On real-time physical systems

Marco L. Della Vedova, Michele Ruggieri, Tulio Facchinetti

On real-time physical systems

Marco L. Della Vedova
University of Pavia
Pavia, Italy
marco.dellavedova@unipv.it

Michele Ruggeri
University of Pavia
Pavia, Italy
michele.ruggeri@gmail.com

Tullio Facchinetti
University of Pavia
Pavia, Italy
tullio.facchinetti@unipv.it

Abstract

This paper introduces a class of real-time systems denoted as Real-Time Physical Systems (RTPS), in which a physical quantity of interest is associated with a real-time resource. The physical quantity behavior is determined by scheduling events generated by a real-time scheduling algorithm. RTPS systems aim to generalize some existing models used in real-time computing systems, namely power-aware and temperature-aware systems. Moreover, they have been conceived to put a bridge across real-time systems and the rapidly growing research field of Cyber-Physical Systems.

In this paper we focus on a specific physical system where a state variable changes with an exponential decay behavior and the associated real-time resource must be scheduled in order to bound the value of the state variable within the desired working range. The aim is to determine the relationship between those physical and real-time parameters. For this purpose interesting properties are highlighted and relevant results are derived regarding the considered system model.

1 Introduction

The problem of scheduling real-time resources while achieving some constraints on a related physical value is being addressed in several kind of applications in these last years. The notion of *Real-Time Physical System* (RTPS) is introduced in this paper to indicate a class of systems in which a physical value exists and its variation is determined by the schedule generated using a real-time scheduling policy. It is worth to outline that the physical value variation is not influenced by the computation performed by the scheduled task. The only events influencing the physical value behavior are determined by the sequence of scheduling decisions generated by the scheduler. In fact, as described later in this section, some special cases of RTPS do not necessarily require the existence of real-time computing tasks, since scheduled real-time resources are not related with computing systems. Figure 1 shows an example of a RTPS in which the physical value $x(t)$ decreases exponentially when the real-time resource is scheduled for execution; otherwise it increases. Notice that the physical value behavior is influenced only by the switching events between execution and idle time.

One of the first examples of RTPS is represented by power-aware real-time systems [1]. In such systems, scheduling policies are proposed to minimize the overall energy consumed by the computing system, while respecting the timing constraints imposed to processing tasks. In this case, the physical value is represented by the total power consumed by one (in uni-processor systems) or more (in multi-processor systems) tasks at any time instant. For this purpose, an energy consumption is associated to each task, and enabling technologies such as Dynamic Voltage Scaling (DVS) and Dynamic Power Management (DPM) are leveraged to optimize the consumed energy while producing a feasible schedule [10, 8]. While DVS saves energy by reducing the system working frequency (clock) and thus running at a lower voltage, DPM saves energy by putting system's components into low power states when they are not necessary for system operations. In recent works both techniques are jointly used to improve the energy saving [3].

More recent examples of RTPS are the so-called temperature-aware real-time systems. In those systems, the physical value is the processor's temperature. Real-time scheduling algorithms are developed to generate a task's schedule such that an upper bound on the processor's temperature is guaranteed. The goal is limiting CPU's overheating and consequently reducing the energy spent by CPU's fans or air conditioning systems. Technical issues produced by high temperatures are becoming extremely relevant in data-centers [15]. On the other hand, working at lower temperatures has the effect of extending the system lifetime. Temperature-aware techniques are usually combined with DVS scheduling approaches [2, 17]. In other works, feedback techniques are used to achieve the physical constraints on the temperature [6].

More recently, the study of Cyber-Physical Systems (CPS) has emerged as a relevant multi-disciplinary research topic [16]. Cyber-Physical Systems are characterized by a tight integration between the computational resource and the physical process to monitor and control. Real-time scheduling is, among other research fields, strictly related to CPS due to its inherent applicability for guaranteeing timing constraints involved in CPS. A growing interest in RTPS related with CPS may thus be expected. A special case of CPS is represented by the so-called Cyber-Physical Energy Systems (CPES) [14]. In CPES, a network of electric devices is the physical system to be monitored and controlled. Embedded systems are integrated to gather information about the most important electric parameters, such as voltage, current, phases, consumed energy and power. Environmental parameters, as temperature, humidity and pressure, are also relevant for system characterization. The acquired data are combined and processed to generate suitable control commands for the electric devices, in order to achieve the desired application goal. And such goals include, or introduce constraints, on power and energy usage. In the literature, CPES have been modeled as dynamic systems [9] or, in a few recent works, using parameters derived from the real-time scheduling domain [4, 5]. In particular, the latter approach has addressed the modeling of networks of electric devices. For this purpose, an analogy is established between real-time computing systems and electrical systems, which represent the physical background of a Cyber-Physical Energy System. Shortly, electric devices are seen as computing tasks and are scheduled according to their timing constraints, which in turn are set to achieve some properties of the physical system. In the referred papers the goal was to reduce the peak load of power consumption. In particular, in [5] the schedule was associated with a linearly varying state variable (i.e., the physical value) to be kept within the desired range.

The present paper proposes an approach and modeling techniques to define a RTPS that have some affinities with *hybrid systems*, a popular research field in the area of control engineering [13]. Hybrid systems study the control issues when both continuous and discrete systems are involved in the control loop, where the presence of discrete systems is motivated by the use of digital computing systems to execute the control strategy. In particular, the so-called *switched hybrid systems* address the control of systems where the state variable's behavior is allowed to switch among different dynamic behaviors (i.e., different sets of differential equations are associated to each behavior) [11]. However, switched hybrid systems are studied from the control perspective, to determine properties such as system stability, observability, etc. [7] or to find suitable control techniques such as optimal control strategies [18]. For this reason they do not address the relationship with real-time scheduling issues that may be involved in the switching signal control.

1.1 Paper organization

After an introduction to the concept of real-time physical systems made in Section 1, the contribution of this paper is stated in Section 2. The system model is presented in Section 3, including the model of the physical system and real-time parameters. Section 4 provides the results and describes the properties of our model, and puts into relationship physical and real-time parameters. In Section 5 an example of application of the proposed techniques is shown, while conclusions and future works are stated in Section 6. Proofs of theorems are in Appendix.

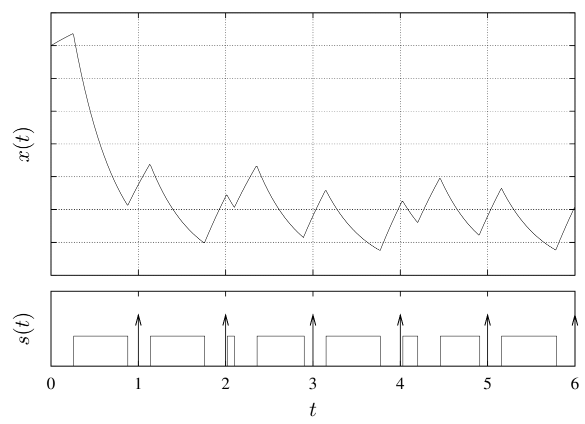

Figure 1. Example of real-time physical system in which the $x(t)$ physical value decreases exponentially when the resource is scheduled for execution, and it increases otherwise.

2 Contributions

This paper presents and discusses a special case of RTPS in which the state variable changes following an exponential behavior. The goal is to schedule resources using traditional real-time scheduling algorithms while achieving some given constraints on the state variable. In particular, we provide results regarding the working range of the state variable, i.e., we are interested in bounding the physical value within a given range. For this purpose, we relate real-time parameters of scheduled resources to the bounds of the working range.

Despite we focus on a specific RTPS, the proposed model and results are general enough to model several real physical systems. In fact, the exponential behavior of the proposed RTPS arises from a modelization based on affine time-invariant dynamic systems, which can describe many existing physical systems. For this purpose, a differential equation is associated to each real-time resource which determine the physical value behavior.

3 System model

The considered system is composed by a set

$$\Lambda = \{\lambda_1, \ldots, \lambda_n\} \tag{1}$$

of n resources that can be turned on and off. A resource is said to be *active* when it is turned on, *inactive* otherwise. The resource activity is controlled by a *resource scheduler* that decides when each resource is activated/deactivated. The activation of each resource is independent of other ones (i.e., no precedence or other kind of constraints among resources are considered). Formally, the scheduler assigns to each resource λ_i a schedule that is modeled by the activation function $s_i(t)$:

$$s_i(t) = \begin{cases} 1 & \lambda_i \text{ is active at } t \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

One *physical system* Γ_i is associated to each resource λ_i. The physical system is composed by a *state variable* x_i (i.e., the physical quantity of interest), which evolves as a function of the activity of the resource λ_i. A *dynamic system* Φ_i determines the behavior of the state variable, which is defined by the following equation:

$$\Phi_i : \frac{dx_i(t)}{dt} = k_i^{\text{off}}\left(h_i^{\text{off}} - x_i(t)\right) + \\ + k_i^{\text{on}}\left(h_i^{\text{on}} - x_i(t)\right) s_i(t) \tag{3}$$

with $k_i^{\text{off}} > 0$ and $k_i^{\text{on}} > 0$.

The state variable behavior is therefore made by two components, describing two exponential decays. The former is when the resource is inactive, and the state variable tends exponentially to h_i^{off}. The latter happens when the resource is active. In this case the state variable tends exponentially to a value determined by a combination of all the parameters, as detailed in Equation 10. It is worth to note that the former component is always present and it does not depend on the resource activity (i.e., it is not multiplied by $s_i(t)$). This behavior models a background "free component" of the physical process that is not affected by resource activity. On the other hand, the same state variable behavior could be equivalently formalized as in Equation 4

$$\Phi_i : \frac{dx_i(t)}{dt} = \tilde{k}_i^{\text{off}}\left(\tilde{h}_i^{\text{off}} - x_i(t)\right)\overline{s_i}(t) + \\ + \tilde{k}_i^{\text{on}}\left(\tilde{h}_i^{\text{on}} - x_i(t)\right) s_i(t) \tag{4}$$

where $\overline{s_i}(t)$ is the logic negation (Boolean *not*) of the $s_i(t)$ signal, while all parameters in Equation 4 can be expressed as a proper function of parameters in Equation 3.

Equation 3 will referred in the reminder since it better reflects the behavior physical system.

An example of state variable's behavior determined by Equation 3 (or Equation 4 as well) is depicted in Figure 1.

Each physical system is characterized by a set of *constraints* Ψ_i on the state variable. In the case considered in this paper, the set of constraints is specified by the following inequalities:

$$\Psi_i : \begin{cases} x_i(t) \leq x_i^{\max} & \forall t > t_i^\star \\ x_i(t) \geq x_i^{\min} & \forall t > t_i^\star \end{cases} \tag{5}$$

In other words, the state variable x_i is required to be bounded in the range $[x_i^{\min}, x_i^{\max}]$ after a certain time instant $t_i^\star$. It is worth to note that it may not be possible to achieve physical constraints for every time instant t. In particular, when the state variable lays outside bounds at $t = 0$, there are no chances to meet physical constraints until the state variable is led within its bounds. Our analysis considers such a possibility. For this reason, time $t_i^\star$ is the time required to lead the state variable within its bounds.

Therefore, a physical system is defined by the pair $\Gamma_i \equiv (\Phi_i, \Psi_i)$, i.e., one dynamic system and a set of constraints.

It is noteworthy that the exponential model defined by Equation 3 may represent a good approximation for many physical systems. In fact, many physical phenomena can be described or approximated by first-order differential equations whose solution have exponential decay.

An icebox is an example of physical system that is suitably described by the proposed model: internal temperature represents the state variable, which requires to be maintained within a given range (e.g., between -5°C and -2°C).

3.1 Real-time modeling

Considering the parameters used to model a traditional real-time computing task, we use the pair $\{T_i, C_i\}$ to define a resource λ_i, where

- T_i is the time frame between two consecutive request times (as in the periodic task model for real-time computing tasks [12]); the k-th request for activating the resource λ_i happens at time $r_{i,k}$ (*request time*), where $r_{i,k} = kT_i$, $k \in \mathbb{N}$; we refer to T_i as the *period* of the i-th resource;

- $C_i(\leq T_i)$ represents the activation time duration of λ_i within each period T_i;

The utilization of λ_i is defined as

$$U_i = \frac{C_i}{T_i} \tag{6}$$

while the total utilization of the resource set is

$$U^{\text{tot}} = \sum_{i=1}^{n} U_i \tag{7}$$

Given the above model, the definition of valid schedule is introduced as follow.

Definition 1. A schedule $\mathcal{S}$ is said to be *valid* if it assigns to each resource λ_i an amount of activity time equal to C_i between two consecutive request times. Formally,

$$\forall \lambda_i, \forall k \quad \int_{r_{i,k}}^{r_{i,k+1}} s_i(t)\, dt = C_i \tag{8}$$

A valid schedule can be generated by a real-time scheduling algorithm as Earliest Deadline First (EDF) or Rate Monotonic (RM) [12] when applied to a feasible set of resources, i.e., the specific schedulability test, applied to the given resource set, is passed for the considered algorithm.

For the sake of convenience, two specific activation functions are defined, since they will be often referred in following sections.

Definition 2. The activation function $s_i^{\Uparrow}(t) : [t_a, t_b] \rightarrow \{0, 1\}$ is defined by the following properties:

- has utilization U_i over the time interval $[t_a, t_b]$
- it holds

$$s_i^{\Uparrow}(t) = \begin{cases} 1, & \text{if } t \in [t_a + U(t_b - t_a), t_b] \\ 0, & \text{otherwise} \end{cases}$$

Definition 3. The activation function $s_i^{\Downarrow}(t) : [t_a, t_b] \rightarrow \{0, 1\}$ is defined by the following properties:

- has utilization U_i over the time interval $[t_a, t_b]$
- it holds

$$s_i^{\Downarrow}(t) = \begin{cases} 1, & \text{if } t \in [t_a, t_a + U(t_b - t_a)] \\ 0, & \text{otherwise} \end{cases}$$

The meaning of definitions 2 and 3 is that the whole activation time of $s_i^{\Uparrow}(t)$ (respectively, $s_i^{\Downarrow}(t)$) is temporally located at the beginning (at the end) of the considered time interval. In Figure 2 both functions are shown.

3.2 Expected results

A feasible RTPS is defined as a system characterized by a valid schedule and whose physical constraints are satisfied. In other words:

$$\begin{cases} \mathcal{S} \text{ is a valid schedule} \\ \Psi_i \text{ are satisfied} \end{cases}$$

Given the considered system model regarding both the physical and real-time modelization, the goal of this paper is to derive the relationship between physical and real-time parameters for obtaining a feasible RTPS. While the scheduling of resources is performed by some available real-time scheduling algorithm, we are interested in finding suitable values for real-time parameters to satisfy physical constraints Ψ_i.

For the sake of clarity, subscripts will be dropped from notation when not required.

4 Properties and results

This section provides some interesting properties and introduces relevant results regarding the relationship between real-time parameters and physical values.

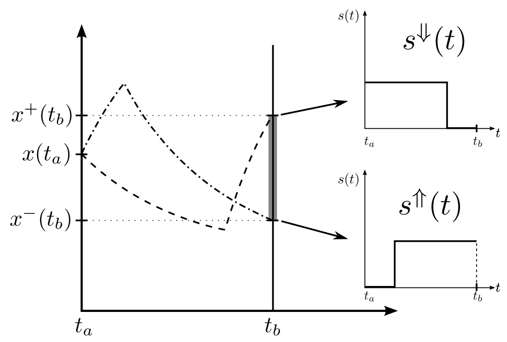

Figure 2. The worst case conditions that determine the maximum and minimum values for $x(t_b)$.

Since we are interested in the evolution of the state variable in the domain of time, we solve Equation 3 obtaining:

$$x(t) = \begin{cases} A - (A - x(0))\, e^{-\alpha t} & \text{if } s(t) = 1 \\ B - (B - x(0))\, e^{-\beta t} & \text{if } s(t) = 0 \end{cases} \quad (9)$$

where

$$\begin{aligned} A &= (k^{\text{on}} h^{\text{on}} + k^{\text{off}} h^{\text{off}})/(k^{\text{on}} + k^{\text{off}}) \\ \alpha &= k^{\text{on}} + k^{\text{off}} \\ B &= h^{\text{off}} \\ \beta &= k^{\text{off}} \end{aligned} \quad (10)$$

The state variable $x(t)$ has an exponential decay behavior characterized by an asymptote A and a time constant α when the resource λ is active; conversely, it has an exponential decay behavior determined by an asymptote B and a time constant β when the resource is inactive.

Theorem 1. *In any time frame $R = [t_a, t_b]$ and for any schedule $s(t)$ in R such that the utilization is U, for any given value $x(t_a)$, the following properties hold for $x(t_b)$:*

1. *the maximum value of $x(t_b)$, i.e. $x^+(t_b)$, is obtained when $s(t) \equiv s^{\Downarrow}(t)$ in R*
2. *the minimum value of $x(t_b)$, i.e. $x^-(t_b)$, is obtained when $s(t) \equiv s^{\Uparrow}(t)$ in R*

Theorem 1 states that, when considering a generic time interval R and an activation function having a given utilization U within R, for any value assumed by the state variable $x(t)$ at $t = t_a$, the worst case combination of schedule activations/deactivations that generates the maximum and minimum values at $t = t_b$ occurs when the activation function is active, respectively, at the beginning and at the end of the time interval. Figure 2 shows an example of this situation.

Thanks to the above properties, it is possible to easily determine the bounds of the state variable when $t = t_b$.

Corollary 1. *In any time frame $R = [t_a, t_b]$ and for any schedule $s(t)$ in R such that the utilization is U, for any given value $x(t_a)$, it holds:*

1. *if $s(t) \equiv s^{\Downarrow}(t)$ in R*

$$x^+(t_b) = B + (A - B)\, e^{-\beta(1-U)(t_b - t_a)} + \\ + (x(t_a) - A)\, e^{-(\alpha U + \beta(1-U))(t_b - t_a)} \tag{11}$$

2. *if $s(t) \equiv s^{\Uparrow}(t)$ in R*

$$x^-(t_b) = A + (B - A)\, e^{-\alpha U(t_b - t_a)} + \\ + (x(t_a) - B)\, e^{-(\alpha U + \beta(1-U))(t_b - t_a)} \tag{12}$$

The results of Corollary 1 can be directly extended to periodic real-time schedules, as specified by Corollary 2.

Corollary 2. *The properties of Theorem 1 and Corollary 1 hold for any valid schedule in the range $[r_k, r_{k+1}]$ for every k. In particular:*

1. *if $s(t) \equiv s^{\Downarrow}(t)$ in $[r_k, r_{k+1}]$*

$$x^+(r_{k+1}) = B + (A - B)\, e^{-\beta(1-U)T} + \\ + (x(r_k) - A)\, e^{-(\alpha U + \beta(1-U))T} \tag{13}$$

2. *if $s(t) \equiv s^{\Uparrow}(t)$ in $[r_k, r_{k+1}]$*

$$x^-(r_{k+1}) = A + (B - A)\, e^{-\alpha UT} + \\ + (x(r_k) - B)\, e^{-(\alpha U + \beta(1-U))T} \tag{14}$$

Corollary 2 states that, for any value $x(r_k)$ the value of $x(r_{k+1})$ is bounded in a range whose limits are function of physical parameters A, B, α and β, and real-time parameters U and T. The exact value of $x(r_{k+1})$ depends on the values assumed by the activation function within the range $[r_k, r_{k+1}]$.

So far, a result regarding schedules within generic time intervals (Theorem 1) has been proven, and then extended to periodic real-time schedules. These results are now used to determine a bound on the value of the state variable in correspondence with request times. For this purpose, the following definition is introduced:

Definition 4. The succession of values of the state variable in correspondence of request times is $S_k = \{x(t)\}$ such that $t = r_k$ for $k = 0, 1, 2, \ldots$.

Theorem 2. *For any valid activation function $s(t)$ the succession S_k for $k \geq k^*$, is bounded between $\tilde{x}^{\text{inf}}$ and $\tilde{x}^{\text{sup}}$, which are expressed by the following equations:*

$$\tilde{x}^{\text{inf}} = \frac{A + (B - A)\, e^{-\alpha UT} - Be^{-(\alpha U + \beta(1-U))T}}{1 - e^{-(\alpha U + \beta(1-U))T}} \tag{15}$$

$$\tilde{x}^{\text{sup}} = \frac{B + (A - B)\, e^{-\beta(1-U)T} - Ae^{-(\alpha U + \beta(1-U))T}}{1 - e^{-(\alpha U + \beta(1-U))T}} \tag{16}$$

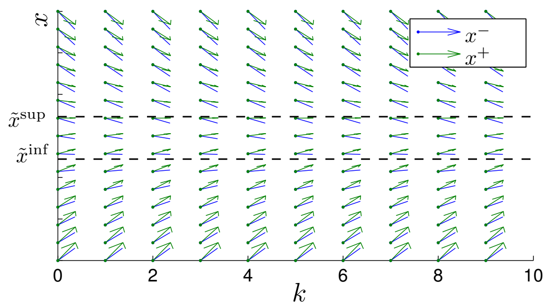

Figure 3. Attraction range for the succession $S_k : x(r_k)$.

Theorem 2 provides a bound on the S_k succession. This means that, after a large enough time instant, the values of $x(r_k)$ remain within the corresponding range, that will be referred as *attraction range*, as shown in Figure 3. Clearly, the result is valid for large enough t, since the state variable behavior may start outside the attraction range. Therefore a suitable time interval is required to converge within the attraction range.

However, in this paper we are interested in bounding the state variable behavior, not only in correspondence with request times, but for every t. On the other hand, the result provided by Theorem 2 allows to study the worst case behavior of the value of the state variable within one period, instead of having to consider the system lifetime for every t. In fact, given the worst case conditions of state variable values in correspondence with request times, it is possible to calculate the worst case conditions within period, and such worst case will hold for every t.

Theorem 3. *For any valid activation function $s(t)$ it holds*

$$x^{\text{inf}} \leq x(t) \leq x^{\text{sup}},\ \forall t > t^*$$

where:

$$x^{\text{inf}} = A - \left(A - \tilde{x}^{\text{inf}}\right) e^{-\alpha UT} \tag{17}$$

$$x^{\text{sup}} = B - \left(B - \tilde{x}^{\text{sup}}\right) e^{-\beta(1-U)T} \tag{18}$$

Theorem 3 provides the bounds for the state variable variation range. It is worth to note that bounding values x^{inf} and x^{sup} are expressed for clarity as function of $\tilde{x}^{\text{int}}$ and $\tilde{x}^{\text{sup}}$ respectively. Therefore, bounds are function of physical and real-time parameters only, and represent the relationship between such parameters. Again, the result holds for large enough t, i.e., we allow a state variable initially set outside the attraction range to evolve towards such range. On the other hand, if the initial value of the state variable is within the feasible range, then Corollary 3 trivially holds.

Corollary 3. *If $\tilde{x}^{\text{inf}} \leq x(0) \leq \tilde{x}^{\text{sup}}$ then $t^* = 0$.*

Finally, it is of interest to notice that the state variable has an asymptotic behavior for $T \to 0$, as outlined by Fact 1.

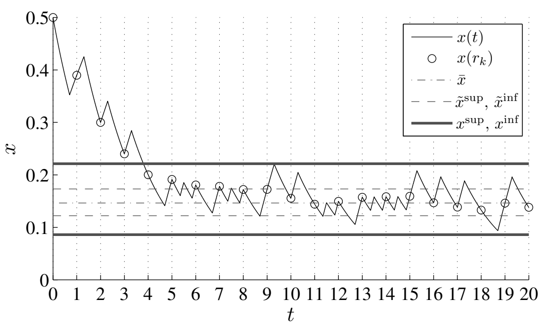

Figure 4. Example of state variable behavior and corresponding bounds.

Fact 1. *For any valid schedule, it holds:*

$$\bar{x} = \lim_{t\to\infty,\, T\to 0} x(t) = \frac{A\alpha U + B\beta(1-U)}{\alpha U + \beta(1-U)} \tag{19}$$

Figure 4 shows an example of state variable behavior. The succession S_k is depicted (indicated by circles) as well as its bounding range. The asymptotic value for $T \to 0$ is also depicted.

4.1 Feasibility region in the $U - T$ space

According to the notion of feasibility given in Section 3.2, we are interested to find real-time parameters, i.e. U and T, which allow to satisfy the physical constraints Ψ. For this purpose, we introduce the following definition.

Definition 5. The *feasibility region* Ω is a region in the $U - T$ plane composed by all and only pairs (U, T) such that the system is feasible.

Given a pair (U, T), it is possible to check whether it belongs to the feasibility region Ω by checking that, for the values $x^{\inf}$ and $x^{\sup}$ obtained from equations 17 and 18, the following inequalities are satisfied:

$$x^{\inf} \geq x^{\min} \quad \text{and} \quad x^{\sup} \leq x^{\max} \tag{20}$$

In general, there exists a set of pairs $(U, T) \in \Omega$. Unfortunately, it is not possible to find the values of U and T from equations 17 and 18 in closed form. Therefore, pairs $(U, T) \in \Omega$ must be found using numerical techniques.

On the other hand, the result from Fact 1 can be used to find the bounds of Ω on the U axis. In fact, it is possible to find a range $[U^{\min}, U^{\max}]$ in which a pair $(U, T) \in \Omega$ can be found. In other words, conditions

$$U^{\min} \leq U \leq U^{\max} \tag{21}$$

are necessary conditions for the system feasibility. Range bounds can be determined by imposing $\bar{x} = x^{\max}$ and $\bar{x} = x^{\min}$, respectively, in Equation 19, leading to the following equations:

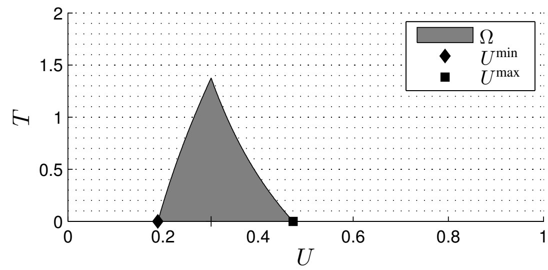

Figure 5. Example of feasible region in the $U - T$ space.

$$U^{\max} = \frac{\beta(B - x^{\max})}{(\alpha-\beta)x^{\max} - (A\alpha - B\beta)} \tag{22}$$

$$U^{\min} = \frac{\beta(B - x^{\min})}{(\alpha-\beta)x^{\min} - (A\alpha - B\beta)} \tag{23}$$

An example of feasibility region is depicted in Figure 5. The plot was obtained by using the following physical parameters: $A = 0$, $B = 1$, $\alpha = 1$, $\beta = 0.1$, $x^{\min} = 0.1$, $x^{\max} = 0.3$.

5 Example of application

This section provides an example of application of the modeling, analysis and control techniques presented in previous sections. The goal is to concurrently schedule a set of three real-time resources such that physical constraints are satisfied.

The proposed physical system associated with each resource can represent a fridge where the state variable x_i is the internal temperature (assumed to be uniform) that must be kept within a working desired range. Physical parameters of each fridge are listed in Table 1.

Values A_i and B_i represent asymptotic values which the state variable would tend to if the resource would be kept always on (A_i) or always off (B_i). In the considered physical system made by fridges, these values represent refrigerant fluid and environmental temperatures respectively; both are constant in our simple example. In particular, environmental temperature value is 20 Celsius degree for all fridges and refrigerant fluid temperatures are -10°C for λ_1 and λ_2, and -30°C, for λ_3. Such values are realistic for common refrigerators.

Values of the decay constants α_i and β_i are related with physical properties such as thermal capacities and heat transfer coefficients, which are in turn related with building materials and size. The greater the decay constant value, the faster fridge temperature dynamics (which means less insulation and/or smaller size of the fridge). On the other hand, the heat transfer coefficient between the fluid refrigerator and the fridge content is related to α_i, while the heat transfer coefficient between the environment and the fridge indoor is related to β_i.

Table 1. Values of physical parameters used for the simulation

i	A	α	B	β	$x^{\min}$	$x^{\max}$	$x(0)$
1	-10	0.10	20	0.04	-4	-1	-1
2	-10	0.15	20	0.03	1	5	2
3	-30	0.20	20	0.03	-15	-10	-12

The refrigerator temperature working ranges are bounded by $x^{\min}$ and $x^{\max}$, which define the minimum and maximum working temperature respectively. Finally, the initial internal temperature ($t = 0$) is $x(0)$.

In order to associate proper real-time parameter values to each physical system, i.e. utilization U and period T, the range $(U^{\min}, U^{\max})$ has been calculated using Equations 22 and 23. The value of U was chosen in the middle of the range, in this example. Regarding the subsequent determination of the value of T, it has been accomplished by trial, taking into account the constraint described by Equation 20 (which involves Equations 15, 16, 17 and 18). Table 2 lists the calculated real-time parameters.

Using the selected parameters, the total utilization is less than 1 ($U^{\text{tot}} = 0.98$). Therefore, it is possible to concurrently schedule the resources using the EDF scheduling algorithm, guaranteeing that *at most one resource* is active at any given time instant. The corresponding schedule is depicted in Figure 6. While the above three graphs show state variable behaviors, the bottom graph represents the resource schedule.

By using the modeling, analysis and control techniques based on real-time parameters and scheduling algorithms presented in this paper and applied to the example application provided in this section, it is possible to strongly limit the simultaneous activation of multiple resources, even if total utilization is greater than 1. In this case it is possible to use real-time scheduling techniques proposed for multiprocessor architectures (see [4] for further information). Limiting simultaneous activations is a key feature of the proposed approach, that can be applied to - for example - the power optimization of energy systems to reduce the peak of electric demand from the energy distribution grid. In such scenario, Real-Time Physical Systems are used to model electric drives that actuate some devices subject to physical constraints. Refrigerators, air conditioning systems, and battery charge/discharge systems are some few examples. In those applications, electric loads must be scheduled to limit the peak load of electric power consumption (which usually has significant impact on system costs) while achieving per-load constraints. The reader can refer to [4, 5] for details of the application of Real-Time Physical Systems to power optimization in Cyber-Physical Energy Systems.

Table 2. Values of real-time parameters used for the simulation

i	T	U	$U^{\min}$	$U^{\max}$
1	2.0	0.55	0.48	0.62
2	3.0	0.21	0.17	0.26
3	1.5	0.22	0.18	0.26

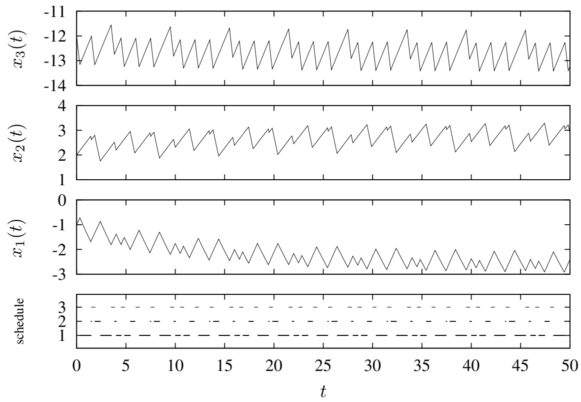

Figure 6. An example of application of the proposed technique to the schedule of 3 **real-time resources.**

6 Conclusions and future works

This paper has introduced the class of real-time physical systems as a type of real-time systems in which scheduling decisions affect the behavior of a tightly related physical value. This class of systems includes existing well-known real-time system models (power-aware and temperature-aware systems), while applying also to newer research topics such as cyber-physical systems.

A particular case of RTPS has been studied, modeled and analyzed in this paper. The focus has been put on the relationship between physical and real-time parameters such that the constraints of the physical value behavior are satisfied.

Future works will address the introduction of feedback techniques on the proposed control strategy, the modelization of errors, the extension to systems with deadlines less than periods, and further inspections of physical properties as a function of real-time parameters.

References

[1] Hakan Aydin, Rami Melhem, Daniel Mossé, and Pedro Mejía-Alvarez. Power-aware scheduling for periodic real-time tasks. *IEEE Transactions on Computers*, 53(5):584–600, May 2004.

[2] Jian-Jia Chen, Shengquan Wang, and Lothar Thiele. Proactive speed scheduling for frame-based real-

time tasks under thermal constraints. In *Proceedings of the IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, April 2009.

[3] Vinay Devadas and Hakan Aydin. DFR-EDF: A unified energy management framework for real-time systems. In *IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, pages 121–130, Los Alamitos, CA, USA, 2010. IEEE Computer Society.

[4] Tullio Facchinetti, Enrico Bini, and Marko 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.

[5] Tullio Facchinetti and Marco 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.

[6] Yong Fu, Nicholas Kottenstette, Yingming Chen, Chenyang Lu, Xenofon D. Koutsoukos, and Hongan Wang. Feedback thermal control for real-time systems. In *Real-Time and Embedded Technology and Applications Symposium (RTAS), 2010 16th IEEE*, pages 111 –120, 12-15 2010.

[7] Joao P. Hespanha. Stability of switched systems with average dwell-time. In *In Proc. 38th IEEE Conf. on Decision and Control*, pages 2655–2660, 1999.

[8] Inki Hong, Gang Qu, Miodrag Potkonjak, and Mani B. Srivastava. Synthesis techniques for low-power hard real-time systems on variable voltage processors. In *Proceedings of the* 19^{th} *IEEE Real-Time Systems Symposium*, pages 178–187, Madrid, Spain, December 1998.

[9] M. D. Ilic, Le Xie, U. A. Khan, and J. M. F. Moura. Modeling future cyber-physical energy systems. In *IEEE Power and Energy Society General Meeting - Conversion and Delivery of Electrical Energy in the 21st Century*, pages 1–9, July 2008.

[10] Tohru Ishihara and Hiroto Yasuura. Voltage scheduling problem for dynamically variable voltage processors. In *Proceedings of the International Symposium on Low Power Electronics and Design*, pages 197–202, Monterey, CA U.S.A., August 1998.

[11] Daniel Liberzon and A. Stephen Morse. Basic problems in stability and design of switched systems. *IEEE Control Systems Magazine*, 19:59–70, 1999.

[12] Chung Laung Liu and James W. Layland. Scheduling algorithms for multiprogramming in a hard real-time environment. *Journal of the Association for Computing Machinery*, 20(1):46–61, January 1973.

[13] Jan Lunze and Francoise Lamnabhi-Lagarrigue, editors. *Handbook of Hybrid Systems Control: Handbook of Hybrid Systems Control*. Cambridge University Press, 2009.

[14] S. T. Morris, A. Srivastava, B. Reaves, S. Abdelwahed K. Pavurapu, R. Vaughn, W. McGrew, and Y. Dandass. Engineering future cyber-physical energy systems: Challenges, research needs, and roadmap. In *Proceedings of the IEEE North American Power Symposium*, October 2009.

[15] N. Rasmussen. Calculating total cooling requirements for data centers. In *American Power Conversion, white paper number 25*, 2007.

[16] Lui Sha, Sathish Gopalakrishnan, Xue Liu, and Qixin Wang. Cyber-physical systems: A new frontier. In *IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC)*, pages 1–9, June 2008.

[17] Shengquan Wang and Riccardo Bettati. Reactive speed control in temperature-constrained real-time systems. *Real-Time Systems*, 39(1–3):73–95, August 2008.

[18] Xuping Xu and Panos J. Antsaklis. Optimal control of switched systems based on parameterization of the switching instants. *IEEE Transactions on Automatic Control*, 49(1), January 2004.

A Appendix: Detailed Proofs

Proof of Theorem 1

Proof. We limit our proof to the first case, since the second one can be proven with similar arguments.

We prove this property for a simple schedule, and we will extend by induction the considerations to generic schedules. Let's consider to divide the $[t_1, t_5]$ range in 4 sub-ranges $R_1 = [t_1, t_2]$, $R_2 = [t_2, t_3]$, $R_3 = [t_3, t_4]$ and $R_4 = [t_4, t_5]$, being $t_i < t_j$ for all $i < j$. Suppose the schedule switches between active and inactive state at each range boundary. An example of state variable behavior generated by this case is depicted by the solid line in Figure 7. According to Equations 9 the following equations hold:

$$x_2 = x(t_2) = A - (A - x_1)e^{-\alpha(t_2 - t_1)} \quad (24)$$

$$x_3 = x(t_3) = B - (B - x_2)e^{-\beta(t_3 - t_2)} \quad (25)$$

$$x_4 = x(t_4) = A - (A - x_3)e^{-\alpha(t_4 - t_3)} \quad (26)$$

$$x_5 = x(t_5) = B - (B - x_4)e^{-\beta(t_5 - t_4)} \quad (27)$$

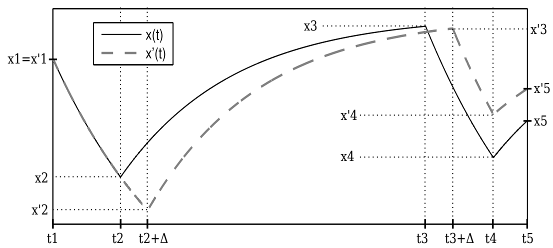

Figure 7. The state variable behavior and related parameters when a slice Δ is moved in the schedule from the interval $[t_3, t_4]$ to $[t_1, t_2]$.

being $x_1 = x(t_1)$.

Now we obtain a second schedule by moving a time slice Δ, being $0 < \Delta \leq (t_4 - t_3)$ of active time from sub-range $[t_3, t_4]$ to $[t_1, t_2]$. Notice that the utilization within $[t_1, t_5]$ is unchanged. We obtain a new set of sub-ranges which are $R'_1 = [t_1, t_2 + \Delta]$, $R'_2 = [t_2 + \Delta, t_3 + \Delta]$, $R'_3 = [t_3 + \Delta, t_4]$ and $R'_4 = [t_4, t_5]$, and the corresponding values at sub-ranges boundaries:

$$x'_2 = x(t_2 + \Delta) = A - (A - x_1)e^{-\alpha(t_2 - t_1)}e^{-\alpha\Delta} \quad (28)$$

$$x'_3 = x(t_3 + \Delta) = B - (B - x_2)e^{-\beta(t_3 - t_2)} \quad (29)$$

$$x'_4 = x(t_4) = A - (A - x_3)e^{-\alpha(t_4 - t_3)}e^{\alpha\Delta} \quad (30)$$

$$x'_5 = x(t_5) = B - (B - x_4)e^{-\beta(t_5 - t_4)} \quad (31)$$

The state variable behavior generated in this case is depicted by the dotted line in Figure 7.

We show that the schedule having utilization equal to U such that the active time is located at the beginning of the time frame (i.e., a schedule in the form of $s^{\Downarrow}(t)$) will generate the maximum possible value of x'_5.

For this purpose, we prove that $x'_5 > x_5$ for any value of Δ. Therefore, since we are considering generic sub-ranges, we will have proven that every time a slice is taken from the interval $[t_3, t_4]$ and it is moved to $[t_1, t_2]$, the value of x'_5 can only increase. Thus a schedule in the form of $s^{\Downarrow}(t)$ generates the highest possible value for $x(t_5)$.

From (31) and (27) it holds

$$x'_5 - x_5 = (x'_4 - x_4)e^{-\beta(t_5 - t_4)} \quad (32)$$

Since $e^{-\beta(t_5 - t_4)} > 0$, then it holds $x'_5 > x_5$ if and only if $(x'_4 - x_4) > 0$. From (30) and (26) we obtain

$$x'_4 - x_4 = e^{-\alpha(t_4 - t_3)}((A - x_3) - (A - x'_3)e^{\alpha\Delta}) \quad (33)$$

Equations 25 and 29 can be put within Equation 32, then Equations 24 and 28 can be inserted in the newly obtained equation. Then, after simple calculations, the following equation can be written

$$x'_4 - x_4 = e^{-\alpha(t_4 - t_3)}(B - A)(1 - e^{-\beta(t_3 - t_2)})(e^{\alpha\Delta} - 1)$$

in which each term is strictly greater than zero (if $\Delta > 0$), thus proving that Equation 32 is always positive.

The proof follows by noting that above procedure and considerations can be generalized to the case of generic schedules having given utilization U in which there is an arbitrary number of switches. □

Proof of Theorem 2

Proof. The proof is presented for Equation 16 only, since the proof of Equation 15 can be done with similar arguments.

Let us consider the succession S_k. From Corollary 2 it can be noticed that S_k decreases regardless of $s(t)$ while $x(r_k) > x^+(r_{k+1})$. We are interested in finding the value of $\tilde{x}^{\sup}$ such that this value delimits the region in which S_k can only decrease. This region corresponds to the range $[\tilde{x}^{\sup}, +\infty]$ in Figure 3.

The value of $\tilde{x}^{\sup}$ can be obtained by imposing $x(r_k) = x^+(r_{k+1})$, i.e., by finding the fixed point of S'_k defined as:

$S'_k \ : \ x(r_{k+1}) = B + (A - B)\, e^{-\beta(1-U)T} + (x(r_k) - A)\, e^{-(\alpha U + \beta(1-U))T}$ □

Proof of Theorem 3

Proof. We provide the proof for Equation 18, since Equation 17 can be proven similarly.

Theorem 2 indicates an upper bound $\tilde{x}^{\sup}$ for the succession $S_k : x(r_k)$, for $k > k^*$. Therefore, the value of $x^{\sup}$ can be calculated by evaluating the maximum increment that can be achieved by $x(t)$ within one period $[r_k, r_{k+1}]$ when starting from the upper bound of S_k, i.e, when $x(r_k) = \tilde{x}^{\sup}$.

With similar arguments with respect to the proof of Theorem 1 it can be proven that the worst case condition takes place when $s(t) = s^{\Uparrow}(t)$. Therefore, the maximum increment can be easily calculated when the worst activation function is known. □