request.

IV. From physical to timing parameters

In [6] it is shown by a worst-case analysis how to derive the required timing parameters, namely the period T and the utilization U, from a first order model of a load like the one expressed by (1)-(2). The external temperature is represented by a constant function $X_o(t) = X_o$. Basically, there exists a set of pair (U, T) for which the temperature is always kept within the comfort range for any possible activation pattern generated by the real-time scheduler using a given value for the (U, T) parameters. In particular, it is shown that U must be chosen within the range $\left[U^{\min}, U^{\max}\right]$, where:

$$\begin{aligned} U^{\min} &= \max\left\{0, \frac{X_g + X^{\min} - X_o}{X_g}\right\} \\ U^{\max} &= \min\left\{\frac{X_g + X^{\max} - X_o}{X_g}, 1\right\} \end{aligned} \tag{6}$$

It is easy to show that a valid choice is $U = \frac{U^{\min}+U^{\max}}{2}$. Once the utilization U has be determined, the period T must be chosen such that both inequalities (7) hold. Following this procedure, it is possible to assign timing parameters to the electric load such that the temperature is always kept within the comfort range. In (7), $\overline{U} = 1 - U$.

$$\begin{aligned} X_o - X^{\min} &> \frac{2X_o e^{UT/\tau} - X_g e^{2UT/\tau} + X_g e^{T(1+U)/\tau}}{1 - e^{T/\tau}} \\ X_g - X_o + X^{\max} &> \frac{2X_o e^{\overline{U}T/\tau} - X_g e^{2\overline{U}T/\tau} + X_g e^{T(2-\overline{U})/\tau}}{1 - e^{T/\tau}} \end{aligned} \tag{7}$$

The set of possible solutions for both (6) and (7) determines the region Ω of pairs in the $U - T$ space, called *feasibility region*. An example of feasibility region is depicted in Fig. 2. As it will be clearer from Section V, the choice of a low value for U helps to obtain a lower peak load. However, lower U brings to lower T, thus generating a higher switching frequency. This is often not tolerable in practical applications such as processes driven by electric motors. Therefore, the selection of U represents a trade-off between peak load and system lifetime.

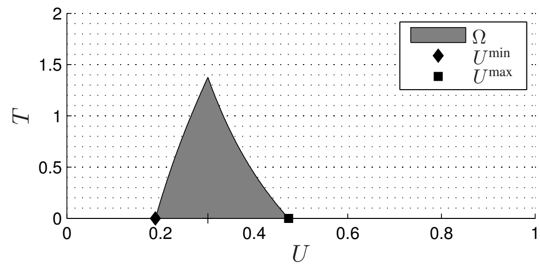

Fig. 2. Example of feasibility region in the $U - T$ space. By scheduling a load with timing parameters in the feasibility region, the achievement of user requirements is guaranteed (see Definition 2).

V. Load partitioning

This paper proposes the use of classic real-time scheduling algorithms to manage the set of electric loads, such as Rate Monotonic (RM) or Earliest Deadline First (EDF) [17]. The scheduling algorithm requires the specification of T_i and C_i for every load λ_i to build a schedule. Well known real-time scheduling algorithms such as RM and EDF can generate a schedule where only one single load is active at any given time. However, this is possible only when the total utilization U^{tot} is less or equal of an upper bound $U^{\text{lub}}(\mathcal{A})$ whose value depends on the considered algorithm $\mathcal{A}$. For example, $U^{\text{lub}}(EDF) = 1$. Therefore, when $U^{\text{tot}} \leq 1$, preemptive EDF can build a schedule $\mathcal{S}$ without activating more than one load at any given time. As a consequence, the peak load $P^* = \max_i P_i$ is minimized.

On the other hand, if $U^{\text{tot}} > U^{\text{lub}}$ then the simultaneous activation of two or more loads can not be avoided, leading to a possibly larger peak power consumption $P > P^*$. The proposed solution is to partition the whole load set into m disjoint sets Λ_j, $j = 1, \ldots, m$, called *scheduling groups*. Scheduling groups are determined such that their total utilization, defined as

$$U_{\Lambda_j} = \sum_{\lambda_i \in \Lambda_j} U_i, \tag{8}$$

is smaller than or equal to $U^{\text{lub}}(\mathcal{A})$. This property enables an uni-processor scheduling algorithm $\mathcal{A}$ to find a valid schedule independently for each scheduling group.

Since there is no relationship between the schedule generated within any pair of scheduling group, the maximum overall peak load will happen when the loads with the highest power are simultaneously activated in every scheduling group. Therefore, an upper bound P^{ub} on the peak load can be found considering the simultaneous activation within every group of the load with the highest power P_i, i.e.:

$$P^{\text{ub}} = \sum_{\Lambda_j} \max_{\lambda_i \in \Lambda_j} P_i. \tag{9}$$

A. Level packing

The problem of partitioning the set of loads can be formalized as a *level packing* problem [18]. Level packing is a special case of the generic two-dimensional packing problem. In level packing, one or more strips are filled to accommodate a set of rectangles such that the total height is minimized. The peculiarity of level packing is that rectangles are partitioned in horizontal strips or levels. The complexity of the level packing problem is NP-hard; in fact, it can be easily reduced to a classical one-dimensional packing problem having NP-hard complexity. Approximation methods have been proposed to face the complexity issue [19]. The approximation methods

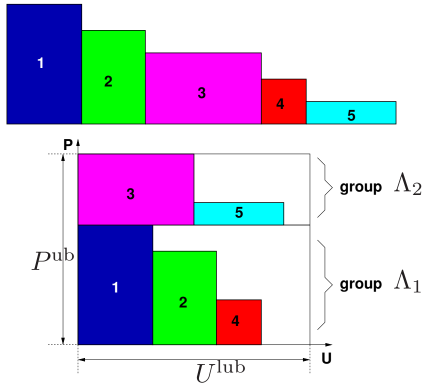

Fig. 3. Example of level packing using the FFDH algorithm. Five items are firstly ordered by non-increasing height and then packed into two levels. Note that item 3 generates a new level Λ_2 since it does not fit at the right of item 2 within level Λ_1. The load utilizations of the 5 loads are respectively $(0.28, 0.26, 0.49, 0.22, 0.30)$. Their consumed power are $(5, 4, 3, 2, 1)$. The schedule generated by this system is depicted in Fig. 1.

are built by ordering the rectangles by non-increasing heights. Rectangles are grouped to fill the strips using different strategies. In each level, items are packed from left to right by non-increasing height, similarly to the arrangement of books within a bookshelf (see Fig. 3). The *First-Fit Decreasing Height* scheme (FFDH) is conceived such that it inserts the next item X (in non-increasing height ordering) on the first level where X fits. If no level can accommodate X, a new level is created. After the packing action, the height of a level is equal to the height of the leftmost item. The interesting aspect of FFDH is that the time complexity is $O(n \log n)$. Moreover, its approximation ratio has been formally derived. In particular, it holds $\text{FFDH}(I) \leq (17/10) \cdot \text{OPT}(I) + 1$, where I is a set of items to be packed, $\text{FFDH}(I)$ is the height obtained by FFDH, and $\text{OPT}(I)$ is the height produced by the optimal algorithm. The asymptotic bound of 1.7 is proved to be tight.

B. Application to the scheduling problem

The application of the level packing to the scheduling of electric loads requires the proper modeling of the loads. Therefore, each load λ_i is represented as a rectangle having height equal to the power consumption P_i and width equal to its utilization U_i, being $U_i \leq 1$. The packing happens in a two-dimensional space where the utilization appears on the x axis, while the consumed power is on the y axis. The width of the packing space corresponds to the least upper bound on the utilization of the considered scheduling algorithm (e.g., $U^{\text{lub}} = 1$ for EDF). The goal to limit the total height of packed rectangles clearly corresponds to the goal of limiting the peak load of power consumption of the whole power system. On the other hand, fitting the items on the x axis in each level corresponds to group a set of loads whose total utilization is less than or equal to U^{lub}, thus composing a set of loads that is successfully schedulable by the considered real-time scheduling algorithm. Once all loads have been grouped into scheduling groups using the level packing, each group of loads is scheduled independently from other groups. The schedulability is guaranteed since the utilization U_{Λ_j} of the group Λ_j is less than or equal to the upper bound U^{lub} that achieves the schedulability of the load set. Fig. 1 shows an example of schedule of loads whose relevant parameters are listed in Fig. 3. The grouping of loads is as in Fig. 3.

The proposed technique recalls the *Rate Monotonic First-Fit Decreasing Utilization* (RM-FFDU) partitioning scheme for scheduling fixed priority real-time tasks on a multi-processor system [20], where bin-packing techniques are used to allocate tasks to processors. However, [20] does not address the optimization of the total power consumption. Moreover, the key distinction is that in this paper the ordering is made with respect to the value of load's consumed power, and utilization is not considered for this purpose.

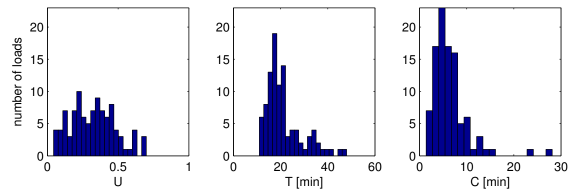

Fig. 4. Distribution of timing parameters values obtained from physical parameters in the 100-loads Monte Carlo simulation.

VI. Performance evaluation

This section assesses the performance of the proposed approach by means of simulation and using realistic parameters. Similarly to [9], from the viewpoint of generating different realistic operating scenarios, Monte Carlo simulations are applied in this paper. By assuming uncertainties on different variables (τ, X, X_o, X_g) that closely resemble real-life operating conditions, Monte Carlo simulations are performed through repeated sampling of uncertain variables.

Each simulation run has been initialized with pre-specified stochastic parameters, chosen with the following method: number of air conditioner – 100; internal temperature distribution – Normal distribution with mean 72 F and standard deviation 12 F, i.e. $X(0) \in N(74, 12)$ [F]; air-conditioner model parameters – $\tau \in N(64, 5)$ [min], $X_g \in N(30, 10)$ [F]; outside air temperature – uniform distribution between 75 F and 90 F $X_o \in [75, 90]$ [F]; desired temperature range – $X^{\min} = 70$ [F], $X^{\max} = 76$ [F]. In terms of power request, 5 air-conditioner sizes are considered – $P \in \{1.2, 2.5, 3.0, 4.5, 6.0\}$ [kW].

Given the load set, the timing parameters (U, T, C) are calculated for each load as illustrated in previous sections of this paper. In particular, it is set $U = \frac{U^{\max} + U^{\min}}{2}$. Then, T is set as the maximum value within the set of feasible values (Definition 2). Fig. 4 shows the resulting histograms of timing parameters. Utilizations range from 0, i.e. always off, to 1, i.e. always on. Periods are in the range $[10, 60]$ minutes.

In order to evaluate the performance of the proposed approach, the RTPS control method is compared with the traditional hysteresis control. In the hysteresis-control approach, each air-conditioner is turned on when the room inside air temperature reaches the upper thermostat set-point $X^{\max}$ and turned off whenever this temperature falls below the lower thermostat set-point $X^{\min}$.

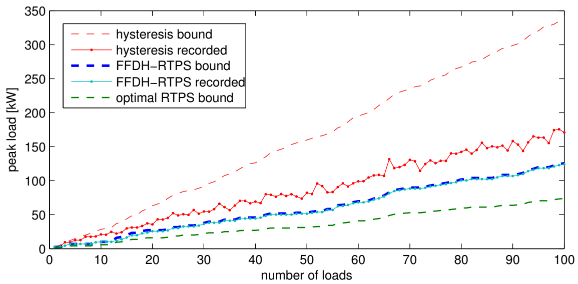

Fig. 5. Comparison between hysteresis- and RTPS-control methods. The RTPS-controlled actual behavior, during the simulation of 120 hours, reduces the peak load of the 35% in respect of hysteresis control in regular working conditions. The improvement on the theoretical bound associated with worst case working conditions is around 60%.

Fig. 5 shows the peak load as a function of the total number of loads in the system under different control strategies. With the hysteresis-based control, the worst case bound is equal to the sum of all loads power, while in the RTPS-controlled system the theoretical bound is obtained by summing the power consumed by the most power-consuming loads in each scheduling group generated by the packing algorithm. It is worth to note that, while the theoretical bound for the hysteresis-controlled method may represent a very unlikely worst-case condition, it is still a possible situation, whose likelihood increases with the system lifetime. For both control methods, the figure also shows the actual peak load recorded by simulating the system behavior over a 120 hours (5 days) time span. The actual (recorded) peak load generated by the RTPS-based control is able to reduce the peak load in average by 35% with respect to the actual (recorded) peak load in absence of coordination, which represents a normal working condition. On the other hand, the peak load is reduced by up to 60% with respect to the worst possible case (theoretical bound) of the hysteresis-controlled case. Finally, it is worth to note that the recorded peak load of the RTPS-controlled system is very close to its theoretical bound This means that the worst situation in the partitioned scheme (i.e., when the most power-consuming loads in every scheduling group are simultaneously activated) do happen almost always.

VII. Conclusion

This paper presented an approach to coordinate the activation of a large sets of HVACs in a Demand-Side Management scenario. The proposed method is based on scheduling techniques adapted from the domain of real-time scheduling. The combination of a level packing strategy and uni-processor scheduling algorithms allows to meet both timing and physical constraints. Simulation results based on realistic parameters prove a relevant improvement of peak load reduction.

References

[1] P. Palensky and D. Dietrich, "Demand side management: Demand response, intelligent energy systems, and smart loads," IEEE Transactions on Industrial Informatics, vol. 7, no. 3, pp. 381–388, aug. 2011.
[2] G. Koutitas and L. Tassiulas, "A delay based optimization scheme for peak load reduction in the smart grid," in Proceedings of the 3rd International Conference on Future Energy Systems: Where Energy, Computing and Communication Meet. ACM, 2012, p. 7.
[3] G. Buttazzo, Hard real-time computing systems: predictable scheduling algorithms and applications, ser. Real-Time Systems Series. Springer, 2005.
[4] D. Novosel, M. M. Begovic, and V. Madani, "Shedding light on blackouts," Power and Energy Magazine, IEEE, vol. 2, no. 1, pp. 32 – 43, jan-feb 2004.
[5] M. L. Della Vedova, E. Di Palma, and T. Facchinetti, "Electric loads as real-time tasks: an application of real-time physical systems," in 7th International Wireless Communications and Mobile Computing Conference (IWCMC), July 2011, pp. 1117–1123.
[6] M. Della Vedova, M. Ruggeri, and T. Facchinetti, "On real-time physical systems," in Proceedings of the 18th International Conference on Real-Time and Network Systems (RTNS), November 2010, pp. 41–49.
[7] A. Molina-Garcia, M. Kessler, J. Fuentes, and E. Gomez-Lazaro, "Probabilistic characterization of thermostatically controlled loads to model the impact of demand response programs," Power Systems, IEEE Transactions on, vol. 26, no. 1, pp. 241–251, Feb. 2011.
[8] T.-F. Lee, M.-Y. Cho, Y.-C. Hsiao, P.-J. Chao, and F.-M. Fang, "Optimization and implementation of a load control scheduler using relaxed dynamic programming for large air conditioner loads," Power Systems, IEEE Transactions on, vol. 23, no. 2, pp. 691 –702, may 2008.
[9] B. Ramanathan and V. Vittal, "A framework for evaluation of advanced direct load control with minimum disruption," IEEE Transactions on Power Systems, vol. 23, no. 4, pp. 1681–1688, November 2008.
[10] C.-M. Chu, T.-L. Jong, and Y.-W. Huang, "A direct load control of air-conditioning loads with thermal comfort control," in IEEE Power Engineering Society General Meeting, June 2005, pp. 664–669 Vol. 1.
[11] M. Pipattanasomporn, M. Kuzlu, and S. Rahman, "An algorithm for intelligent home energy management and demand response analysis," Smart Grid, IEEE Transactions on, vol. PP, no. 99, p. 1, 2012.
[12] T. Facchinetti, E. Bini, and M. Bertogna, "Reducing the peak power through real-time scheduling techniques in cyber-physical energy systems," in Proceedings of the First International Workshop on Energy Aware Design and Analysis of Cyber Physical Systems (WEA-CPS), April 2010, pp. 18–25.
[13] T. Facchinetti and M. L. Della Vedova, "Real-time modeling and control of a cyber-physical energy system," in Proceedings of the First International Workshop on Energy Aware Design and Analysis of Cyber Physical Systems (WEA-CPS), April 2010, pp. 10–17.
[14] ——, "Real-time modeling for direct load control in cyber-physical power systems," IEEE Transactions on Industrial Informatics, special issue on Information Technology in Automation, vol. 7, no. 4, pp. 689–698, November 2011.
[15] S. Shao, M. Pipattanasomporn, and S. Rahman, "Development of physical-based demand response-enabled residential load models," Power Systems, IEEE Transactions on, vol. PP, no. 99, p. 1, 2012.
[16] M. L. Della Vedova and T. Facchinetti, "Feedback scheduling of real-time physical systems with integrator dynamics," in Proceedings of 17th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), September 2012.
[17] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard real-time environment," Journal of the Association for Computing Machinery, vol. 20, no. 1, pp. 46–61, January 1973.
[18] A. Lodi, S. Martello, and D. Vigo, "Models and bounds for two-dimensional level packing problems," Journal of Combinatorial Optimization, vol. 8, no. 3, pp. 363–379, 2004.
[19] A. Lodi, S. Martello, and M. Monaci, "Two-dimensional packing problems: A survey," European Journal of Operational Research, vol. 141, no. 2, pp. 241 – 252, 2002.
[20] Y. Oh and S. Son, "Allocating fixed-priority periodic tasks on multiprocessor systems," Real-Time Systems, vol. 9, no. 3, pp. 207–239, 1995.