sensors

Article

Optimal Control for Aperiodic Dual-Rate Systems With Time-Varying Delays

Ernesto Aranda-Escolástico [1,*], Julián Salt [2], María Guinaldo [1], Jesús Chacón [1] and Sebastián Dormido [1]

[1] Departamento de Informática y Automática, Universidad Nacional de Educación a Distancia, 28040 Madrid, Spain; mguinaldo@dia.uned.es (M.G.); jchacon@bec.uned.es (J.C.); sdormido@dia.uned.es (S.D.)

[2] Departamento de Ingeniería de Sistemas y Automática, Instituto Universitario de Automática e Informática Industrial, Universitat Politécnica de Valéncia, 46022 Valencia, Spain; julian@isa.upv.es

* Correspondence: earandae@bec.uned.es; Tel.: +34-913987147

Received: 23 March 2018; Accepted: 3 May 2018; Published: 9 May 2018

Abstract: In this work, we consider a dual-rate scenario with slow input and fast output. Our objective is the maximization of the decay rate of the system through the suitable choice of the n-input signals between two measures (periodic sampling) and their times of application. The optimization algorithm is extended for time-varying delays in order to make possible its implementation in networked control systems. We provide experimental results in an air levitation system to verify the validity of the algorithm in a real plant.

Keywords: multi-rate systems; optimization; aperiodic control; time-varying delay; air levitation system

1. Introduction

This paper is defined in the context of multi-rate systems. Specifically, we consider a dual-rate system, where the output of the system is sampled with sampling period T_s but the input is applied to the system in a faster way. Traditionally, if the controller transmits n input signals, these signals are applied during T_s/n units of time (see [1,2] and references therein). In some cases, the times of application of the input signals might be irregular [3–5] but there are no criteria in the literature to decide this irregularity. In this work, we propose an optimization algorithm which provides not only the input signals but also their times of application, and maximizes the decay rate of the system.

Literature review: The theory of multi-rate control systems has been extensively studied [6–8] due to its multiple applications to real systems. For example, multi-rate control has been used for chemical analyzers [9,10] , visual feedback [11,12], flight control [13], or disk drive servo systems [14,15]. Its application in networked control systems is more recent [5,16,17] and has enabled a saving in communication resources since similar performance can be achieved with a reduced number of transmissions. From the point of view of the optimization problem, optimal controllers have been designed in the framework of multi-rate systems. In [18] an H_2 optimal control is described. A periodic optimal control is proposed in [19]. In [20], an optimal controller is designed based on linear matrix inequality optimization. Several predictive controllers have been studied in [21–24]. However, even when these works propose different optimal controllers, they do not consider irregular sampling or time-varying delays. In [25], the predictive control for multi-rate systems is extended to the distributed case. However, even when these works propose different optimal controllers, they consider neither irregular sampling nor time-varying delays. Recently, the ideas of minimum attention control [26,27] and anytime attention control [28,29] have been studied for networked control systems. In the minimum attention control, the time of application of the control signal is maximized while a certain level of performance is maintained. In anytime attention control, the time of application is previously fixed depending on the available resources and the performance is maximized for that period. The method proposed in this work attempts, in some way, to carry out a combination of both. In the algorithm we maximize the decay rate of the system after a full sampling period of the output, which is fixed, however within this interval of time we vary the times of application as desired to not only maintain the performance but also to improve it.

Statement of contributions: In the present work, we continue the ideas proposed in [30]. In [30], we developed a dual-rate controller to maximize the decay rate of the system through the computation of optimal input signals. In this work, we extend the framework in four directions. Firstly, we consider the optimization not only of the values of the input signals but also of their times of application. This implies obtaining an aperiodic controller that decides the times of application, which is significantly different with respect to other multi-rate aperiodic solutions (see [3] and references therein). Secondly, we study the optimization problem under a model plant mismatching scenario, which makes the algorithm more applicable to real processes. Thirdly, while in [30] constant delays are considered, in this work we extend the solutions for time-varying delays. This approach is more realistic from the point of view of networked control systems [31]. Fourthly, we provide guidelines to implement the optimization algorithm in real systems with limited computation resources and show satisfactory experimental results controlling an air levitation system. Therefore, the optimization with respect to the times of application considering time-varying delays offers not only a larger decay rate than in [30] but also allows dealing with the effect of the delays more efficiently. The contributions of the paper with respect to the literature can be seen from different perspectives. The proposed algorithm is the only one, to the best of our knowledge, that combines optimization and irregular sampling. Hence, the proposed algorithm implies more degrees of freedom in the optimization than other multi-rate optimization algorithms [19–24] because it considers the times of application. Moreover, it provides a criterion to efficiently decide the form of this irregular sampling, which differs from other irregular sampling schemes [3–5], where the times of application are predetermined.

Organization: The reminder of the paper is organized as follows. In Section 2, the problem is presented and the lifted multi-rate model is developed. In Section 3, the proposed algorithm is presented and we extend the results for time-varying delay systems. A discussion about how to perform the optimization more efficiently in systems with hard computation constraints is also provided. In Section 4, the experimental results are provided. Finally, a discussion about the benefits of the method is given in Section 5.

Preliminaries

We define the set of real numbers and the set of natural numbers as $\mathbb{R}$ and $\mathbb{N}$, respectively. The n-dimensional real space is defined by $\mathbb{R}^n$. We refer to the euclidean norm of vector $x \in \mathbb{R}^n$ as $\|x\| = \sqrt{x^\top x}$. Let $M \in \mathbb{R}^{n\times m}$; $M^\top$ denotes the transpose matrix of M. In addition, if M is a symmetric real matrix, then the maximum and the minimum eigenvalue of M are denoted by $\lambda_{\max}(M)$ and $\lambda_{\min}(M)$, respectively. We further denote a symmetric positive-definite matrix $P \in \mathbb{R}^{n\times n}$ as $P > 0$, while $P \geq 0$, $P < 0$, and $P \leq 0$ refer to symmetric positive-semidefinite, negative-definite, and negative-semidefinite matrices, respectively. We denote the identity matrix $\mathbb{I} \in \mathbb{R}^{n\times n}$ by $\mathbb{I}_n$. Let $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^{n\times m}$. We define $\mu(A) = \max\left\{\mu | \mu \in \lambda\left((A+A^\top)/2\right)\right\}$. The norm of the matrix exponential [32] can be bounded then such that

$$\|e^{A\theta}\| \leq e^{\mu(A)\theta} \leq e^{\mu(A)(\theta+\epsilon)},\ \forall\epsilon \geq 0. \tag{1}$$

$B(\theta, A)$ denotes

$$B(\theta, A) = \begin{cases} 0 & \text{if } \theta \leq 0 \\ \int\limits_0^\theta e^{As}Bds & \text{if } \theta > 0 \end{cases} \tag{2}$$

and $B_\mu(\theta, A)$

$$B_\mu(\theta, A) = \begin{cases} 0 & \text{if } \theta \leq 0 \\ \int\limits_0^\theta e^{\mu(A)s}\|B\|ds & \text{if } \theta > 0 \end{cases}.$$

Consequently,

$$\|B(\theta, A)\| \leq B_\mu(\theta, A) \leq B_\mu(\theta + \epsilon, A),\ \forall \epsilon \geq 0. \tag{3}$$

Finally, we define the exponential stability of a system

$$\dot{x}(t) = f(t, x) \tag{4}$$

where $f : [0, \infty) \times D \rightarrow \mathbb{R}^n$ is piecewise continuous in t and locally Lipschitz in x on $[0, \infty) \times D$, and $D \in \mathbb{R}^n$ is a domain that contains the origin $x = 0$ as follows [33]:

Definition 1. *The equilibrium point $x = 0$ of (4) is exponentially stable if there exist positive constants ϵ, c, and α such that*

$$\|x(t)\| \leq ce^{-\alpha(t-t_0)}\|x(t_0)\|,\ \forall \|x(t_0)\| < \epsilon \tag{5}$$

and globally exponentially stable if (5) is satisfied for any initial state $x(t_0)$.

2. Problem Statement

Let us consider a continuous linear time-invariant (LTI) plant denoted by

$$\begin{aligned} \dot{x}_p(t) &= A_p x_p(t) + B_p u(t) \\ y(t) &= C_p x_p(t) \end{aligned} \tag{6}$$

with $x_p(t) \in \mathbb{R}^{n_{xp}}$ the state vector of the plant, $u(t) \in \mathbb{R}^{n_u}$ the input vector, $y(t) \in \mathbb{R}^{n_y}$ the output vector, and $A_p \in \mathbb{R}^{n_{xp} \times n_{xp}}$, $B_p \in \mathbb{R}^{n_{xp} \times n_u}$, and $C_p \in \mathbb{R}^{n_y \times n_{xp}}$ the constant matrices. We assume that the output of (6) is sampled with sampling period T_s as shown in Figure 1. In contrast, the controller changes the input signal n times during T_s. Let us assume that each input u_i for $i = 1, ..., n$ remains constant using a zero-order-hold (ZOH) and is applied during the period of time T_i satisfying $T_s = \sum\limits_{i=1}^{n} T_i$. Then,

$$u(t) = \begin{cases} u_1 \text{ if } t \in [kT_s, kT_s + T_1) \\ u_2 \text{ if } t \in [kT_s + T_1, kT_s + T_1 + T_2) \\ \vdots \\ u_n \text{ if } t \in [kT_s + \sum\limits_{i=1}^{n-1} T_i, kT_s + \sum\limits_{i=1}^{n} T_i) \end{cases}. \tag{7}$$

The discretization of (6)–(7) leads to a periodic linear time-varying discrete system, which implies different problems in order to guarantee the stability and, consequently, in order to maximize the decay rate. A way to avoid this is through lifting techniques [7,8]. We can write the dual-rate sampling scenario in a lifted fashion as

$$x_p(t) = e^{A_p(t-T_s)}x_p(k) + \sum_{i=1}^{n} B_p^i\left(t - T_s, A_p\right)u_i \tag{8}$$

with

$$B_p^i(t - T_s, A_p) = e^{A_p\left(t - T_s - \min\left(t - T_s, \sum\limits_{j=0}^{i} T_j\right)\right)} \times B_p\left(\min\left(t - T_s - \sum_{j=0}^{i-1} T_j, T_i\right), A_p\right). \tag{9}$$

Consequently, the state after a whole sampling period of the output is

$$x_p((k+1)T_s) = \tilde{A}_p x_p(kT_s) + \tilde{B}_p U \tag{10}$$

with $U = \begin{bmatrix} u_1 & \cdots & u_n \end{bmatrix}^\top$, $\tilde{A}_p = e^{A_p T_s}$, and $\tilde{B}_p = \begin{bmatrix} B_p^1(T_s, A_p) & \cdots & B_p^n(T_s, A_p) \end{bmatrix}$.

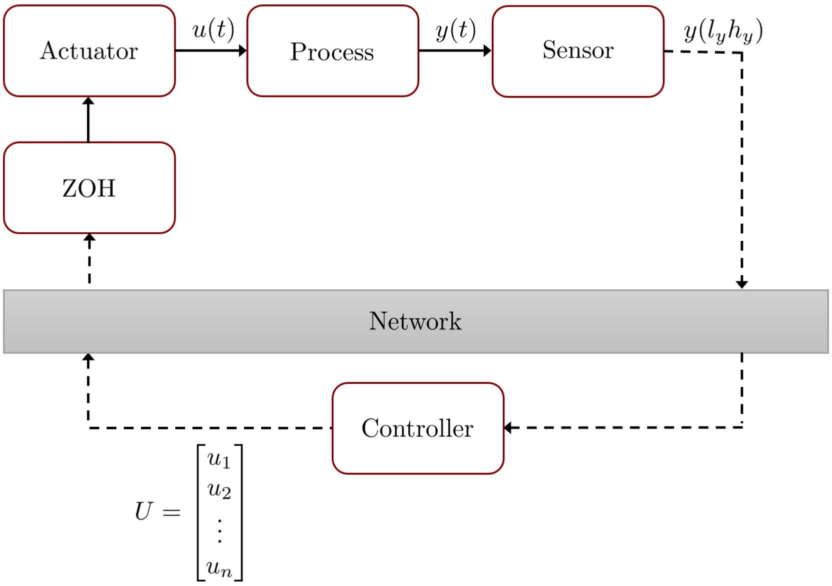

Figure 1. Block diagram of a dual-rate system with fast input.

3. Control Algorithm for Decay Rate Optimization

In this section, we design the algorithm to maximize the decay rate of the system. A priori, the decay rate of a system can be maximized as much as desired if the input signal is sufficiently large. Naturally, this is not an appropriate solution for a real system, where the actuator might saturate the input signal or might not allow fast changes. To overcome this issue, we consider an auxiliary LTI controller for which we guarantee global asymptotic stability with a given maximum gain. Then, we provide the necessary constraints to the optimization problem so that the decay rate is maximized maintaining this maximum gain.

For the optimization of the decay rate, it is necessary to ensure the exponential stability of the system. With this motivation, let us recall the following Lemma.

Lemma 1. *The exponential stability of the discretized plant* (10) *is guaranteed with decay rate* $\hat{\alpha} > 0$, *if there exists a Lyapunov function* $V(x_p(kT_s)) = V(k) : \mathbb{R}^{n_{xp}} \to \mathbb{R}$ *and positive scalars* λ_1, λ_2 *and positive integer* q *such that*

$$\lambda_1 \|x_p(k)\|^q \leq V(k) \leq \lambda_2 \|x_p(k)\|^q \tag{11}$$

and

$$V(k+1) - e^{q\hat{\alpha}T_s} V(k) \leq 0. \tag{12}$$

Next, to compute the input signals $\{u_i\}$ and their times of application $\{T_i\}$, we require an estimation of the model of the plant in order to obtain an estimated set of outputs $\{y_{pm,i}\}$. Let us consider that the model of the plant is described by the estimated state x_{pm}, and let us define the matrices A_{pm}, B_{pm}, C_{pm} analogously to (6). Then, this produces the output

$$
\begin{aligned}
y_{pm,1} &= y_p(kT_s) \text{ for } t \in [kT_s, (k+1)T_s) \\
y_{pm,i} &= C_{pm} x_{pm}(kT_s + \sum_{j=0}^{i-1} T_j) \text{ for } t \in [kT_s, (k+1)T_s) \text{ and } 1 < i \leq n.
\end{aligned}
$$

Let us consider that there exists an auxiliary LTI controller as

$$
\begin{aligned}
\dot{x}_c(t) &= A_c x_c(t) + B_c e(t) \\
\hat{u}(t) &= C_c x_c(t) + D_c e(t),
\end{aligned} \tag{13}
$$

where $e(t) = r(t) - y(t)$, with $r(t)$ being the reference signal (we can take $r(t) = 0$ for simplicity), $x_c(t) \in \mathbb{R}^{n_{x_c}}$ the state vector of the controller, $\hat{u}(t) \in \mathbb{R}^{n_u}$ the auxiliary input vector, and $A_c \in \mathbb{R}^{n_{x_c} \times n_{x_c}}$, $B_c \in \mathbb{R}^{n_{x_c} \times n_y}$, $C_c \in \mathbb{R}^{n_u \times n_{x_c}}$, and $D_c \in \mathbb{R}^{n_u \times n_y}$ the constant matrices. In a multi-rate scenario and using a lifted framework, a virtual closed loop of the augmented state $\hat{x}^\top = \begin{bmatrix} x_p & x_c \end{bmatrix}^\top$ can be written as

$$
\hat{x}((k+1)T_s) = \Pi(\hat{T}_1, ..., \hat{T}_n)\hat{x}(kT_s), \tag{14}
$$

where $\hat{T}_i$ is the time of application of $\hat{u}_i$, and

$$
\begin{aligned}
\Pi(\hat{T}_1, ..., \hat{T}_n) &= e^{AT_s} + e^{A\left(T_s - \hat{T}_1\right)} B(\hat{T}_1, A) K \\
&+ \sum_{i=2}^{n} e^{A\left(T_s - \sum_{j=1}^{i} \hat{T}_j\right)} B(\hat{T}_i, A) K_m \prod_{j=1}^{i} \left(e^{A_m \hat{T}_{i-j}} + B_m(\hat{T}_{i-j}, A_m) K_m \right).
\end{aligned} \tag{15}
$$

$A = \begin{bmatrix} A_p & 0 \\ 0 & A_c \end{bmatrix}$, $B = \begin{bmatrix} B_p & 0 \\ 0 & B_c \end{bmatrix}$, $K = \begin{bmatrix} -D_c C_p & C_c \\ -C_p & 0 \end{bmatrix}$, $A_m = \begin{bmatrix} A_{pm} & 0 \\ 0 & A_c \end{bmatrix}$, $B_m = \begin{bmatrix} B_{pm} & 0 \\ 0 & B_c \end{bmatrix}$, $K_m = \begin{bmatrix} -D_c C_{pm} & C_c \\ -C_{pm} & 0 \end{bmatrix}$ and $\hat{T}_0 = T_0 = 0$.

We further introduce the following assumption:

Assumption 1. *Matrices C_c and D_c, a positive definite matrix P, and scalars $\hat{T}_1, ..., \hat{T}_n \geq 0$ exist such that*

$$
\sum_{i=1}^{n} \hat{T}_n = T_s \tag{16}
$$

$$
\Pi^\top(\hat{T}_1, ..., \hat{T}_n) P \Pi(\hat{T}_1, ..., \hat{T}_n) - e^{-2\hat{\alpha}T_s} P \leq 0, \tag{17}
$$

are satisfied for some $\hat{\alpha} > 0$.

Then, the following theorem can be stated.

Theorem 1. *Consider the auxiliary discretized closed loop system (14). Suppose that Assumption 1 holds. Then, (14) is globally exponentially stable with, at a minimum, a decay rate of $\hat{\alpha} > 0$ and a gain*

$$
\begin{aligned}
c(\hat{T}_1, ..., \hat{T}_n) &= \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}} e^{\hat{\alpha}T_s} \left(e^{\mu(A)T_s} + e^{\mu(A)\left(T_s - \hat{T}_1\right)} B_\mu(\hat{T}_1, A) \|K\| \right. \\
&\left. + \sum_{i=2}^{n} e^{\mu(A)\left(T_s - \sum_{j=1}^{i} \hat{T}_j\right)} B_\mu(\hat{T}_i, A_p) \|K_m\| \prod_{j=1}^{i} \left(e^{\mu(A_m)\hat{T}_{i-j}} + B_{m\mu}(\hat{T}_{i-j}, A_m) \|K_m\| \right) \right).
\end{aligned} \tag{18}
$$

Proof of Theorem 1. Consider a Lyapunov function of the form

$$V(kT_s) = \hat{x}^\top(kT_s)P\hat{x}(kT_s) \tag{19}$$

where $P > 0$. Then,

$$\lambda_{\min}(P)\|\hat{x}(kT_s)\|^2 \leq V(kT_s) \leq \lambda_{\max}(P)\|\hat{x}(kT_s)\|^2$$

is satisfied and (11) is fulfilled. The exponential decrease of the system (14) with decay rate $\hat{\alpha}$ for some $\hat{T}_1, ..., \hat{T}_n$ is achieved if (12) in Lemma 1 is satisfied, i.e., if

$$\begin{aligned} &V((k+1)T_s) - e^{-2\hat{\alpha}T_s}V(kT_s) = \\ &\quad \hat{x}^\top(kT_s)\left(\Pi^\top(\hat{T}_1, ..., \hat{T}_n)P\Pi(\hat{T}_1, ..., \hat{T}_n) - e^{-2\hat{\alpha}T_s}P\right)\hat{x}(kT_s) \leq 0, \end{aligned} \tag{20}$$

which is guaranteed by (17). To find $c(\hat{T}_1, ..., \hat{T}_n)$, we use (19) and (20). From (20),

$$V\left((k+1)\,T_s\right) \leq e^{-2\hat{\alpha}T_s}V\left(kT_s\right) \leq \cdots \leq e^{-2\hat{\alpha}kT_s}V\left(0\right)$$

and using (19), we obtain

$$\|\hat{x}(k)\| \leq \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}}e^{-\hat{\alpha}kT_s}\|\hat{x}(0)\|.$$

The state of the plant in any instance of time $t = kT_s + \delta, \forall\delta \in [0, T_s]$ is also exponentially stable since it can be bounded in the view of (8) as follows

$$\begin{aligned} \|\hat{x}(t)\| \leq \Big(&\|e^{AT_s}\| + \|e^{A(T_s-\hat{T}_1)}\|B(\hat{T}_1, A)\|\|K\| \\ &+ \sum_{i=2}^{n} e^{\|A\|\left(T_s-\sum\limits_{j=1}^{i}\hat{T}_j\right)}\|B(\hat{T}_i, A)\|\|K_m\|\prod_{j=1}^{i}\left(\|e^{A_mT_{i-j}}\| + \|B_m(\hat{T}_{i-j}, A_m)\|\|K_m\|\right)\Big)\|\hat{x}(kT_s)\| \end{aligned} \tag{21}$$

Since $\|\hat{x}\left(kT_s + \sum\limits_{j=0}^{i}T_j\right)\|$ can be bounded recursively until obtaining a bound which depends on $\hat{x}(0)$, and using (1) and (3), we can write

$$\begin{aligned} \|\hat{x}(t)\| \leq \tilde{c}(\hat{T}_1, ..., \hat{T}_n)\|\hat{x}(kT_s)\| &\leq \tilde{c}(\hat{T}_1, ..., \hat{T}_n)\sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}}e^{-\hat{\alpha}kT_s}\|\hat{x}(0)\| \\ = c(\hat{T}_1, ..., \hat{T}_n)e^{-\hat{\alpha}(t)}\|\hat{x}(0)\|. \end{aligned} \tag{22}$$

Thus, the proof is completed. □

Having proved that there exists a controller able to globally exponentially stabilize (14), we can design the input signals and their times of application, which maximize the decay rate of the system (10). As aforementioned, maximizing the decay rate enlarges the gain. Therefore, we need to establish constraints to maintain c in acceptable values. Let us consider the following assumptions in order to guarantee the optimization of the decay rate while satisfying these conditions.

Assumption 2. *A bound $\bar{c}$ can be computed such that $\bar{c} = \max(c(\hat{T}_1, ..., \hat{T}_n))$.*

Assumption 3. *The set of control signals* $\{u_i\}$ *for* $i = 1, ..., n$ *satisfies that*

$$\|u_1\| \leq \|K\|\|x\left(kT_s\right)\| \tag{23}$$

$$\|u_i\| \leq \|K_m\|\|x_m\left(kT_s + \sum_{j=0}^{i-1} T_j\right)\|, \text{ for } i > 1. \tag{24}$$

Then, the following theorem can be stated.

Theorem 2. *Consider the discretized closed loop system* (10) *and control signals* $\{u_i\}$ *applied during periods* $\{T_i\}$ *for* $i = 1, ..., n$. *Suppose that Assumption 2 and 3 hold. Then, the system* (10) *is globally exponentially stable with at least decay rate* $\alpha \geq \hat{\alpha}$ *and gain* $\bar{c}$.

Proof of Theorem 2. By Assumption 2, $\bar{c} \geq c(T_1, ..., T_n)$. From Theorem 1 it is guaranteed at least $\alpha = \hat{\alpha}$ if $u_i = \hat{u}_i$ and $T_i = \hat{T}_i$ for $i = 1, ..., n$. Equations (23) and (24) from Assumption 3 imply that $\|x(t)\|$, for $t \in [kT_s, (k+1)T_s)$, satisfies (in the view of (21))

$$\begin{aligned} \|x(t)\| \leq & \left(\|e^{AT_s}\| + \|e^{A(T_s - T_1)}\|\|B(T_1, A)\|\|K\|\right. \\ & + \sum_{i=1}^{n} e^{\|A\|\left(T_s - \sum_{j=2}^{i} T_j\right)} \|B(T_i, A)\|\|K_m\| \\ & \left. \times \prod_{j=1}^{i} \left(\|e^{A_m T_{i-j}}\| + \|B_m(T_{i-j}, A_m)\|\|K_m\|\right)\right)\|x(kT_s)\| \\ & \leq c(T_1, ..., T_n)\|x(0)\| \leq \bar{c}\|x(0)\|. \end{aligned} \tag{25}$$

Hence, Definition 1 is fulfilled for the closed loop system. If we consider that the initial state of the auxiliary controller is zero, then

$$\|x_p(t)\| \leq \bar{c}e^{\alpha(t)}\|x_p(0)\|, \tag{26}$$

Thus, the process is globally exponentially stable with, at least, decay rate α and gain $\bar{c}$ and the proof is completed. □

Remark 1. *If we assume that the model matches with the actual system dynamics, i.e., if we assume that* $A_m = A_p$, $B_m = B_p$ *and* $C_m = C_p$, *then* $\bar{c} = c(T, ..., T)$, *where* $T = T_s/n$ *(see Appendix A).*

3.1. Extension for Time-Varying Delay Systems

In this section, the results from Section 3 are formalized for the case with time-varying delays. Delays might affect to a control system in different ways. In our case, there are two main sources of delays. On the one hand, the transmission of information through the network might cause time-varying delays. On the other hand, the computation of the solutions of the optimization problem also induces a delay τ_{com}, which we can assume constant [34]. Let us denote the transmission delays from the sensor to the controller and from the controller to the sensor as τ_{sc}, τ_{ca}, respectively, and let us assume that they are upper bounded as $0 \leq \tau_{sc} \leq \tau_{M,net}$ and $0 \leq \tau_{ca} \leq \tau_{M,net}$. Hence, the input signals in a meta-period k reach the actuator with a total delay $\tau(k)$, which is bound such that $\tau_{com} \leq \tau(t) \leq \tau_{com} + 2\tau_{M,net} = \tau_M$. To simplify the stability analysis, let us consider a minimum unit of delay τ_m such that $\tau_M = l\tau_m$ for $l \geq 1$. Taking this into account, we consider that the actuator waits to apply the inputs signals until the next multiple of τ_m is larger than $\tau(k)$, i.e., the actual delay applied to the plant is $\hat{\tau}(k) = \min\{l\tau_m | \tau(k) \leq l\tau_m\}$.

Now, we follow the description in [5,35] to deal with delays in multi-rate systems. Let us assume that during the delay, the last input signal of the previous output sampling period is held. Let us also consider a vector $\psi(k)$ which corresponds to the last input signal in the previous metaperiod. Then, we can define the following augmented state

$$\xi(t) = \begin{pmatrix} x(t) \\ \psi(k) \end{pmatrix}, \; \forall t \in [kT_s, (k+1)T_s). \tag{27}$$

As in Section 3, the auxiliary controller (13) is used to define a virtual closed loop

$$\hat{\xi}\left((k+1)T_s\right) = \Pi_d \hat{\xi}(kT_s) = \begin{bmatrix} \Pi_d^{11}(T_0, \hat{T}_1, ..., \hat{T}_n) & \Pi_d^{12}(T_0, \hat{T}_1, ..., \hat{T}_n) \\ \Pi_d^{21}(T_0, \hat{T}_1, ..., \hat{T}_n) & \Pi_d^{22}(T_0, \hat{T}_1, ..., \hat{T}_n) \end{bmatrix} \hat{\xi}(k) \tag{28}$$

where $\hat{\xi}^\top(k) = \begin{pmatrix} \hat{x}(k) & \psi(k) \end{pmatrix}$ and

$$\begin{aligned} \Pi_{11}^d(T_0, \hat{T}_1, ..., \hat{T}_n) &= e^{AT_s} + \sum_{i=1}^{n} e^{A\left(T_s - \sum\limits_{j=1}^{i} \hat{T}_j\right)} B(\hat{T}_i, A) K_m \\ &\times \prod_{j=1}^{i} \left(e^{A_m \hat{T}_{i-j}} + B_m(\hat{T}_{i-j}, A_m) K_m \right) e^{A_m T_0} \\ \Pi_{12}^d(T_0, \hat{T}_1, ..., \hat{T}_n) &= e^{A(T_s - T_0)} B(T_0, A) + \sum_{i=1}^{n} e^{A\left(T_s - \sum\limits_{j=1}^{i} \hat{T}_j\right)} B(\hat{T}_i, A) K_m \\ &\times \prod_{j=1}^{i} \left(e^{A_m \hat{T}_{i-j}} + B_m(\hat{T}_{i-j}, A_m) K_m \right) B_m(T_0, A_m) \end{aligned}$$

$$\begin{aligned} \Pi_{21}^d(T_0, \hat{T}_1, ..., \hat{T}_n) &= K_m \left(e^{AT_s} + \sum_{i=1}^{n-1} e^{A\left(T_s - \sum\limits_{j=1}^{i} \hat{T}_j\right)} B(\hat{T}_i, A) K_m \right. \\ &\left. \times \prod_{j=1}^{i} \left(e^{A_m \hat{T}_{i-j}} + B_m(\hat{T}_{i-j}, A_m) K_m \right) e^{A_m T_0} \right) \\ \Pi_{22}^d(T_0, \hat{T}_1, ..., \hat{T}_n) &= K_m \left(e^{A(T_s - T_0)} B(T_0, A) + \sum_{i=1}^{n-1} e^{A\left(T_s - \sum\limits_{j=1}^{i} \hat{T}_j\right)} B(\hat{T}_i, A) K_m \right. \\ &\left. \times \prod_{j=1}^{i} \left(e^{A_m \hat{T}_{i-j}} + B_m(\hat{T}_{i-j}, A_m) K_m \right) B_m(T_0, A_m) \right), \end{aligned}$$

where $T_0 = \tau(k)$. Then, we can use again Lemma 1 to guarantee the exponential stability of (28). Let us recall and extend Assumption 1 for the augmented state $\hat{\psi}(k)$.

Assumption 4. *Matrices C_c and D_c, a positive definite matrix P, and scalars τ_m, $\hat{T}_{1,l} = \tau_M - l\tau_m$, $\hat{T}_2, ..., \hat{T}_n \geq 0$ exist such that*

$$l\tau_m + \hat{T}_{1,l} + \sum_{i=2}^{n} \hat{T}_n = T_s \tag{29}$$

$$\Pi_d^\top(l\tau_m, \hat{T}_{1,l}, \hat{T}_2 ..., \hat{T}_n) P \Pi_d(l\tau_m, \hat{T}_{1,l}, \hat{T}_2 ..., \hat{T}_n) - e^{-2\hat{\alpha}T_s} P \leq 0, \tag{30}$$

are satisfied for some $\hat{\alpha} > 0$ and for all $l \in \mathbb{N}$ such that $\tau_m \leq l\tau_m \leq \tau_M$.

The following Theorem is formulated.

Theorem 3. *Consider the auxiliary discretized closed loop system (28). Suppose that Assumption 1 holds. Then, (28) is globally exponentially stable with, at least, a decay rate $\hat{\alpha} > 0$ and gain*

$$\hat{c} = \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}} e^{\hat{\alpha} T_s} \left(\hat{c}_1 + \hat{c}_2\right), \tag{31}$$

where $\hat{c}_i = \max\left(c_i\left(l\tau_m, \hat{T}_{1,l}, \hat{T}_2, ..., \hat{T}_n\right)\right)$ for $i = 1, 2$, and

$$\begin{aligned} c_1(T_0, \hat{T}_1, ..., \hat{T}_n) = {} & e^{AT_s} + \sum_{i=1}^{n} e^{\mu(A)\left(T_s - \sum_{j=1}^{i} \hat{T}_j\right)} B_\mu(\hat{T}_i, A)\|K_m\| \\ & \times \prod_{j=1}^{i} \left(e^{\mu(A_m)\hat{T}_{i-j}} + B_{m\mu}(\hat{T}_{i-j}, A_m)\|K_m\|\right) e^{\mu(A_m)T_0} \\ c_2(T_0, \hat{T}_1, ..., \hat{T}_n) = {} & e^{\mu(A)(T_s - T_0)} B_\mu(T_0, A) + \sum_{i=1}^{n} e^{\mu(A)\left(T_s - \sum_{j=1}^{i} \hat{T}_j\right)} B_\mu(\hat{T}_i, A)\|K_m\| \\ & \times \prod_{j=1}^{i} \left(e^{\mu(A_m)\hat{T}_{i-j}} + B_{m\mu}(\hat{T}_{i-j}, A_m)\|K_m\|\right) B_{m\mu}(T_0, A_m). \end{aligned}$$

Proof of Theorem 3. The discretized augmented state $\|\hat{\xi}(kT_s)\|$ can be exponentially bounded such that

$$\|\xi(kT_s)\| \leq \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}} e^{-\hat{\alpha}(kT_s)} \|\xi(0)\|$$

due to (30). In addition, if it holds that $\|\psi(0)\| = 0$ (in the initial conditions there is no previous input signal), then

$$\|\xi(kT_s)\| \leq \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}} e^{-\hat{\alpha}(kT_s)} \|\hat{x}(0)\|. \tag{32}$$

Furthermore, with the same arguments as in Theorem 1, we guarantee that at any time the global exponential stability of the closed-loop system is as follows:

$$\begin{aligned} \|\hat{x}(t)\| \leq {} & \left(\|\Pi_{11}^d(l\tau_m, \hat{T}_{1,l}, \hat{T}_2, ..., \hat{T}_n)\| + \|\Pi_{21}^d(l\tau_m, \hat{T}_{1,l}, \hat{T}_2, ..., \hat{T}_n))\|\right) \|\xi(kT_s)\| \\ & (c_1(l\tau_m, \hat{T}_{1,l}, \hat{T}_2, ..., \hat{T}_n)) + c_2(l\tau_m, \hat{T}_{1,l}, \hat{T}_2, ..., \hat{T}_n))) \|\xi(kT_s)\| \end{aligned}$$

Thus, computing the maximum value of c_1 and c_2, (31) is obtained. □

Trivially, Theorem 2 is still valid for the delay case and the optimization can be carried out. Since we now take into account the delay, we perform the optimization in a more realistic way, which helps to maximize the decay rate in a real scenario.

3.2. Implementation of Optimization Algorithm in Real-Time Systems

The proliferation of board computers like Raspberry Pi or Beaglebone Black [36,37] together with the emergence of communication networks to connect the elements in a control loop implies that the available computation time might be reduced. For this reason, it is important to limit the computational effort made by the controller in order to guarantee acceptable behavior. In our case, the optimization algorithm can be relaxed in different ways.

A significant source of computations is the discretization of closed-loop system. Since (2) implies solving a matrix integral and since one of the optimization variables is the upper limit of the integral, the optimization algorithm has to execute a hard computation in real time to obtain the discretized model for the corresponding times of application. However, we know the maximum and minimum possible times of application of each signal because they should satisfy (29). Then, we can consider a sufficiently small sampling period h in the view of n and the sampling period of the plant T_s to carry out an offline computation of a discretized matrices grid, i.e., we compute offline $e^{A(mh)}$ and $B(mh, A)$ for all $m \in \mathbb{N}$ such that $0 \leq mh \leq T_s$. Hence, given the discretized matrices grid, the optimization function should only access the required value instead of computing the corresponding discrete matrix.

The constraints in (23)–(24) might also be a source of delay in the computation. When the times of application T_i are taken into account in the optimization, we should consider (23)–(24) as nonlinear constraints of the optimization problem. If there is a large number of input signals, this might constitute a hard computation effort. To avoid this, the optimization algorithm might be solved in two steps. A first step provides the times of application that optimize the decay rate for the auxiliary controller. With fixed times of application, Equations (23)–(24) are transformed in constant limits for the second part of the optimization algorithm, which obtains the actual values of the input signals. Naturally, this results in a suboptimal solution of the problem. In spite of that, the solution always improves the performance of the auxiliary controller.

A final observation in order to minimize the computation time is the consideration of a variable number of input signals to be optimized. Since we know the delay between the sensor and the controller, we can make an estimation of the available computation time and then analyze if it is preferable to reduce the number of input signals (and their corresponding times of application) which are optimization variables, i.e., $n_{\text{opt}}(kT_s) \leq n$, while the rest of the input signals $n - n_{\text{opt}}$ are computed using the auxiliary controller, which typically requires fewer computation resources. We hope to study in detail this possibility in future works in order to increase the algorithm's applicability.

The implementation of these characteristics in the optimization problem is schematically described in Algorithm 1.

Algorithm 1 Optimization algorithm with limited computation resources.

```
Offline Computation
  Step 1off   Synthesis of the auxiliary controller satisfying Assumption 4.
  Step 2off   Computation of the minimum decay rate $\hat{\alpha}$ and the maximum gain $\hat{c}$.
  Step 3off   Discretization of the closed-loop system for all times $mh$, $m \in \mathbb{N}$, which satisfy $0 \leq mh \leq T_s$.
Online Computation
  Step 1on    Reception of the output signal $y(kT_s)$ with a delay $\tau_{sc}(kT_s)$.
  Step 2on    Estimation of the state of the plant from the output.
  Step 3on    Decision of the number of input signals $n_{\text{opt}}$ to be optimized as function of the available
              computation.
  Step 4on    Optimization of the decay rate as function of the times of application of the input signals
              maintaining the auxiliary controller.
  Step 5on    Computation of the maximum and minimum bounds to the actual input signals in
              Assumption 3.
  Step 6on    Optimization of the decay rate as function of the values of the input signals maintaining
              constant the times obtained in Step 4on.
```

4. Practical Case: Air Levitation System

In this section, we present the experimental results obtained with the optimization algorithm, where we take into account the considerations explained in Section 3.2. We have used the air levitation system proposed in [38] for the experimental validation. Next, a description of the hardware is given. The model of the system and the experimental results are presented afterwards.

4.1. Hardware Description

The air levitation system experimentation platform, based on Arduino Nano, provides two working modes: *standalone*, i.e., the controller is deployed into the Arduino board, and *passthrough*, in which the controller is implemented in the PC and the Arduino board acts mainly as a gateway to interact with the hardware. The system architecture is depicted in Figure 2. The block diagram represents the different functional subsystems: the controller is implemented in the system on a chip (SoC) Arduino board, which also provides digital-to-analog (D/A) and analog-to-digital (A/D) converters. Additional circuitry is required to condition the sensor readings, and to amplify the control signal in order to provide enough energy for the actuators (fan and servo).

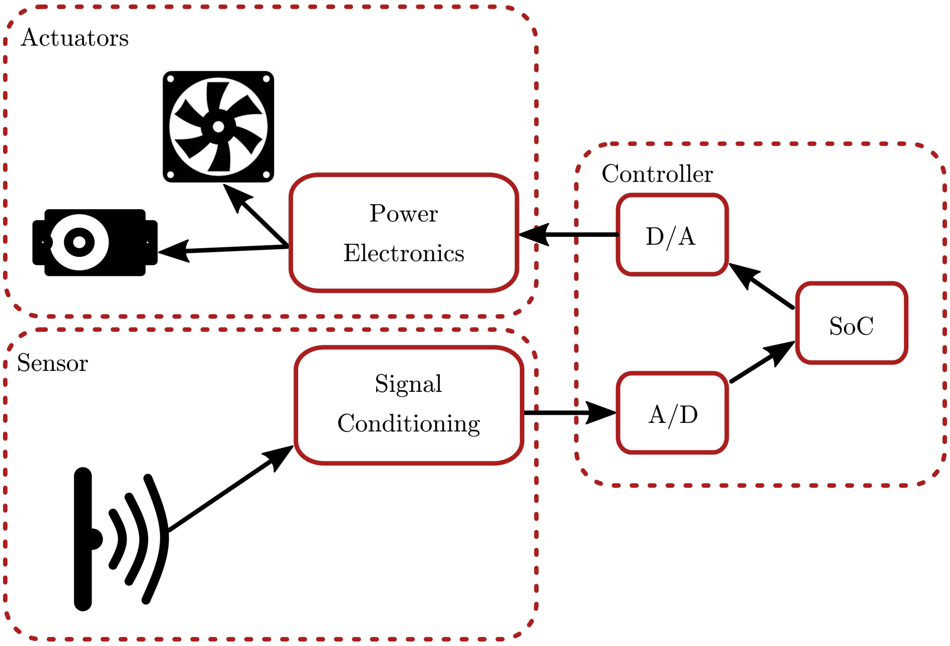

Figure 2. Schematic view of the air levitation control loop.

To measure the position of the ball, several possibilities were considered: visual recognition, ultrasonic sensors, and infrared sensors. While it could be interesting to use a video cam to get the ball position, and it may even be adequate for teaching with an image-processing subject, the complexity and cost of the system would increase, so this was discarded as a possibility. With respect to the ultrasonic distance sensors, they are a valid alternative to infrared ones, and are similar in cost and complexity. However, the latter option was finally chosen. The position of the ball is measured with an infrared beam sensor, specifically a Sharp GP2Y0A21YK0F Analog Distance Sensor (Sharp Corporation), which can obtain measurements between 10 and 80 cm. There are other similar models that are electrically compatible and have different ranges, such as the GP2Y0A21YK0F (4–30 cm) and the GP2Y0A02YK0F (20–150 cm), and thus a choice can be made in order to adapt to different tube lengths. All the aforementioned sensors are analog, yielding a signal roughly in the range of 0–5 V, which is proportional to the inverse of the distance measured. The sensor is composed of two infrared LEDs, an emitter which projects a light beam, and a receiver that measures the bounce in the detected object. Since the sensor measures the light reflected by the object, it may be affected by the color, shape and movement of the object. Also, it has an update period of approximately 40 ms. These aspects must be taken into account to get reliable measures.

The fan speed control, as well as the servo position, are achieved through *Pulse Width Modulation* (PWM). The fan speed, and consequently the air flow, can be controlled with a high frequency PWM. Also, most digital interfaced servomotors admit a PWM signal, with a carrier frequency of 20 HZ, which codes the angular position into the duty cycle, though it is not the same for all models. For example, many servos provide a mapping between (1 ms, 2 ms)$\mapsto$(0°, 180°). While Arduino provides built-in PWM outputs, these provide limited current, so there is a need for additional components, as mentioned before.

The *hardware interface* subsystem is in charge of reading measures from the sensors and sending values to the actuators. Though it is obviously platform-dependent, it is good practice to use standard libraries and protocols. For example, the Arduino application programming interface (API) is widely used for its simplicity and it has been exported to other hardware, like the Beaglebone boards or Raspberry PI. The functionality to be covered can usually be reduced to read and write digital or analog input and outputs. In the air levitation system, the hardware interface task is accomplished using *Javascript Object Notation* (JSON), which is human-friendly (it can be easily read), and at the same time can be processed by most programming languages.

There is a *real-time loop* implementing the time-critical actions: reading sensors, updating the controller, and writing outputs. A lower priority loop is responsible for two tasks: sending measurements and logs to the PC and receiving user commands (for example, to change the setpoint or other parameters). Once the values have been acquired, it is important to store them in order to be accessed whenever is required. The *datalogging* capabilities have been separated into a low priority task that periodically dumps measures and control actions to a database, so the data is stored and can be accessed to perform off-line processing of past sessions.

The *control* subsystem implements a *proportional–integral–derivative PID* controller for which parameters can be modified and tuned. The control subsystem is prepared to be extended with more sophisticated controllers without much development effort. Apart from this PID implementation and template, a passthrough mode is provided that can be used to use the user PC as a controller. This is convenient for research, for example to test a prototype developed in MATLAB, or when there are high computation power needs.

4.2. System Model

The only forces acting over the levitating object are the upward effect of the air flow, and the downwards effect of the gravity, as shown in Figure 3. According to this, Newton's second law gives the nonlinear model

$$m\Delta\ddot{z}(t) = \frac{1}{2}C_d\rho A\left(v_w - \dot{z}(t)\right)^2 - mg, \tag{33}$$

where m is the mass of the object to levitate, z is the vertical position of the object in the tube, ρ is the density of air, A is the object's area exposed to the upwards air flow, v_w is the velocity of the air inside the tube, g is the gravitational acceleration, and C_d is the drag coefficient.

Following [38], a linearized system from (33) can be obtained with matrices

$$A_p = \begin{pmatrix} -8.52 & -2.97 & 0 \\ 4.00 & 0 & 0 \\ 0 & 1.00 & 0 \end{pmatrix},\ B_p = \begin{pmatrix} 2.00 \\ 0 \\ 0 \end{pmatrix}, \qquad C_p = \begin{pmatrix} 0 & 0 & 3.36 \end{pmatrix},\ D_p = 0.$$

We use this model to solve the optimization problem. The consideration of the times of application as variables of the optimization problem provides not only more degrees of freedom in the solutions but also a better treatment of the delays, since T_1 and hence the first imput signal u_1, can be computed accordingly. As auxiliary controller, we make use of a PI controller with gains $K_p = 0.006$ and $K_c = 0.002$. We consider a sampling period of the output $T_s = 300$ ms. We assume that there is a constant computation delay $\tau_{\text{com}} = 50$ ms and there is a time-varying delay because of the communication through the network, which is bounded by $\tau_{M,\text{net}} = 25$ ms. We consider $n = 2$, i.e., the input signal is changed twice in each T_s. Applying Theorem 3, we obtain $\hat{\alpha} = 0.0017$ and $\hat{c} = 734.24$.

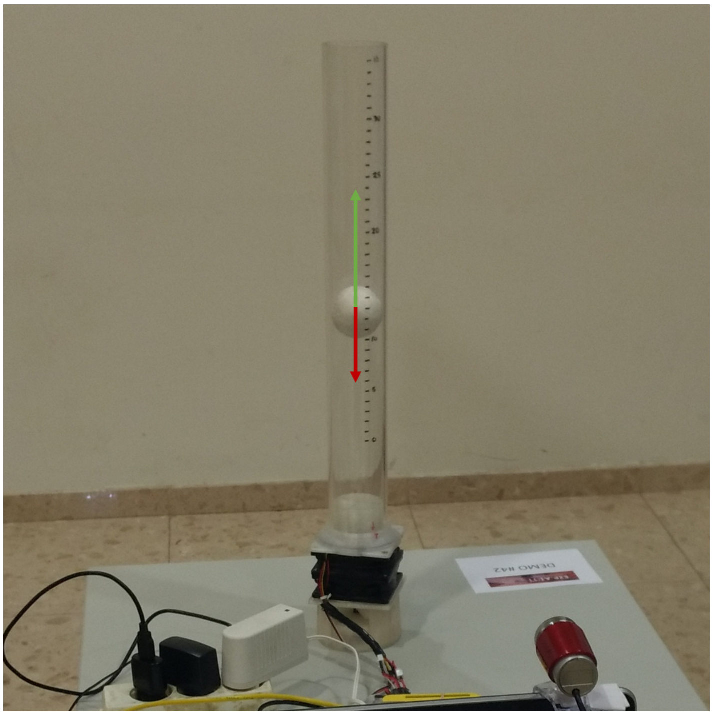

Figure 3. Balance of forces of the air levitation system used in the experiment. The red arrow shows the gravitational force, while the green arrow shows the force applied by the air flow.

4.3. Experimental Results

Firstly, as a preliminary test of the validity of the developed theory, we carry out a simulation of the linear model. In the simulation, we stabilize the state of the system in the origin. As shown in Figure 4, the decay rate is considerably improved using the optimization algorithm with respect to the auxiliary PI controller. In addition, we can further improve the results by performing the optimization as function of the times of application. The novel algorithm allows the use of the knowledge of the delays to appropriately choose the times of application. This can be observed in the histogram of Figure 5.

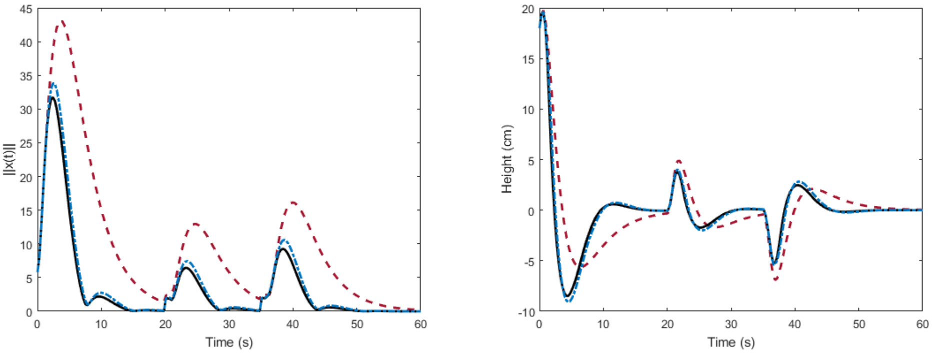

Figure 4. Norm of the state vector (**left**) and output response (**right**) of the linear model of the air levitation system. Solid black line: the proposed algorithm. Dot-dashed blue line: the algorithm in [30]. Dashed red line: auxiliary PI controller.

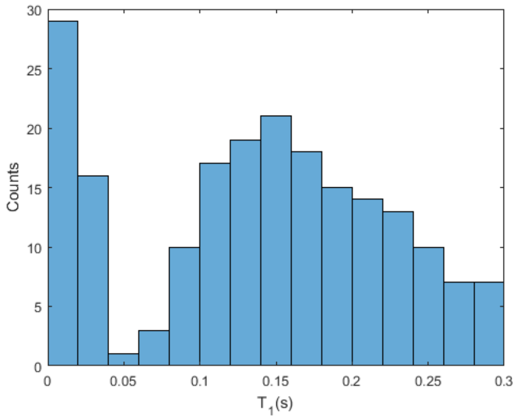

Figure 5. Histogram of times of application of u_1 in the linear model of the air levitation system.

Secondly, we perform two experiments in the real system. In Figure 6, we can observe the results obtained with the proposed algorithm with respect to the auxiliary PI controller and with respect to the optimization algorithm considering constant times of application. In the experiment, the signal reference is set to 18 cm, and later two reference changes are made. At 60 s the reference is set to 28 cm, and at 100 s a disturbance is introduced into the servo. The input signals are shown in Figure 7. The optimization of the decay rate implies a more aggressive input. In spite of the inherent limitations (sensor sensitivity, disturbances, model errors, etc.) of the experimental system, which contribute to complicate the optimization, the proposed algorithm provides improvements during the stabilization in measure of the integral square error (ISE) of 31.77% and 5.35%, and in the settling time of 42.86% and 20.00% with respect to the auxiliary controller and the algorithm in [30], respectively, as shown in Table 1. In the histogram of Figure 8, we observe the distribution of times of application of input signal u_1. The times of application of u_2 can be computed taking into account the delay.

In Figure 9, the response of the system is represented for the three different methods when different disturbances are applied. Specifically, we apply a disturbance with form of sawtooth wave between 80 and 160 s. Later, we apply several impulse disturbances between 200 and 260 s, as shown in Figure 10. Similarly to the first experiment, the optimization methods cause more changes in the input signal (see Figure 11), which is the price to pay for maximizing the decay rate. Table 1 gathers the information about the ISE during the different disturbances, obtaining improvements of 27.65% and 13.59% in the sawtooth wave disturbance rejection with respect to the auxiliary controller and the algorithm in [30], respectively, and of 33.01% and 27.37% in the impulse disturbance rejection, respectively.

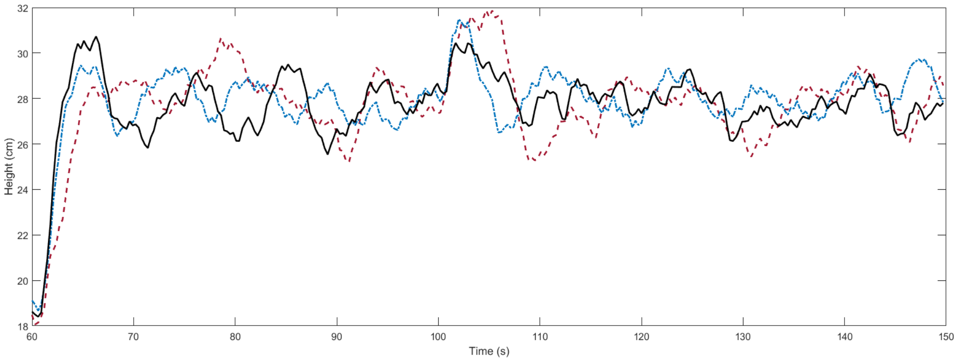

Figure 6. Output response of the air levitation system in the stabilization experiment. Solid black line: proposed algorithm. Dot-dashed blue line: algorithm in [30]. Dashed red line: auxiliary PI controller.

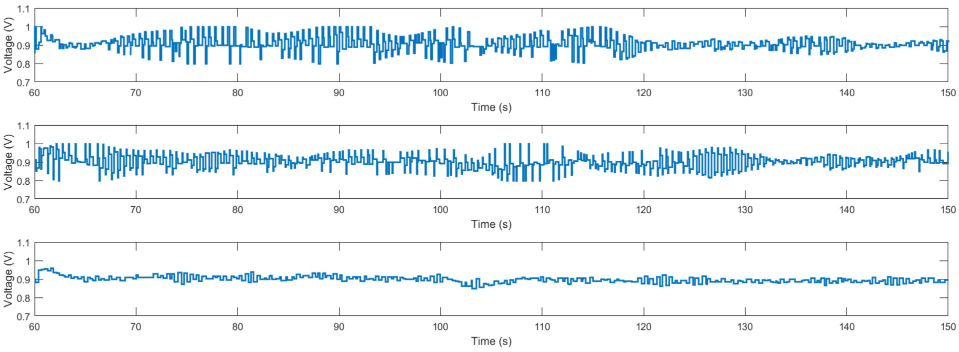

Figure 7. Input voltage applied to the air levitation system in the stabilization experiment. From top to bottom: the proposed algorithm, the algorithm in [30], and the auxiliary PI controller.

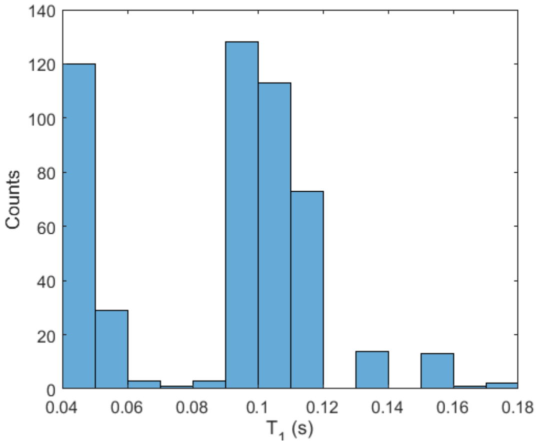

Figure 8. Histogram of times of application of u_1 in the real system.

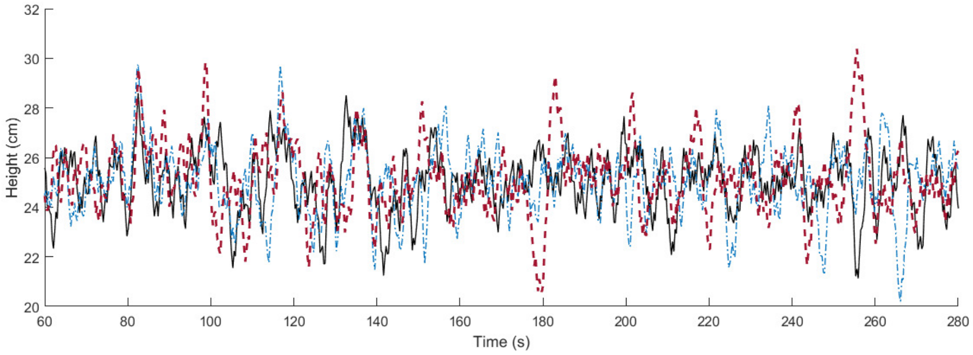

Figure 9. Output response of the air levitation system in the disturbance rejection experiment. Solid black line: proposed algorithm. Dot-dashed blue line: algorithm in [30]. Dashed red line: auxiliary PI controller.

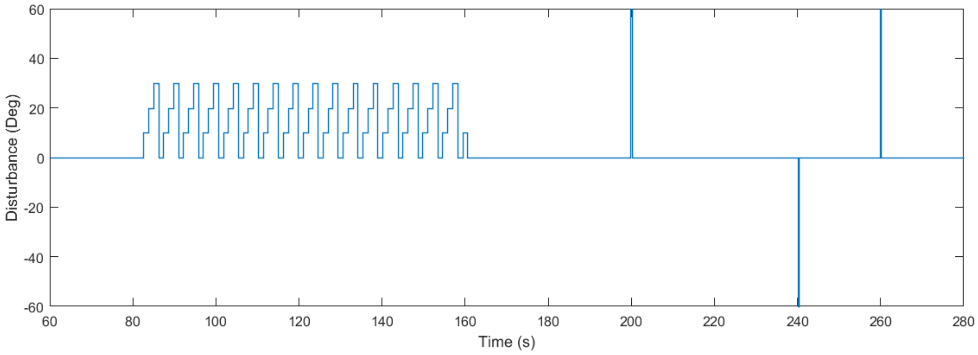

Figure 10. Disturbance applied to the system. Solid black line: proposed algorithm. Dot-dashed blue line: algorithm in [30]. Dashed red line: auxiliary PI controller.

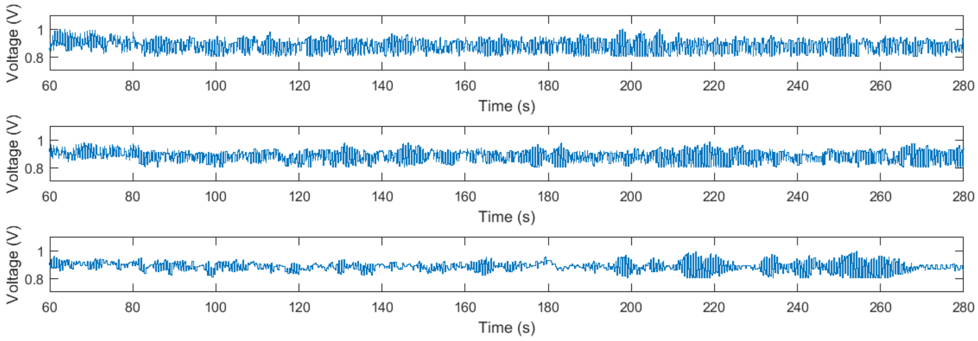

Figure 11. Input voltage applied to the air levitation system in the disturbance rejection experiment. From top to bottom: the proposed algorithm, the algorithm in [30], and the auxiliary PI controller.

Table 1. Comparison of different controllers.

Controller	Settling Time	ISE (Stabilization)	ISE (Sawtooth Wave)	ISE (Impulse)
Auxiliary PI controller	4.2 s	421.39	291.01	172.43
Algorithm in [30]	3.0 s	303.78	243.64	159.03
Proposed algorithm	2.4 s	287.53	210.54	115.51

5. Discussion

In this paper, we propose an optimization algorithm for a multi-rate input control system. The algorithm is designed to be implemented in real networked control systems. For this reason, the algorithm is designed to take into account time-varying delays. The proposed method presents some advantages with respect to to the existing literature. It takes into account the times of application and the time-varying delays in the optimization problem as opposed to [19–24]. In addition, these variable times of application cause irregular sampling, as considered in [3–5], but with the difference that the irregular sampling in the proposed method is motivated by a theoretical criterion to maximize the decay rate, and in [3–5] it is just a tuning parameter. In contrast, the computing time of the algorithm might be considerably large depending on the number of control signals and time of application to be optimized. Some solutions, such as precomputing the discretization of the matrices, are provided to reduce the computational load of the algorithm in order to be implemented in single-board computers. These solutions enable the implementation of the method in a larger number of plants. As an example, we successfully apply the control scheme to an air levitation system, while the algorithm implemented without these consideration, such as in [30], requires a larger computing time and the control objective cannot be achieved. Despite this, the proposed algorithm always induces a larger delay than the associated auxiliary controller. Hence, its implementation in fast systems should be carefully considered. Both the theoretical and experimental results show that the proposed algorithm allows a faster response of the system while dealing with network delays.

The algorithm could be improved in several ways. A more accurate estimation of the delays, or even some kind of disturbance estimation, would provide a better model for the optimization and, naturally, an improvement in the system response. As aforementioned, the main issue of the algorithm is the delay that it induces in the system due to the running time. For this reason, novel methods to reduce the running time might be a way to improve the results of the algorithm. Finally, in this paper we consider a dual-rate scheme with fast input, but future works might include other multi-rate configurations.

Author Contributions: E.A.-E. and M.G. designed the optimization algorithm. J.S. provided the multi-rate formalism and the extension for delays. J.C. designed the experimental setup. E.A.-E., J.C. and M.G. analyzed the data. S.D. performed the theoretical conception. E.A.-E. wrote the paper and it was revised and modified by the rest of the authors.

Acknowledgments: This work was supported in part by the Spanish Ministry of Economy and Competitiveness (MINECO) under the Projects DPI2012-31303 and DPI2014-55932-C2-2-R.

Conflicts of Interest: The authors declare no conflict of interest. The founding sponsors had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, and in the decision to publish the results.

Appendix A

First of all, we observe that if the estimation model coincides with the actual plant, (15) can be written as

$$\Pi = \prod_{i=1}^{n} e^{A\hat{T}_i} + B\left(\hat{T}_i, A\right) K.$$

Consequently,

$$c(\hat{T}_1, ..., \hat{T}_n) = \sqrt{\frac{\lambda_{\max}(P)}{\lambda_{\min}(P)}} e^{\hat{\alpha} T_s} \prod_{i=1}^{n} \left(e^{\mu(A)\hat{T}_i} + B_\mu(\hat{T}_i, A)\|K\| \right).$$

For simplicity, let us consider the two first control actions. They are applied to the plant during periods $\hat{T}_1$ and $\hat{T}_2$, respectively, such that $\hat{T}_1 + \hat{T}_2 = \hat{T}_{12}$. We can search the maximum of

$$c_{12}(\hat{T}_1) = \left(e^{\mu(A)(\hat{T}_{12}-\hat{T}_1)} + B_\mu((\hat{T}_{12} - \hat{T}_1), A)\|K\| \right) \left(e^{\mu(A)\hat{T}_1} + B_\mu(\hat{T}_1, A)\|K\| \right),$$

which can be considered independently of the rest of the actuation periods. The derivative of c_{12} results in

$$\begin{aligned} \frac{dc_{12}(\hat{T}_1)}{d\hat{T}_1} &= (\mu(A) + \|B\|\|K\|)\, e^{\mu(A)(\hat{T}_{12}-\hat{T}_1)} \left(e^{\mu(A)\hat{T}_1} + B_\mu(\hat{T}_1, A)\|K\| \right) \\ &+ \left(e^{\mu(A)(\hat{T}_{12}-\hat{T}_1)} + B_\mu((\hat{T}_{12} - \hat{T}_1), A)\|K\| \right) (\mu(A) + \|B\|\|K\|)\, e^{\mu(A)(\hat{T}_1)}, \end{aligned}$$

which has an unique critical point (the maximum) in $\hat{T}_1 = \hat{T}_{12}/2$ and, logically, $\hat{T}_2 = \hat{T}_1$. If we now consider the period $\hat{T}_1 + \hat{T}_2 + \hat{T}_3 = \hat{T}_{13}$, we can use the above argument to write $\hat{T}_1 = (\hat{T}_{13} - \hat{T}_3)/2$ and $\hat{T}_3 = (\hat{T}_{13} - \hat{T}_1)/2$. Hence, $\hat{T}_1 = \hat{T}_2 = \hat{T}_3 = \hat{T}_{13}/3$. Finally, we can use the argument recursively to prove that $c(\hat{T}_1, ..., \hat{T}_n) \leq c(T_s/n, ..., T_s/n)$. □

References

1. Mansano, R.K.; Godoy, E.P.; Porto, A.J. The Benefits of Soft Sensor and Multi-Rate Control for the Implementation of Wireless Networked Control Systems. *Sensors* **2014**, *14*, 24441–24461. [CrossRef] [PubMed]
2. Shao, Q.M.; Cinar, A. System identification and distributed control for multi-rate sampled systems. *J. Process Control* **2015**, *34*, 1–12. [CrossRef]
3. Albertos, P.; Salt, J. Non-uniform sampled-data control of MIMO systems. *Annu. Rev. Control* **2011**, *35*, 65–76. [CrossRef]
4. Cuenca, Á.; Salt, J. RST controller design for a non-uniform multi-rate control system. *J. Process Control* **2012**, *22*, 1865–1877. [CrossRef]
5. Cuenca, Á.; Ojha, U.; Salt, J.; Chow, M.Y. A non-uniform multi-rate control strategy for a Markov chain-driven Networked Control System. *Inf. Sci.* **2015**, *321*, 31–47. [CrossRef]
6. Kalman, R.E.; Bertram, J.E. General synthesis procedure for computer control of single-loop and multiloop linear systems (An optimal sampling system). *Trans. Am. Inst. Electr. Eng. Part II Appl. Ind.* **1959**, *77*, 602–609. [CrossRef]
7. Khargonekar, P.P.; Poolla, K.; Tannenbaum, A. Robust control of linear time-invariant plants using periodic compensation. *IEEE Trans. Autom. Control* **1985**, *30*, 1088–1096. [CrossRef]
8. Bamieh, B.; Pearson, J.B.; Francis, B.A.; Tannenbaum, A. A lifting technique for linear periodic systems with applications to sampled-data control. *Syst. Control Lett.* **1991**, *17*, 79–88. [CrossRef]
9. Li, D.; Shah, S.L.; Chen, T.; Qi, K.Z. Application of dual-rate modeling to CCR octane quality inferential control. *IEEE Trans. Control Syst. Technol.* **2003**, *11*, 43–51. [CrossRef]
10. Salt, J.; Albertos, P. Model-based multirate controllers design. *IEEE Trans. Control Syst. Technol.* **2005**, *13*, 988–997. [CrossRef]
11. Nemani, M.; Tsao, T.C.; Hutchinson, S. Multi-rate analysis and design of visual feedback digital servo-control system. *J. Dyn. Syst. Meas. Control* **1994**, *116*, 45–55. [CrossRef]
12. Sim, T.P.; Hong, G.S.; Lim, K.B. Multirate predictor control scheme for visual servo control. *IEE Proc.-Control Theory Appl.* **2002**, *149*, 117–124. [CrossRef]
13. Huang, X.; Nagamune, R.; Horowitz, R. A comparison of multirate robust track-following control synthesis techniques for dual-stage and multisensing servo systems in hard disk drives. *IEEE Trans. Magn.* **2006**, *42*, 1896–1904. [CrossRef]
14. Wu, Y.; Liu, Y.; Zhang, W. A discrete-time chattering free sliding mode control with multirate sampling method for flight simulator. *Math. Probl. Eng.* **2013**, *2013*, 865493. [CrossRef]
15. Salt, J.; Tomizuka, M. Hard disk drive control by model based dual-rate controller. Computation saving by interlacing. *Mechatronics* **2014**, *24*, 691–700. [CrossRef]
16. Salt, J.; Casanova, V.; Cuenca, A.; Pizá, R. Multirate control with incomplete information over Profibus-DP network. *Int. J. Syst. Sci.* **2014**, *45*, 1589–1605. [CrossRef]
17. Liu, F.; Gao, H.; Qiu, J.; Yin, S.; Fan, J.; Chai, T. Networked multirate output feedback control for setpoints compensation and its application to rougher flotation process. *IEEE Trans. Ind. Electron.* **2014**, *61*, 460–468. [CrossRef]
18. Khargonekar, P.P.; Sivashankar, N. H2 optimal control for sampled-data systems. *Syst. Control Lett.* **1991**, *17*, 425–436. [CrossRef]
19. Tornero, J.; Albertos, P.; Salt, J. Periodic optimal control of multirate sampled data systems. *IFAC Proc. Vol.* **2001**, *34*, 195–200. [CrossRef]
20. Kim, C.H.; Park, H.J.; Lee, J.; Lee, H.W.; Lee, K.D. Multi-rate optimal controller design for electromagnetic suspension systems via linear matrix inequality optimization. *J. Appl. Phys.* **2015**, *117*, 17B506. [CrossRef]
21. Lee, J.H.; Gelormino, M.S.; Morarih, M. Model predictive control of multi-rate sampled-data systems: A state-space approach. *Int. J. Control* **1992**, *55*, 153–191. [CrossRef]
22. Mizumoto, I.; Ikejiri, M.; Takagi, T. Stable Adaptive Predictive Control System Design via Adaptive Output Predictor for Multi-rate Sampled Systems. *IFAC-PapersOnLine* **2015**, *48*, 1039–1044. [CrossRef]
23. Geng, Y.; Liu, B. Guaranteed cost control for the multi-rate networked control systems with output prediction. In Proceedings of the IEEE International Conference on Information and Automation, Lijiang, China, 8–10 August 2015; pp. 3020–3025.

24. Carpiuc, S.C.; Lazar, C. Real-time multi-rate predictive cascade speed control of synchronous machines in automotive electrical traction drives. *IEEE Trans. Ind. Electron.* **2016**, *63*, 5133–5142. [CrossRef]
25. Roshany-Yamchi, S.; Cychowski, M.; Negenborn, R.R.; De Schutter, B.; Delaney, K.; Connell, J. Kalman filter-based distributed predictive control of large-scale multi-rate systems: Application to power networks. *IEEE Trans. Control Syst. Technol.* **2013**, *21*, 27–39. [CrossRef]
26. Anta, A.; Tabuada, P. On the minimum attention and anytime attention problems for nonlinear systems. In Proceedings of the 49th IEEE Conference on Decision and Control (CDC), Atlanta, GA, USA, 15–17 December 2010; pp. 3234–3239.
27. Donkers, M.C.F.; Tabuada, P.; Heemels, W.P.M.H. Minimum attention control for linear systems. *Discret. Event Dyn. Syst.* **2014**, *24*, 199–218. [CrossRef]
28. Gupta, V. On an anytime algorithm for control. In Proceedings of the 48th IEEE Conference Decision and Control (CDC), Shanghai, China, 16–18 December 2009; pp. 6218–6223.
29. Quevedo, D.E.; Ma, W.J.; Gupta, V. Anytime control using input sequences with Markovian processor availability. *IEEE Trans. Autom. Control* **2015**, *60*, 515–521. [CrossRef]
30. Aranda-Escolástico, E.; Guinaldo, M.; Cuenca, Á.; Salt, J.; Dormido, S. Anytime Optimal Control Strategy for Multi-Rate Systems. *IEEE Access* **2017**, *5*, 2790–2797. [CrossRef]
31. Guinaldo, M.; Sánchez, J.; Dormido, S. Event-based Control for Networked Systems: From Centralized to Distributed Approaches. *RIAI-Rev. Iberoam. Autom. Inform. Ind.* **2017**, *14*, 16–30. [CrossRef]
32. Loan, C.V. The sensitivity of the matrix exponential. *SIAM J. Numer. Anal.* **1977**, *14*, 971–981. [CrossRef]
33. Khalil, H. *Nonlinear Systems*, 3rd ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
34. Hazan, E. Introduction to Online Convex Optimization. *Found. Trends Optim.* **2016**, *2*, 157–325. [CrossRef]
35. Sala, A.; Cuenca, Á.; Salt, J. A retunable PID multi-rate controller for a networked control system. *Inf. Sci.* **2009**, *179*, 2390–2402. [CrossRef]
36. Upton, E.; Halfacree, G. *Raspberry Pi User Guide*; John Wiley & Sons: Hoboken, NJ, USA, 2014.
37. Coley, G. *Beaglebone Black System Reference Manual*; Texas Instruments: Dallas, TX, USA, 2013.
38. Chacon, J.; Saenz, J.; Torre, L.D.L.; Diaz, J.M.; Esquembre, F. Design of a Low-Cost Air Levitation System for Teaching Control Engineering. *Sensors* **2017**, *17*, 2321. [CrossRef] [PubMed]