The Many Faces of Real-Time Scheduling Applied to Power Load Management

Tullio Facchinetti
Dipartimento di Ingegneria Industriale e dell'Informazione
University of Pavia, Italy
e-mail: tullio.facchinetti@unipv.it

***Abstract*—In the field of power systems there is an active research on methods to limit the peak load of power demand. Recently, a new approach to the automatic management of electric loads has been proposed, which is based on scheduling techniques derived from real-time computing systems. The research on real-time scheduling encompasses several aspects, including scheduling algorithms and analysis techniques, that may suitably be adapted to the management of sets of power loads. This paper overviews background concepts on real-time scheduling, and fosters their possible application to the power load management, with special focus on peak load reduction. Several issues regarding the power load management and possible solutions based on real-time scheduling are considered. Due to the wide scope of the proposed methodologies, only the indication of their applicability is provided in this paper. For more complex scenarios, references to previous related works are provided when available.**

Keywords-Modeling; Real-time systems; Power system control; Scheduling; Smart Grid; Demand-Side Management; Direct Load Control; Load Shedding.

I. Introduction

The Smart Grid is the emerging technology in the field of electric power management, generation, distribution and usage. Its architecture combines the power distribution infrastructure with a digital communication network. Load balancing is a key challenge addressed by the Smart Grid [1]. Peak load conditions are generated by the simultaneous request of electricity by many users. Such situations may bring to severe consequences, arising technical and economic issues for both suppliers and users [2]. Therefore, an appropriate load management targeted to obtain predictable load conditions would lead to remarkable benefits.

This paper describes the potentials of a new methodology based on real-time scheduling to manage the predictable activation/shedding of power loads. The background idea is to establish an analogy between real-time computing systems and power systems, in order to use real-time modeling, scheduling and analysis methodologies to coordinate the activation of a set of loads. The goal is to delineate the characteristics of a general framework where different types of loads, physical constraints, system dynamics and objectives can be merged to allow system-wide real-time optimization of the peak load. For this purpose, electric loads are described using timing parameters derived from the real-time scheduling research domain. In this way, existing

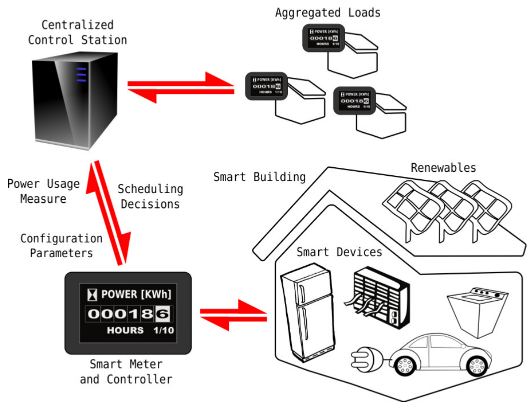

Figure 1. Overview of the smart distributed coordination infrastructure enabling the advanced control features based on real-time scheduling envisioned in the paper.

well-known real-time scheduling algorithms can be applied to predictably activate/shed each load, while guaranteeing given constraints on the underlying physical process associated with the electric load.

There are several relevant motivations to investigate the application of real-time scheduling techniques to power systems. Some of the most important benefits include:

- the predictable behavior of real-time systems, that can be a-priori guaranteed in a mathematically strong form;
- the availability of several modeling and control methodologies that can be adapted and applied to the load management;
- the automatic derivation of load priorities based upon the characteristics of both the load and the underlying physical process;
- the possibility to manage large sets of loads, thanks to powerful scheduling policies with low computational complexity;
- the availability of efficient heuristics and optimization-based methods to deal with more complex cases.

The aim of this paper is to shed the light on the chances offered by the application of real-time scheduling techniques to the coordination of power loads. In particular, the paper illustrates and categorizes the available approaches in the field of real-time systems that can be suitably applied to the management of electric loads. Therefore, each technique is briefly described and its possible application to the management of power loads is suggested.

Fig. 1 shows the infrastructure suitable for the implementation of the proposed management system. Scheduling decisions are taken at the centralized control station and dispatched to the smart metering and control devices installed within the smart building. Such decisions trigger the activation/shedding of smart devices within the building. During the configuration, the smart meter collects configuration parameters from the controlled devices, and send them to the control station. At run-time, it measures the power usage and collect other interesting values and provides the information to the control station. In this scenario, loads belonging to different buildings can be coordinated in an aggregated manner to boost the balancing effect.

The paper is organized as follows. Section II provides a short introduction to basic concepts related with real-time scheduling. A generalization of the applicability of real-time scheduling techniques is presented in Section III. Section IV discusses scheduling-based works related with smart power management. Characteristics and features of real-time scheduling techniques suitable to electric loads management are described in Section V. Finally, Section VI concludes the paper.

II. Notions of real-time scheduling

Real-Time Systems (RTSs) are studied in the domain of computing systems to allow the timely execution of concurrent processing tasks on one or more processors [3]. The peculiarity of a RTS is its strong relationship with the physical system under monitoring and control. Moreover, a distinguishing feature is the analysis of system properties typically based on the evaluation of worst case conditions, in order to achieve the desired behavior in every possible situation. This fact makes RTSs especially suitable for critical applications, such as in automotive systems, avionics, factory automation, and process control (e.g., chemical or nuclear plants).

In the traditional system model a set of computing tasks are required to be executed on one processor. Since only one task can use the processor at any given time, a scheduling algorithm assigns the use of the processor to the task having the highest priority. Differently from other approaches, the priority of tasks is not explicitly set by the system designer using some empirical assessment. Instead, it is automatically inferred by the scheduling algorithm based on values of timing parameters used to describe the task.

The simplest and most investigated system model in RTSs consists of periodically activated tasks. Each task τ_i becomes ready for the execution at the request time r_i, which is an integer multiple of the *period* T_i, i.e., the k-th instance of τ_i (called *job*) is released at time kT_i. The job duration is at most C_i time units. The parameter C_i is referred to as Worst-Case Computation Time (WCET). A relative deadline D_i is associated with the task τ_i. A deadline D_i means that the k-th job must complete no later than D_i time units after the release time, i.e., it must execute in the time frame $[kT_i, kT_i + D_i)$, where $d_i = kT_i + D_i$ is the absolute deadline of the k-th job. Often, relative deadlines are assumed equals to the period, i.e., a job must terminate before the next period. In this case, the task model is said having *implicit deadlines*. The scheduling algorithm controls the proper execution of each task within its period time frame. For instance, the Earliest Deadline First scheduling algorithm (EDF) assigns the highest priority to the task having the earliest absolute deadline. This model is suitable in several concrete applications. For instance, control applications require a periodic sensor sampling and actuator driving.

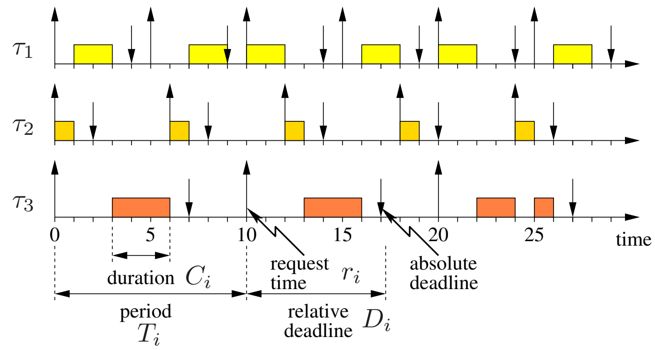

Figure 2. Example of concurrent scheduling of 3 tasks by EDF. Each task is described by the tuple (T_i, D_i, C_i). Task parameters are $\tau_1 = (5, 4, 2)$, $\tau_2 = (6, 2, 1)$ and $\tau_3 = (10, 7, 3)$.

The example in Fig. 2 shows the schedule generated for 3 tasks by the EDF algorithm. The task τ_3, e.g., have a period of $T_3 = 10$ time units and a computation time $C_3 = 3$. Each job must complete no later than $D_3 = 7$ time units after the release time. Notice that, as allowed in typical RTSs, the third job of τ_3 is interrupted at time $t = 24$ after 2 time units and resumed later for being completed. This interruption is called *preemption*, and it plays an important role to let all tasks to meet their deadline. Preemptions can represent an important issue when real-time techniques are applied to the management of electric loads, as will be discussed later.

The periodic task model has several attractive properties. A very interesting one is the possibility to define a figure called *utilization*, defined as $U = \sum_i C_i/T_i$. The utilization represents the load of a processor, being the percentage of computing time required by all tasks. A remarkable use of the utilization is for the so-called *schedulability test*. A schedulability test is a condition that, whether satisfied, guarantees that every task will meet its timing constraints, i.e., the execution of every job terminates before the deadline in every possible situation (i.e., considering worst case conditions). This test can be used to obtain *a-priori guarantees* on the scheduling process. For example, when the

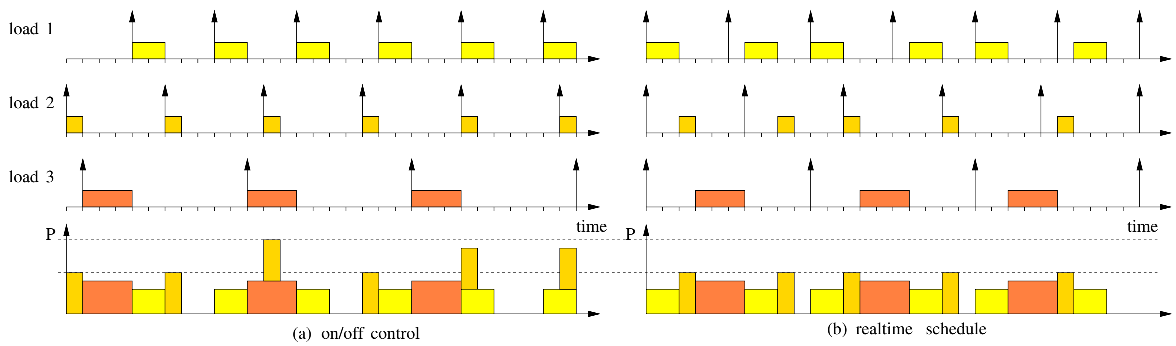

Figure 3. Comparison between (a) a traditional on/off control method and (b) a coordinated load activation. The picture shows the comparison between the achieved peak loads. The time-base is 5 minutes.

preemptive EDF scheduling algorithm is used, and implicit deadlines are considered, timing constraints are guaranteed iff $U \leq 1$. The schedulability test formulation depends essentially on the adopted system model, task model and scheduling algorithm.

III. Real-time scheduling from computing to power systems

The problem of using real-time scheduling methods for managing sets of power loads is related with the modeling of electric loads behaviors in terms of timing parameters such as periods and deadlines – as introduced in Section II. In practice, it is necessary to determine the value of timing parameters to obtain the desired behavior of the modeled load. The derivation of timing parameters enables the application of real-time scheduling algorithms for the automatic management of loads activation/shedding. Moreover, real-time analysis methods can be used to a-priori determine the expected system behavior and to assess its performance.

The periodic model described in Section II can be easily adapted to model power loads. For instance, a HVAC (Heating, Ventilation and Air Conditioning) may be periodically activated for 5 minutes (WCET) every 30 minutes (period) to maintain the desired temperature within a room. The measurements carried out in [4] confirm the periodic nature of such type of loads. Load 2 depicted in Fig. 3 is an example of a load having such timing parameters.

The schedule represented in Fig. 2 can thus be reconsidered from the load management viewpoint. The 3 depicted tasks may represent the corresponding number of air-conditioners whose activation is periodically scheduled. Each load is characterized by a specific power demand when active. Fig. 3.b shows this application. In the figure, the schedule is completed by the time-line representing the instantaneous power consumption. The scheduling action automatically eliminates the simultaneous activation occurring in case of absence of coordination. This allows to obtain a peak load that is equal to the power consumed by the most power-consuming load. The activation pattern generated by the real-time schedule is compared with the one produced by a classical on/off hysteresis controller (Fig. 3.a). The controller is used to regulate the variation of a physical quantity (e.g., the temperature) in order to limit its variation within a predefined range. It works by switching on the system when a range threshold is exceeded and keeping the system active until the lower threshold is reached. This control scheme leads to a behavior that can be approximated with a strictly periodic activation of the load, as in Fig. 3.a. The result is the absence of coordination among loads, leading to higher peak loads.

Clearly, this is a simplified example. For instance, the values of timing parameters need to be related to the physical variables of interest (internal/external temperatures, room insulation, open doors/windows, etc.). However, the goal of this paper is precisely to illustrate how such factors can be integrated in the model and accounted during the system functioning.

On the other hand, this approach is able to seamlessly manage large sets of heterogeneous loads. In fact, once a load is modeled in terms of timing parameters, its activation can be scheduled in coordination with other loads by an algorithm such as EDF. The type of loads that may be fruitfully managed by this approach include refrigerators, HVACs, air compressors, pumps, battery charge/discharge (e.g., in electric vehicles), lighting, and household appliances.

IV. Related works

The techniques described in this paper can be seen as viable heuristics to face the problem of peak load reduction of power demand, as well as methods to reduce the overall electric energy demand. While there is an extended literature on power systems addressing the problem of peak load limitation and load balancing in general, in this paper we will focus on scientific results related to scheduling approaches applied to power management.

In [5] scheduling techniques are applied to the control of on/off smart loads in the Smart Grid. Several control approaches are presented, including a priority-based and a round-robin scheduling schemes. An efficient heuristic to handle large sets of heterogeneous loads is presented in [6]. The problem is formulated as an optimization problem, and a flexible and efficient heuristic is developed to solve the problem. In [7] there is an accurate physical modeling of several kind of loads to be incorporated into a home energy management (HEM) system. The modeling is used to propose a priority-based control scheme to achieve a bound on the peak load of power demand.

Recently, the notion of *Real-Time Physical System* (RTPS) has been introduced to indicate a general class of systems where the variation of a physical value is determined by the schedule generated by a real-time scheduling policy [8]. This approach is behind the ideas described in this paper. The mentioned work addresses systems with linear dynamics, thus having an exponential behavior in the time domain, under timing and physical constraints. In [9] an optimization method is proposed to reduce the peak load when the activation of multiple loads at the same time is allowed, which is a common situation in large systems. However, the system model does not consider constraints on the state variable. In [10], constant dynamics were considered without feedback on the state variable. The concept of RTPS in absence of constraints on the physical variables (user requirements) is introduced in [11]. The paper provides a statistical evaluation of the benefit of using RTPS against the case of absence of explicit load control. Finally, in [12] RTPS are extended with a feedback scheme for the management of electric load in presence of uncertainties on the values of modeling parameters. The contribution of these papers is to provide the relationship between the variation of physical variables (temperature, air pressure, battery charge, etc.) and the timing parameters adopted to model the related load.

V. Real-time concepts and load management

This section introduces some of the most highly investigated issues in the field of Real-Time Systems. Their applicability and usefulness to the management of power loads is described, outlining their benefits.

A. A-priori guarantees

A peculiar feature of results developed in the field of RTSs is their strong theoretical background. This means that the vast majority of results, including analysis methodologies, characteristics of scheduling algorithms and their performance, are mathematically proved. The mathematical strength of the real-time scheduling analysis includes the derivation of schedulability tests for many task and system models [3]. Tests are based on the timing parameters of tasks. Under adequate assumptions, if the test is passed then the system will behave as expected in every possible condition. In this way, schedulability tests provide a mathematically proved *a-priori guarantee* on the system behavior.

Such guarantees can be extended to the peak load generated by a load set, thus resulting in physical processes that perform as expected while achieving an upper bound on the peak load of required power. This approach is used in [9] to determine a worst case bound of the peak load.

B. Scheduling algorithms

The scheduling algorithm encapsulates the policy that determines the sequence of activation of tasks/loads. In practice, it sets the actual priority of a task based on its timing parameters. At any given time, the task with the highest priority is selected for execution, while the remaining tasks are delayed. The study of scheduling algorithms is a central argument in the research on real-time systems, and several algorithms covering a wide range of scheduling problems variants are available.

This concept has a relevant relationship with power/energy systems. Load shedding techniques are often based on explicit specification of load priorities required to select the sheddable ones. The main advantage of real-time techniques is that the priority selection is not left to the system designer. Instead, it is deterministically based on timing and physical requirements that are translated into the timing constraints associated with tasks.

C. Aperiodic activities

Aperiodic activities are those tasks that need to be scheduled "on demand", i.e., without a fixed periodic activation. In the domain of electric loads management, examples of aperiodic tasks include TVs, elevators and ovens. In general, such loads must be activated after a user request without delay. In [7], this type of loads is said *critical*.

There are several approaches to integrate aperiodic tasks in a schedule. When such tasks need to coexist with periodic activities, the common solution is to use a *server* that periodically serves the execution of aperiodic requests. In this way, aperiodic requests can nicely be incorporated in a schedule made by periodic tasks. Other solutions exist. For example, in the gravitational task model a set of aperiodic activities need to be executed at precise time instant [13]. The gravitational model allows to minimize the offset of a task execution with respect to the desired time instant.

D. Online and offline scheduling

Most scheduling approaches in the field of RTSs are developed as online algorithms. An online algorithm dynamically generates the scheduling decisions at run-time, i.e., while the system is actually working. This approach has the strong benefit to allow a timely and flexible management of faulty conditions, such as overloads. The possibility to use online scheduling algorithms comes at the price of imposing simplifying assumptions on the system model.

Therefore, complex constraints can not be easily managed by online algorithms. For this purpose, offline scheduling is conceived as an optimization process that is performed before the actual system working, to a-priori allocate the required resources [3]. This method extends the potential of real-time scheduling, at the price of less run-time flexibility and higher computational complexity.

Similar considerations apply to loads management. Simple cases can be efficiently managed by online algorithms, while more complex situations, in terms of constraints, can benefit of off-line solutions. For example, the complex interplay between energy demand and energy production from renewable sources may require a costly computation to satisfy the imposed economic constraints.

E. Hard/soft task models: worst-case vs average cases

Almost all RTSs models are based on the assignment of *deadlines* to the task to be scheduled, as stated in Section II. When a process terminates its activity after the assigned deadline the correct system behavior may be jeopardized. Regarding the achievement of deadlines, real-time systems are either classified as *hard* or *soft*. While the former do not tolerate the violation of any deadline, in latter systems missing deadlines are allowed. Examples of hard systems are safety critical applications (e.g., avionics, automotive, etc.), while soft systems are found in the multimedia (non-critical) domain [3].

The concept of hard/soft systems can be easily extended to load management. Beside timing constraints, there are often physical requirements associated with the activation of a load. While in some cases the missing of a physical constraint is not allowed, in many cases the imposed physical constraints can be sometimes violated to account for special situations. For example, consider the room temperature regulated by an HVAC that is required to remain within the range $18 - 20°C$. A missed deadline may cause the temperature to temporarily reach $21°C$ without considerably affect the user comfort. In other cases the worst case conditions need to be achieved, since they are the potential source of power provisioning disruption (black outs) [2].

F. Scalability to large systems

The set of loads to be controlled in a realistic scenario can be composed by a large number of devices. Scalability issues may arise in large networks. In RTSs many useful techniques, including scheduling algorithms and analysis methodologies, have linear or polynomial computational complexity. Thus their application can suitably fit to large load sets. This section illustrates the approaches that can be leveraged to face the management of large load sets.

1) Multiprocessor scheduling: In RTSs, multiprocessors are those systems that allow to execute more than one task at any given time, since each task runs on a different processor [14]. Multiprocessors can clearly manage a larger number of tasks with respect to uniprocessor systems. However, in computing systems the number of processors is imposed by the computing platform (i.e., it is a system constraint). Therefore, the typical goal is to provide an answer to the question: “Is the considered task set successfully schedulable by the available processors?”, where a successful schedule is the one that satisfies the timing constraints of every task.

The application of real-time scheduling to the load management is inherently close to the multiprocessor scenario. The number of loads can be large, thus the simultaneous activation of two or more loads can hardly be avoided. However, the number of simultaneously activated loads is not a system constraint in this case. Instead, the reduction of unnecessary simultaneous activations is the objective of the approach based on real-time scheduling. Therefore, the above question can be translated to “What is the maximum number of simultaneous activated loads?”. The final goal is to determine the resulting peak load in the worst case. The resulting value can be used to either determine the size of a new infrastructure or to verify the suitability of the schedule to the power constraints imposed by an existing system (e.g., to check whether contractual limits are satisfied).

2) Hierarchical scheduling: A typical method to manage large systems is to identify a hierarchy in the system components. In RTSs the hierarchical scheduling is used to aggregate heterogeneous scheduling policies in a predictable manner [15]. A schedule generated at a higher level can contain a sub-schedule, whose formal properties influence and can be used to analyze the overall system behavior.

The modeling of power infrastructure can be straightforwardly organized in a hierarchical manner. The hierarchy starts from electric loads/devices at a lower level and grows into aggregated levels as apartments/houses, buildings, districts and cities. The idea is that a predictable schedule guaranteed at lower levels, and thus a predictable power load, allows to achieve predictable aggregated power loads at higher levels. This approach has huge potentials in the management of large and heterogeneous power systems.

G. Approximated scheduling

The estimation of the value of relevant parameters to characterize a physical process is often affected by some degree of uncertainty. The uncertainty can derive from incomplete knowledge of the process, from simplified assumption to obtain simpler models, etc.

In RTSs, an approach to deal with uncertainties is the so-called probabilistic scheduling. It allows to incorporate the uncertainty regarding physical and timing parameters in the system model. For this purpose, the common timing parameters such as periods and execution times are expressed by stochastic variables. The stochastic nature of timing parameters is used to derive probabilistic guarantees on the system behavior [16].

The possibility to incorporate a sufficient degree of uncertainty in the model is essential in the practical applicability

TABLE I
A RESUME OF THE PROPOSED REAL-TIME METHODOLOGIES AND THEIR APPLICATION TO POWER LOAD MANAGEMENT.

Feature	Application to power load management
A-priori guarantees	To determine useful system properties (e.g., the peak load) from timing parameters without the need of simulations
Scheduling algorithms	To coordinate the intelligent activation of power consuming devices, avoiding unnecessary simultaneous activations
Aperiodic activities	To efficiently manage the concurrent activation of periodic loads and aperiodic ones (e.g., ovens, dishwashers, washing machines)
Online/offline scheduling	To deal with mixed complex application constraints (offline) and dynamic adaptation of system requirements at runtime (online)
Hard/soft task models	To integrate both critical (hard) loads and less urgent (soft) ones
Multiprocessor scheduling	To manage the activation of two or more loads at the same time, when this condition can not be avoided (e.g., in large load sets)
Hierarchical scheduling	To coordinate a large power system that is decomposed into a set of sub-systems organized in a hierarchical manner
Approximated scheduling	To cope with modeling uncertainties introduced in the modeling of power loads and systems

of the scheduling approach to energy systems, where the exact value of physical parameters are always affected by errors and/or approximations. Robust scheduling methods are then required to cope with such issues and achieve reliable results under uncertain conditions.

VI. CONCLUSIONS

This paper has suggested the applicability of some existing real-time scheduling approaches with explicit indication of their possible applications to power loads management. In particular, basic concepts of real-time scheduling have been introduced to allow the subsequent coverage of more advanced topics in connection with power management.

REFERENCES

[1] A. Fazeli, E. Christopher, C. Johnson, M. Gillott, and M. Sumner, “Investigating the effects of dynamic demand side management within intelligent smart energy communities of future decentralized power system,” in Innovative Smart Grid Technologies (ISGT Europe), 2nd IEEE PES Int. Conf. and Exhibition on, dec. 2011, pp. 1–8.

[2] 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.

[3] G. Buttazzo, Hard Real-Time Computing Systems: Predictable Scheduling Algorithms and Applications. Springer, 2011.

[4] 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.

[5] G. Koutitas, “Control of flexible smart devices in the smart grid,” Smart Grid, IEEE Transactions on, vol. PP, no. 99, pp. 1–11, 2012.

[6] T. Logenthiran, D. Srinivasan, and T. Z. Shun, “Demand side management in smart grid using heuristic optimization,” Smart Grid, IEEE Transactions on, vol. PP, no. 99, pp. 1–9, 2012.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] 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.

[12] T. Facchinetti and M. L. Della Vedova, “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.

[13] R. Guerra and G. Fohler, “A gravitational task model with arbitrary anchor points for target sensitive real-time applications,” Real-Time Systems, vol. 43, no. 1, pp. 93–115, 2009.

[14] R. I. Davis and A. Burns, “A survey of hard real-time scheduling for multiprocessor systems,” ACM Computing Surveys, vol. 43, no. 4, pp. 1–44, Oct. 2011.

[15] J. Regehr, A. Reid, K. Webb, M. Parker, and J. Lepreau, “Evolving real-time systems using hierarchical scheduling and concurrency analysis,” in Proceedings of the IEEE Real-Time Systems Symposium, dec. 2003, pp. 25–36.

[16] A. Burns, S. Punnekkat, L. Strigini, and D. Wright, “Probabilistic scheduling guarantees for fault-tolerant real-time systems,” in Dependable Computing for Critical Applications 7, 1999, nov 1999, pp. 361–378.