Feedback Scheduling of Real-Time Physical Systems with Integrator Dynamics

Marco L. Della Vedova and Tullio Facchinetti
Dpt. of Industrial and Information Engineering, University of Pavia
via Ferrata 1, 27100 Pavia, Italy
{marco.dellavedova, tullio.facchinetti}@unipv.it

Abstract

This paper addresses the application of real-time scheduling for the reduction of the peak load of power consumption generated by electric loads in a power system. The considered physical processes are characterized by integrator dynamics and modeled as sporadic real-time activities. To enable the applicability in realistic scenarios, modeling approximations and uncertainties on physical parameters are explicitly included in the model. A feedback control strategy is proposed to guarantee the requirements on physical values under control in presence of modeling and measurement uncertainties. To compensate such uncertainties, the value of timing parameters used by the scheduler are dynamically adapted. Formal results have been derived to put into relationship the values of quantities describing the physical process with real-time parameters used to model and to schedule the activation of loads.

1 Introduction

Peak load reduction is fundamental for the correct and efficient operation of power systems [4, 9]. Peak load conditions, i.e., arising from usage of a large amount of electric power by many simultaneously activated loads, may cause severe problems such as the disruption of power provision, leading to technical and economic issues for both energy providers and customers. There is an extended literature on power systems dealing with related topics as peak shaving, load balancing, Demand-Side Management, and Direct Load Control. Possible approaches include control systems and optimization techniques, artificial intelligence (fuzzy logic, neural networks, expert systems, etc.), and methods facing economic/regulatory issues. An up-to-date overview and classification of available techniques can be found in [12].

Recently, methods based on real-time scheduling have been proposed to manage a sset of loads in a power system [8]. Later, Real-Time Physical Systems (RTPSs) have been introduced as a paradigm to model and control a physical process where the variation of physical variables is associated with a real-time schedule [5, 7]. The primary application of RTPSs is the management of electric loads in power systems to achieve the reduction of the peak load. In RTPSs, the value of physical variables change according to the state of activation (on/off) of the real-time schedule. Such value is required to remain bounded within a specified working range. The management of concurrency featured by real-time scheduling algorithms is leveraged to optimize the peak load of electric power consumption in a energy/power system subject to physical constraints. The idea is to timely and predictably schedule the activation of power-consuming devices in order to limit unnecessary simultaneous activations, thus reducing the peak load. The physical system is modeled using timing parameters typically used in real-time systems. Real-time parameters must be properly set such that every state variable is bounded within the desired working range. It is worth to outline that RTPSs do not deal with real-time processing tasks whose computation triggers the activation/deactivation of a power load. Instead, the activation of power loads is the actual entity which is directly associated with scheduling events (activations/deactivations) generated by a real-time scheduling algorithm.

This paper proposes a dedicated feedback control scheme to deal with modeling errors and uncertainties in a RTPS. This is particularly relevant in practical applications, where values of modeling parameters are always affected by some source of uncertainty. The derived results extend the ones carried out in [8], where a similar system model and control techniques have been developed, but without accounting for modeling errors. The method was essentially an open-loop control strategy, since no information was acquired and used at run-time to adjust possible mismatches between expected and actual system trends. In particular, without an adequate compensation, the correct system behavior could have been jeopardized by the effect of unmodeled errors, as shown in this paper. In this work, errors are explicitly introduced and properly modeled. On the basis of this modeling effort, a feedback scheduling rule is proposed to dynamically compensate the effect of uncertainties. The result is the achievement of requirements on working ranges imposed on physical variables. The provided analysis allows to determine the relationship between physical parameters and timing parameters. Therefore, the derived results translate the requirements on the physical variables into timing constraints used to schedule the activation of electric loads. In [7] RTPSs have been developed considering constraints on state variable variations and modeling errors. However, uncertainties have been specified in terms of statistical distributions (soft physical requirements), while worst-case scenarios are considered in this paper (hard physical requirements), leading to a completely different analysis approach.

2 Real-Time Physical Systems model

In general terms, a RTPS is made by a set of elements, where each element is composed by two interacting aspects: a *physical subsystem*, characterized by a continuous timing, and a *controller*, which has a discrete behavior. A state variable is associated with the physical subsystem, while the controller models and regulates the physical process by generating a schedule that determines the actual behavior of the state variable. The schedule is built using a real-time scheduling algorithm. For example, Figure 1 shows two components whose state variables x have integrator dynamics (i.e., ramps in the time domain). The physical subsystems represented in Figure 1 may be two refrigerators. The physical quantity of interest is the internal temperature, which decreases when the refrigerator is active, while it increases otherwise. By definition, a controller is said to be active when it is scheduled for execution/running; otherwise it is not active. It is worth to note that, clearly, considering integrator dynamics for a refrigerators (as well as for other types of loads that usually have linear dynamics, i.e., exponential behavior in the time domain) represents a modeling approximation. However, it can be realistically afforded in many practical situations (e.g., [10, 3]).

In RTPS the state variable behavior is only affected by the state of activation of the corresponding load. Therefore, the state of activation of a load is related with typical *scheduling events* of a real-time schedule, such as activations, terminations, and preemptions. The key observation is that the behavior of a state variable does not depend on the computation outcome of a real-time processing task, as it happens in traditional control systems implemented on top of a Real-Time Operating System. In fact, in real-time control tasks, the control action imposed on a physical devices mainly derives from *numerical results* produced by the control algorithm implemented by the task. In a RTPS the control action is implicitly determined by the state of activation of the controller, i.e., it directly depends on the schedule generated by the controller. As a result, the control problem of the physical process requires to be studied in conjunction with the real-time scheduling problem. In particular, the scheduling of controllers affects the dynamics behavior of the physical process and related state variables. On the other hand, constraints on the physical process determine the selection of timing parameters at design time, while the value of state variables

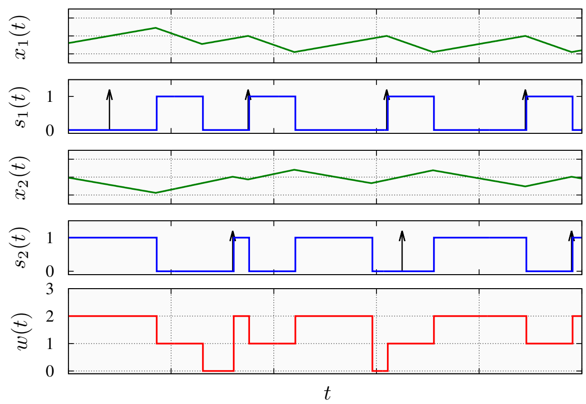

Figure 1. Example of RTPS made by two components. The behavior of state variables x_i are determined by the schedule s_i. The total power demand $w(t)$ is the sum of power consumed by the two devices.

can drive scheduling decisions at run-time.

2.1 System dynamics

The system considered in this paper is composed by n electric on/off loads. Each load acts on one subsystem, whose dynamic behavior is described by a time-dependent controlled switched hybrid system [11] with integrator dynamics. Subsystems are independent from each others, and each load operates on one physical subsystem only. The dynamics of the i-th subsystem is described by (1).

$$\begin{cases} \dfrac{dx_i(t)}{dt} = f_i^{s_i(t)} = \begin{cases} -\alpha_i^{\text{on}} & \text{if } s_i(t) = 1 \\ +\alpha_i^{\text{off}} & \text{if } s_i(t) = 0 \end{cases} \\ x_i(0) = \bar{x}_i \end{cases} \tag{1}$$

The quantities involved in (1) are:

- $t \in \mathbb{R}^+$ is the continuous time span;
- $x_i(t) \in \mathbb{R}$ is the state variable of the subsystem and represents the physical quantity of interest;
- $\bar{x}_i$ is the initial value of the state variable;
- $s_i(t) \in \mathbb{B} \equiv \{0, 1\}$ is the *operation mode* of the subsystem. It represents the activation status of the i-th load: $s_i(t) = 0$ if the load is not active at time t, and conversely, $s_i(t) = 1$ when the load is active;
- $f_i^{s_i(t)} \in \mathbb{R}$ is the integrator dynamics' parameter of the subsystem i-th at time t; its value can be either $\alpha_i^{\text{on}} \in \mathbb{R}$ or $\alpha_i^{\text{off}} \in \mathbb{R}$ depending on the value of $s_i(t)$.

Considering the whole system, $s : \mathbb{R}^+ \to \mathbb{B}^n = [s_1 \ \dots \ s_n]$ is called *switching signal* or *schedule*. The previous model describes a system where the evolution over time of the state variables depends on the activation status of the loads. Loads' status is driven by the switching signal. This signal is generated by a centralized controller, called *scheduler*.

2.2 Switching signal

The distinguishing point of a RTPS is that the switching signal is generated by a real-time scheduling algorithm, such as the Earliest Deadline First algorithm (EDF). The result is an effective control strategy to reduce the unnecessary simultaneous activations of loads. In fact, at any given instant, the scheduler will automatically limit the activation to those loads that are required to avoid the violation of timing constraints. Since the peak load of power consumption in a given t time instant is the sum of the power consumed by all loads simultaneously active at time t, by reducing number of simultaneous activations the RTPS approach results an efficient method to obtain the reduction of the peak load.

Considering above observations, the modeling and control problem translates to the assignment of proper values to timing parameters and constraints associated with each load. For this purpose, a set of real-time parameters are associated to each electric load. The adopted model derives from the sporadic task model [1]. The generic i-th load is associated with the tuple (T_i, D_i, C_i). The meaning of such parameters is the following:

- $T_i \in \mathbb{R}^+$: it is the minimum time frame 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 $r_{i,k+1} - r_{i,k} \geq T_i$, $k \in \mathbb{N}$;
- $D_i \in \mathbb{R}^+ : D_i \leq T_i$ is the relative deadline; it defines the time frame $[r_{i,k}, r_{i,k} + D_i]$, for $k \in \mathbb{N}$, in which a load must perform its activity within each period;
- $C_i \in \mathbb{R}^+ : C_i \leq D_i$ represents the activation time duration of a load within each period T_i;

Previous real-time parameters are used by the scheduling algorithm to generate the switching signal, i.e., the schedule. The scheduling algorithm is said to be *closed-loop* if it considers the actual value of the state variables $x(t)$ to generate the schedule; it is said *open-loop* otherwise. In case of a closed-loop algorithm, which is the one considered in this paper, the scheduler sets proper values of real-time parameters at each request time. This is required to compensate possible uncertainties on physical modeling parameters. Based on the measurement of the state variable at the generic k-th request time $x_i(r_{i,k})$, the scheduler computes the values of $T_{i,k}$, $D_{i,k}$ and $C_{i,k}$ to be used in the next period. The scheduler also sets the next request time to $r_{i,k+1} = r_{i,k} + T_{i,k}$. The values of timing parameters will be properly bounded in order to allow the use of bounds to perform the schedulability analysis in the worst case. Results derived in this paper are oriented to determine the values of such bounds.

In the prospective of using a closed-loop scheduling approach, a valid schedule is defined as follows.

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 $C_{i,k} \leq C_i$ within each time interval $[r_{i,k}, r_{i,k} + D_{i,k}]$, having $C_{i,k} \leq D_{i,k} \leq T_{i,k}$. Formally, it holds:*

$$\forall i, \forall k \quad \int_{r_{i,k}}^{r_{i,k}+D_{i,k}} s_i(t)\,\mathrm{d}t = C_{i,k} \tag{2}$$

Note that the definition of valid schedule is slightly different from the one applicable to traditional real-time systems. In particular, (2) is an equality instead of a less-then-equal relation.

2.3 User requirements

User requirements are a set of constraints on the physical quantities of interest. They capture the desired behavior of the physical process. User requirements considered in this paper are stated such as the physical quantity of interest of each subsystem requires to be bounded within a given working range:

$$x_i(t) \in \Psi_i \equiv \left[x_i^{\min}, x_i^{\max}\right] \tag{3}$$

An example of this kind of requirements is the internal temperature of a refrigerator, which needs to be maintained within the desired range.

2.4 The RTPS feasibility problem

As from previous definitions, a RTPS is composed by a dynamic system, user requirements, real-time parameters and a scheduling algorithm. While the dynamic system and user requirements are related with the underlying physical process, 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 of feasible RTPS.

Definition 2 (Feasibility)**.** *A RTPS is said to be* feasible *if and only if user requirements can be satisfied by a valid schedule.*

Equation (2) identifies 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 user requirements are guaranteed. This problem translates to the identification of suitable values for C_i, D_i and T_i to drive the evolution of physical variables in compliance with user requirements.

The analysis approach 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.

Definition 2 refers to *hard user requirements*. Hard user requirements are those that can be never violated. In [7], instead, authors have addressed *soft user requirements*, which are those that can be occasionally infringed.

2.5 Peak load limitation

The application of RTPSs proposed in this paper is to limit the peak load of power consumption generated by a set of electric loads, while meeting requirements on physical values. Each electric device can be either active or not. The activity of loads is controlled by the scheduler that generates the s_i schedule for the i-th load. The i-th device consumes either a $P_i \in \mathbb{R}^+$ amount of electric power when active, no power otherwise. Hence, the power consumption over time is modeled with the function $p : \mathbb{R}^+ \to \mathbb{R}^+$ defined in (4).

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

This paper does not consider transient phases between active and inactive states.

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

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

The maximum value of $w(t)$ is the peak load, and the goal of the proposed approach is to reduce it. The peak load minimization problem can be formally defined as follows.

Definition 3 (Peak load minimization problem)**.** *The peak load minimization problem consists in finding the optimal schedule $s^* : \mathbb{R}^+ \to \mathbb{B}^n$, which minimizes the peak load and satisfies user requirements.*

$$s^* = \arg\min_{s} \max_{t\geq 0} \sum_{i=1}^{n} P_i s_i(t) \tag{6a}$$

$$\text{subj. to} \quad \dot{x}_i(t) = f_i^{s(t)}, \quad \forall i \tag{6b}$$

$$x_i(t) \in \Psi_i, \quad \forall i \tag{6c}$$

A RTPS scheduler generates the optimal schedule s^* when a uniprocessor 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 power consumed by 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 uniprocessor 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. For example, in [6] an heuristics based on bi-dimensional bin-packing has been investigated to divide the load set in subsets such that each subset can be scheduled independently by an uniprocessor scheduling algorithm. This method resembles the partitioned scheduling approach proposed for multiprocessor real-time systems. In this case the peak load is bounded by the sum of powers of the most power-consuming load in each subset.

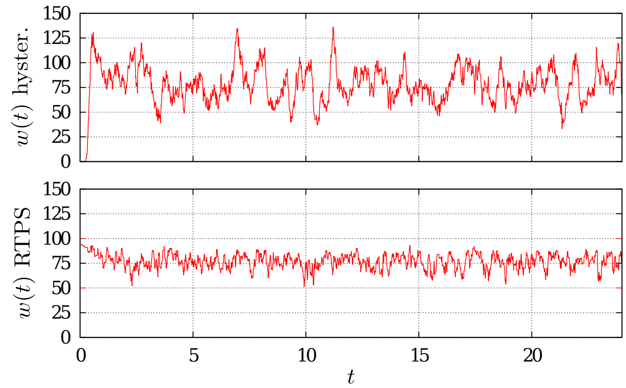

Figure 2. Comparison of the total power consumption of 100 randomly generated loads controlled by an hysteresis controller, and a RTPS controller.

Figure 2 shows a simulation example of the total power demand of a set of 100 loads, which have been scheduled by a traditional hysteresis controller in comparison to the proposed RTPS approach. In this case, the peak load is reduced by the 31% and the standard deviation of the power demand is reduced by the 61%. The overall power consumption remains close to the average value, as desired.

3 Properties and results

In this section, interesting properties regarding the relationship between real-time and physical parameters are derived. Since physical subsystems are independent from each others, the analysis will deal with one subsystem only. 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 $U^{\text{tot}} \doteq \sum_{i=1}^{n} U_i$. The former is the fraction of time in which the i-th load is active, while the latter is the total fraction of activity time of the whole load set.

This section firstly re-calls some results originally presented and proved in [8] regarding RTPS with integrator dynamics in absence of modeling errors, that will be extended to include modeling errors in next sections.

3.1 Properties with no model mismatch

The first result regards the evolution of the state variable when the system is driven by a valid switching signal. In particular, Observation 1 indicates the state variable value in correspondence of request times.

Observation 1. *For a dynamic system* (1) *controlled by a valid schedule* (2)*, it holds:*

$$x_i(r_{i,k+1}) = x_i(r_{i,k}) - \alpha_i^{\text{on}} C_{i,k} + \alpha_i^{\text{off}}(T_{i,k} - C_{i,k}) \quad (7)$$

Taking the previous observation into account, it is possible to establish a relationship between the load utilization U_i and the dynamics of the related physical process.

Observation 2. *Given the system dynamics as in* (1)*, if the switching signal is valid and it is characterized by $C_{i,k} = C_i$ and $T_{i,k} = T_i$, $\forall k$, then it holds:*

$$x_i(kT_i) = \bar{x}_i,\ \forall k \in \mathbb{N} \iff U_i = \frac{\alpha_i^{\text{off}}}{\alpha_i^{\text{on}} + \alpha_i^{\text{off}}}. \quad (8)$$

Observation 2, which can be easily verified by replacing the value of U_i into (7), states that, for every load, the state variable assumes the same value $\bar{x}_i$ at every request time $r_{i,k}$ if and only if its utilization is set as in (8). It also shows that, to achieve this result, the load utilization U_i depends only on α^{off} and α^{on}. Two key consequences derive from Observation 1 and 2. First, since a state variable assumes the same value at every request time, the analysis of global properties (i.e., for every time t) of the state variable can be performed by restricting the analysis to one period. Second, since the remaining results derived in this section are based on Observation 2, they hold for deadlines less than periods. In fact, (7) does not depend from the actual points in time when a load is activated within a period. Therefore, deadlines can be shortened to improve the system responsiveness without affecting the achievement of user requirements. Clearly, the shortening can be performed as far as timing constraints can be met by the scheduling algorithm. The formal derivation of this property can be found in [8]. This is a relevant result since, in the analysis of traditional real-time computing systems, substantial complications arise when deadlines are allowed to be less than periods. Due to above observations, deadlines will not be considered in the reminder of the paper.

The next relevant result is re-called by Theorem 1. It allows to calculate the upper bound on the period T_i such that, if used together with the load utilization U_i as in (8), it guarantees that load the state variable $x_i(t)$ is maintained within the required range $[x_i^{\min}, x_i^{\max}]$.

Theorem 1. *For a dynamic system* (1) *driven by a valid switching signal* (2) *characterized by U_i and T_i, if U_i is assigned as in* (8) *then it exist an upper bound T_i^* for the switching signal period such that:*

$$T_i < T_i^* \implies x_i^{\min} \le x_i(t) \le x_i^{\max},\ \forall t \in \mathbb{R}^+ \quad (9)$$

and the value for this upper bound is:

$$T_i^* = \min\left\{ \frac{x_i^{\max} - \bar{x}_i}{\alpha_i^{\text{off}}(1 - U_i)}, \frac{\bar{x}_i - x_i^{\min}}{\alpha_i^{\text{on}} U_i} \right\} \quad (10)$$

Theorem 1 allows to determine suitable values of timing parameters to achieve the requirements on physical variables.

3.2 Effects of modeling errors

The results introduced in Section 3.1 essentially consist of an open-loop control strategy whose parameters are tuned to meet the desired system constraints and requirements on the state variable. In particular, the utilization is set according to Observation 2. However, this approach may lead to the violation of user requirements when inaccuracies are present and not properly taken into account. Inaccuracies are determined by several factors: mismatch between the physical system and the adopted model, rounding in calculations, noise or interference on the physical system. For example, when the value of α_i^{off} and/or α_i^{on} is subject to variations due to external factors (with respect to the adopted model) the results of Observation 2 may no longer hold. In other words, the value of a state variable in correspondence to the k-th request time $r_{i,k}$ may differ from the one in $r_{i,k+1}$. To cope with the effect of uncertainties, the relationship between T_i^* and physical parameters requires a deeper analysis.

This section extends the model presented in previous sections by introducing errors that model the uncertainties on (i) slopes α_i^{off} and α_i^{on} and (ii) time quantization. Since hard user requirements are addressed in this paper, uncertainties will be modeled in terms of worst case conditions. For this purpose, every parameter will be modeled with an unknown real value, which is assumed to be bounded within a given interval.

The uncertainties on state variable slopes are modeled by the parameters introduced in Definition 4.

Definition 4. *The difference between the real (unknown) values of the state variable slopes, i.e. α_i^{on} and α_i^{off}, and the actual parameter values used for the control, i.e. $\tilde{\alpha}_i^{\text{on}}, \tilde{\alpha}_i^{\text{off}} \in \mathbb{R}$, is bounded such that:*

$$\|\alpha_i^{\text{on}} - \tilde{\alpha}_i^{\text{on}}\| \le \delta_i^{\text{on}} \quad \text{(11a)}$$

$$\|\alpha_i^{\text{off}} - \tilde{\alpha}_i^{\text{off}}\| \le \delta_i^{\text{off}} \quad \text{(11b)}$$

In Definition 4, the terms $\delta_i^{\text{on}}, \delta_i^{\text{off}} \in \mathbb{R}^+$ indicate the known maximum gaps between real values and actual values of slopes used to trigger the control action.

A second source of modeling approximation is related with the quantization of real-time parameters with respect to a given time-base. Since the controller is based on a digital clock, actual scheduling actions (i.e. load activations/deactivations) can only occur at integer multiples of a time quantum $\tau \in \mathbb{R}^+$. The time quantum can be either imposed by the system, e.g. by the digital clock of the computer performing the scheduling algorithm, or it can be considered as a design parameter. In this second case, it allows to bound the minimum amount of time between two consecutive switching actions of loads. In both cases, a possible source of approximation is due to the quantization on the values of activation time C_i and period T_i with respect to the granularity introduced by τ. Errors determined by quantization arise since the values of real-time parameters derived from (8) may not necessarily be integer multiples of the time-base τ. In presence of errors, it may happen that some requirements on the state variable variation could be violated. Quantization errors are defined by Definition 5.

Definition 5. *Quantization errors δ_i^T and δ_i^C on real-time parameters T_i and C_i are defined as follows*

$$C_i = \tilde{C}_i \pm \delta_i^C \tag{12a}$$

$$T_i = \tilde{T}_i \pm \delta_i^T \tag{12b}$$

where $\tilde{T}_i$ and $\tilde{C}_i$ represent the ideal values obtained from calculations, while C_i and T_i are the quantized values.

Supposing to round the ideal value to the closest lower multiple of τ, it is straightforward to determine the values of quantization errors by observing that

$$\delta_i^C = \tilde{C}_i \bmod \tau \leq \tau \tag{13a}$$

$$\delta_i^T = \tilde{T}_i \bmod \tau \leq \tau \tag{13b}$$

It is worth to note that the effect of δ_i^T can be easily eliminated by selecting $T_i = k\tau$, for some $k \in \mathbb{N}$, provided that $T_i \leq T_i^*$ as required by Theorem 1 for achieving the user requirements. Moreover, the time quantum τ is the upper bound of the quantization error.

The effect of approximations on the system behavior is essentially related to the fact that the result of Observation 2 may no longer hold due to approximations. In other words, the value of a state variable x_i may be different in correspondence to different request times. Hence, the state variable may drift from the desired value $\bar{x}_i$. A bound on the maximum variation of the state variable due to illustrated approximations is provided by the following theorem.

Theorem 2. *At the k-th request time, the drift of the state variable value from its initial value is bounded to:*

$$\|x(r_{i,k}) - \bar{x}_i\| \leq k\epsilon_i \tag{14}$$

where

$$\epsilon_i = \tilde{C}_i(\delta_i^{\text{on}} + \delta_i^{\text{off}}) + \tilde{T}_i\delta_i^{\text{off}} + \tilde{\alpha}_i^{\text{on}}\delta_i^C + \tilde{\alpha}_i^{\text{off}}(\delta_i^C + \delta_i^T) + \delta_i^{\text{on}}\delta_i^C + \delta_i^{\text{off}}\delta_i^C + \delta_i^{\text{off}}\delta_i^T \tag{15}$$

Proof. The proof is based on the observation that, once errors have been modeled as in Definition 4 and 5, the terms in (7) provided by Obs. 1 are all affected by errors. The goal is to find the value of ϵ_i, which is an upper bound on the absolute value of the error on $x(kT_i)$ by (14). The value of ϵ_i can be derived from (7) by considering the additive and multiplicative properties of uncertain values (see [13] for details). □

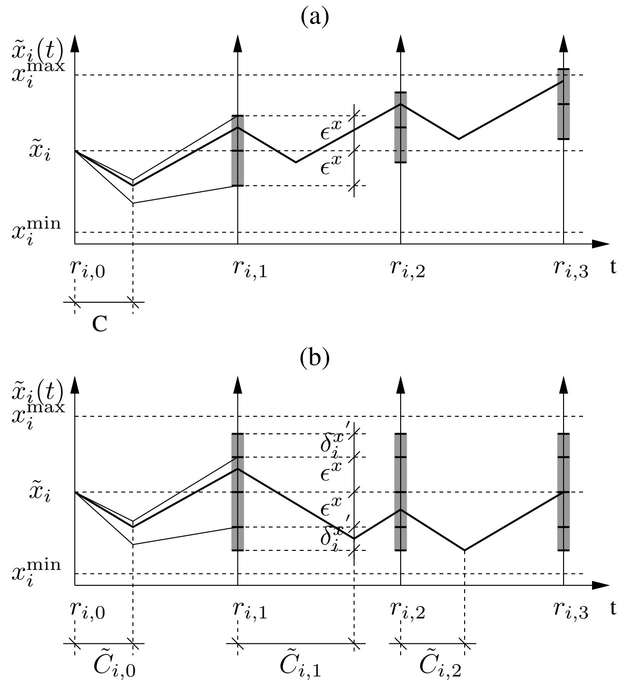

Figure 3. Effect of errors in absence (a) and with compensation (b). In (a) the state variable constantly drifts from the desired value $x_i(r_{i,k})$, causing the violation of user requirements. In (b) the state variable at every request time $r_{i,k}$ is maintained within a bounded range centered in $\bar{x}_i$ by properly setting the value of $\tilde{C}_{i,k}$.

The state variable value results to be increased or decreased by a maximum of ϵ_i at each subsequent request time, as illustrated in Figure 3.a. Therefore, the system can no longer be suitably controlled by using the value of timing parameters as calculated in absence of errors. This issue can not be avoided unless a proper feedback on the state variable is introduced to dynamically adapt real-time parameters according to measured values. In other words, it is mandatory to suitably *measure* the actual value of the state variable to compensate the effect of uncertainties.

3.3 Using feedback to cope with uncertainties

To properly control the system in presence of errors the effect of such errors must be compensated. For this purpose, a feedback approach is proposed to adapt the value of timing parameters $T_{i,k}$ and $C_{i,k}$ at every request time $r_{i,k}$. The adapted value will be valid for the next time frame $[r_{i,k}, r_{i,k+1})$. The idea is to measure the value of the state variable in correspondence to a request time. The measured value at time t is denoted with $\hat{x}_i(t)$. The detected gap between measured value $\hat{x}_i(r_{i,k})$ and expected value $\bar{x}_i$ is used to calculate the actual values of $C_{i,k}$ and $T_{i,k}$. Such values are set to guarantee that the state variable will fall into a bounded range in correspondence to the next request time $r_{i,k+1}$, and user requirements are met in the time frame $[r_{i,k}, r_{i,k+1})$.

Since the feedback technique is based on the measurement of the state variable in correspondence with request times, the measurement error is firstly defined to account for the uncertainty on the measurement.

Definition 6. *The measurement error on the state variable $x_i(t)$ is bounded by a known constant δ_i^x, defined as*

$$\|\hat{x}_i(t) - x_i(t)\| \leq \delta_i^x \tag{16}$$

where $\hat{x}_i(t)$ represents the measured value, while $x_i(t)$ is its unknown real value.

Considering the model for the measurement error, a theoretical result is provided to allow the compensation of errors arising from sources modeled by definitions 4, 5 and 6.

Theorem 3. *Given the system model* (1)*, the definitions of parameter uncertainties* (11)-(12) *and the error model on sensor measurements* (16)*, if $\tilde{C}_{i,k}$ and $\tilde{T}_{i,k}$ are set in order to balance the following equation*

$$\bar{x}_i - \hat{x}_i(r_{i,k}) = \tilde{T}_{i,k}\tilde{\alpha}_i^{\text{off}} - \tilde{C}_{i,k}\left(\tilde{\alpha}_i^{\text{on}} + \tilde{\alpha}_i^{\text{off}}\right) \tag{17}$$

then

$$\|x_i(r_{i,k}) - \bar{x}_i\| \leq \epsilon_{i,k} + \delta_i^x, \quad \forall k \in \mathbb{N} \tag{18}$$

Proof. The goal is to determine the values of $T_{i,k}$ and $C_{i,k}$ so that $x_i(r_{i,k+1}) = x_i(r_{i,k}) = \bar{x}_i$ in (7). However, the terms in (7) are affected by errors with known bounds, as stated in Definitions 4, 5 and 6, where known terms are $\hat{x}_i$, $\tilde{\alpha}_i^{\text{off}}$, $\tilde{\alpha}_i^{\text{on}}$. Therefore, it can not be achieved to obtain exactly $x_i(r_{i,k+1}) = \bar{x}_i$. However, it is guaranteed that $x_i(r_{i,k+1})$ will fall in an interval containing $\bar{x}_i$, as stated in (18). This latter is obtained simply from (7) by inserting the expression of the errors. Finally, since $x_i(r_{i,0}) = \bar{x}_i$ by the system model definition (1), (18) holds for every $k \in \mathbb{N}$. □

In (18), the term $\epsilon_{i,k}$ has the same expression of ϵ_i (i.e. (15)) where the terms $\tilde{C}_{i,k}$ and $\tilde{T}_{i,k}$ replace $\tilde{C}_i$ and $\tilde{T}_i$, respectively. Considering a bound on the maximum value of period and activation time and calling it $C_i^* \doteq \sup_k C_{i,k}$ and $T_i^* \doteq \sup_k T_{i,k}$, it is possible to bound $\epsilon_{i,k}$ as follows:

$$0 \leq \epsilon_{i,k} \leq \epsilon_i^*, \quad \forall k \in \mathbb{N}. \tag{19}$$

Theorem 3 provides a convenient method to achieve the desired system behavior in presence of errors. In fact, the state variable is bounded in the range $\bar{x}_i \pm (\epsilon_i + \delta_i^x)$ at each request time $r_{i,k} : \forall k \in \mathbb{N}$. See Figure 3.b for an illustration where $\tilde{C}_{i,k}$ is changed while $\tilde{T}_{i,k}$ is kept constant.

In order to meet the user requirements Ψ_i expressed in (3), Theorem 1 must be extended to take into account modeling and measurements errors and the variability of $\tilde{T}_{i,k}$ and $\tilde{C}_{i,k}$.

Theorem 4. *If the values of $\tilde{T}_{i,k}$ and $\tilde{C}_{i,k}$ are chosen to balance Eq.* (17) *and the following conditions are satisfied*

$$\begin{cases} x_i^{\min} \leq \hat{x}_i(r_{i,k}) - \delta_i^x - (\tilde{\alpha}_i^{\text{on}} + \delta_i^{\text{on}})C_{i,k} \\ x_i^{\max} \geq \hat{x}_i(r_{i,k}) + \delta_i^x + (\tilde{\alpha}_i^{\text{off}} + \delta_i^{\text{off}})(T_{i,k} - C_{i,k}) \end{cases} \tag{20}$$

then

$$x_i(t) \in \left[x_i^{\min}, x_i^{\max}\right], \quad \forall t \in \mathbb{R}^+$$

Proof. In [8] has been proved that the maximum (minimum) possible value of the state variable x_i, depending on the scheduling signal s_i, occurs when the activation time is concentrated at the end (beginning) of the time frame defined by one period. Formally,

$$\inf_{t \in [r_{i,k}, r_{i,k+1}]} x_i(t) = x_i(r_{i,k}) - \alpha_i^{\text{on}} C_{i,k} \tag{21a}$$

$$\sup_{t \in [r_{i,k}, r_{i,k+1}]} x_i(t) = x_i(r_{i,k}) + \alpha_i^{\text{off}}(T_{i,k} - C_{i,k}) \tag{21b}$$

Thus it must be guaranteed that:

$$\begin{cases} x_i^{\min} \leq x_i(r_{i,k}) - \alpha_i^{\text{on}} C_{i,k} \\ x_i^{\max} \geq x_i(r_{i,k}) + \alpha_i^{\text{off}}(T_{i,k} - C_{i,k}) \end{cases} \tag{22}$$

Considering the worst case scenario for errors that affect the terms in (22), (20) follows directly and the thesis of the Theorem holds since (21). □

From (16), (18) and (19), it follows that the gap between measurement and expected value of the state variable at the k-th request time is bounded:

$$\|\hat{x}_i(r_{i,k}) - \bar{x}_i\| \leq \epsilon_i^* + 2\delta_i^x, \quad \forall k \in \mathbb{N} \tag{23}$$

Thanks to (23) and considering the constraints derived in (20), upper bounds T_i^* and C_i^* can be calculated, respectively on period and activation time, so that user requirements are satisfied.

The opportunity to select appropriate periods and activation times at every request time suggests some interesting considerations related to the guarantee of timing constraints. In fact, the utilization $U_{i,k} = C_{i,k}/T_{i,k}$ ranges in an interval from 0% (i.e. when $C_{i,k} = 0$ and $T_{i,k} > 0$, which are possible values according to the constraints) to 100% (for some $C_{i,k} = T_{i,k} > 0$). From (17), it is possible to derive the expression of $U_{i,k}$ in the $[r_{i,k}, r_{i,k+1}]$ time frame as follows. To simplify the presentation of subsequent results, two equivalent expressions are provided, respectively, as a function of $C_{i,k}$ and $T_{i,k}$.

$$U_{i,k} = \frac{C_{i,k}\tilde{\alpha}_i^{\text{off}}}{C_{i,k}(\tilde{\alpha}_i^{\text{off}} + \tilde{\alpha}_i^{\text{on}}) + \bar{x}_i - \hat{x}_i(r_{i,k})} \tag{24a}$$

$$= \frac{T_{i,k}\tilde{\alpha}_i^{\text{off}} + \hat{x}_i(r_{i,k}) - \bar{x}_i}{T_{i,k}(\tilde{\alpha}_i^{\text{off}} + \tilde{\alpha}_i^{\text{on}})} \tag{24b}$$

In order to apply existing utilization-based schedulability tests, it is possible to set a constant activation time $C_{i,k} = C_i \leq C_i^*, \forall k$, and set periods $T_{i,k}$ according to the result of Theorem 3. In this case, taking into account equations (23) and (24a), the highest possible utilization can be expressed as follows.

$$U_i \doteq \sup_{k \in \mathbb{N}} U_{i,k} = \frac{C_i \tilde{\alpha}_i^{\mathrm{off}}}{C_i(\tilde{\alpha}_i^{\mathrm{off}} + \tilde{\alpha}_i^{\mathrm{on}}) - \epsilon_i^* - 2\delta_i^x} \qquad (25)$$

The utilization derived in (25) can be used to test the schedulability of the load set.

4 Considerations on results

To summarize, to compensate the effects of uncertainties the actual values of $T_{i,k}$ and $C_{i,k}$ are adapted depending to the gap between the measured value of the state variable $\hat{x}_i$ and its expected value $\bar{x}_i$. While there is the possibility to simultaneously change both period and activation time, a simpler solution is to keep constant one of the two parameters and change the other one.

The option to maintain a constant period T_i and to change the activation time $C_{i,k}$ makes the resulting system identical to the well known task model with strict periods. In this case, the longest possible activation time plays the role of the Worst Case Execution Time (WCET) in the traditional task model. It is straightforward to show that the longest possible activation time occurs when the measured value of the state variable is equal to the highest possible value. The corresponding utilization can be calculated using (24b), and such value can be used for the schedulability test.

The other option is to maintain a constant activation time C_i and to change the period $T_{i,k}$ at each request time. This option makes the resulting system working as the traditional sporadic task model, where periods represent the minimum time frame between two consecutive request times. In this case, the shortest possible period will occur when the measured value of the state variable is equal to the lowest possible value, and the utilization is expressed by (25). The option to dynamically change the period within an interval given by:

$$T_{i,k} \in \left[\frac{C_i}{U_i}, \; \frac{C_i(\tilde{\alpha}_i^{\mathrm{off}} + \tilde{\alpha}_i^{\mathrm{on}}) + \epsilon_i^* + 2\delta_i^x}{\tilde{\alpha}_i^{\mathrm{off}}} \right] \qquad (26)$$

suggests the possibility to apply techniques like the elastic scheduling [2] to the system model used in this paper. An interesting result could be to formally put into relationship the value of the elastic coefficient to physical system parameters. Depending on the system characteristics and requirements, the designer can choose the most convenient option.

5 Conclusion

This paper presented a modeling and control approach for power systems where the physical system is modeled as a set of real-time activities that can be scheduled by adapting traditional real-time scheduling algorithms. In particular, this paper discusses the application to the peak load minimization where electric loads having integrator dynamic are controlled under timing and physical constraints. To make the proposed methods applicable in realistic scenarios, the analysis is carried out considering physical parameters affected by unknown (but bounded) errors. A feedback control technique is provided to cope with modeling errors, and formal properties of the proposed method have been derived.

References

[1] S. K. Baruah, A. K. Mok, and L. E. Rosier. Preemptively scheduling hard-real-time sporadic tasks on one processor. In *Proc. of the* 11^{th} *IEEE Real-Time Systems Symposium*, pages 182–190, Dec. 1990.

[2] G. C. Buttazzo, G. Lipari, M. Caccamo, and L. Abeni. Elastic scheduling for flexible workload management. *IEEE Trans. on Computers*, 51(3):289–302, Mar. 2002.

[3] D. Callaway and I. Hiskens. Achieving controllability of electric loads. *Proceedings of the IEEE*, 99(1):184 –199, jan. 2011.

[4] A. I. Cohen. An optimization method for load management scheduling. *IEEE Transactions on Power Systems*, 3(2), May 1988.

[5] 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)*, pages 41–49, November 2010.

[6] T. Facchinetti, E. Bini, and M. Bertogna. Reducing the peak power through real-time scheduling techniques in cyber-physical energy systems. In *Proc. of the First Int. Workshop on Energy Aware Design and Analysis of Cyber Physical Systems (WEA-CPS)*, April 2010.

[7] T. Facchinetti and M. L. Della Vedova. Real-time modeling for direct load control in cyber-physical power systems. *Industrial Informatics, IEEE Transactions on*, 7(4):689 –698, nov. 2011.

[8] T. Facchinetti and M. L. D. Vedova. Real-time modeling and control of a cyber-physical energy system. In *Proc. of the First Int. Workshop on Energy Aware Design and Analysis of Cyber Physical Systems (WEA-CPS)*, April 2010.

[9] H. Hermanns and H. Wiechmann. Future design challenges for electric energy supply. In *Proceedings of the IEEE Conference on Emerging Technologies and Factory Automation (ETFA)*, September 2009.

[10] F. Kupzog and C. Roesener. A closer look on load management. In *Industrial Informatics, 2007 5th IEEE Int. Conf. on*, volume 2, pages 1151 –1156, june 2007.

[11] J. Lunze and F. Lamnabhi-Lagarrigue, editors. *Handbook of Hybrid Systems Control: Handbook of Hybrid Systems Control*. Cambridge University Press, 2009.

[12] P. Palensky and D. Dietrich. Demand side management: Demand response, intelligent energy systems, and smart loads. *Industrial Informatics, IEEE Transactions on*, 7(3):381 –388, aug. 2011.

[13] J. R. Taylor. *An Introduction to Error Analysis, The Study of Uncertainties in Physical Measurements*. University Science Books, 1997.