A Unified Event-based Control Approach for FOPTD and IPTD Processes Based on the Filtered Smith Predictor

Ángel Ruiz[1], Manuel Beschi[2], Antonio Visioli[3], Sebastián Dormido[4] and Jorge E. Jiménez[1]

Abstract

A new unified design of an event-based PID control architecture for self-regulating and integral processes is investigated in this work. The design is based on the Symmetrical-Send-On-Delta sampling technique (SSOD) and on the filtered Smith predictor (FSP). In particular, the conditions to achieve robust stability to limit cycles are studied from the point of view of the filter parameters through the describing functions theory. In this context, a simple frequency domain-based tuning methodology to address the cases of first-order-plus-dead-time and integrator-plus-dead-time processes is proposed. With this method, the properties of the filter parameters are also explored to cope with the set-point tracking and the load disturbance rejection tasks. The effectiveness of the general approach is evaluated by simulations and experimental tests.

Index Terms

Event-based PID control, self- and non self-regulating processes, filtered Smith predictor, tuning.

I. Introduction

Until now, the industrial applications of process control have been practically handled by time-based feedback control systems, most of them governed by standard proportional-integral-derivative (PID) controllers. A significant factor of their success is the existence of numerous tuning rules and design techniques [1], which allow the user to provide a satisfactory performance for many processes with a relatively easy design [2].

Despite PID control is well established as a technology, the advancement of industrial processes continuously demands for new developments. A significant challenge has arisen in process control because of the more and more use of wireless sensors and actuators, which conform the well-known networked control systems (NCS) (see [3], [4]). Such systems are a valuable example of distributed architectures that suffer from the lack of efficient sampling techniques in order to minimize the effect of data dropout and stochastic time delays, while saving energy and bandwidth (essentially in battery-powered systems) [5]. In this context, it makes sense to transmit information between agents (sensors, actuators controllers, networks coordinators, etc.) when something significant has occurred and this calls for the techniques based on events [6], [7]. As demonstrated in recent works in the field of event-based PID control [8]–[12], a satisfactory trade-off between information exchange and the closed-loop performance can be achieved when a tight tracking control is not of main concern and a small steady-state error is acceptable.

In event-based control it is not the progress of time but the occurrence of events which forces agents to transmit information. Although this is conceptually simple, the design of an efficient control scheme and of a systematic tuning method poses a challenging problem. Note that the event condition may be any mathematical function (see [13], [14] for typical event-based conditions), which can be combined with different techniques (event-based, self-triggered, sporadic, etc.), obtaining different architectures and nonlinear responses [15], [16]. In particular, when wireless communications are considered, the protocol constraints and the energy consumption need to be considered as a part of the control loop design [17].

Even so, some relevant consensus has been encountered. The Send-On-Delta (SOD) algorithm is the most employed in practice (see [4], [18]–[20]) because of its simplicity of implementation. It consists in computing a new sample when a signal changes more than a predefined threshold. Additionally, the use of hybrid sampling schemes, that is, those that combine time-based and event-based sampling schemes in agents involved in the closed-loop, have become one of the most extended design approaches. By this way, the event-based behavior can be characterized using the well-known automatic discrete control theory. Based on the synergy between both frameworks, many theoretical results for event-based PID control have been recently presented and tested on real plants, but unfortunately they are based on particular solutions either for self-regulating or non self-regulating (integral) processes. Indeed, the case of integral processes has actually not been properly addressed until now.

In this work, a unified framework for self-regulating and integral processes is proposed. It is based on the general model discussed in [21] under the assumption that the event-based sampling technique is applied on the error signal [9], [10], [22].

[1]Ángel Ruiz and Jorge E. Jiménez are with the Department of Computer Science and Numerical Analysis, University of Córdoba (UCO), Spain. aruiz@uco.es, jjimenez@uco.es

[2]Manuel Beschi is with the ITIA-CNR, Italy. manuel.beschi@itia.cnr.it

[3] Antonio Visioli is with the Dipartimento di Ingegneria Meccanica e Industriale, University of Brescia, Italy. antonio.visioli@unibs.it

[4]Sebastián Dormido is with the Departamento de Informática y Automática, Universidad Nacional de Educación a Distancia (UNED).Spain.sdormido@dia.uned.es

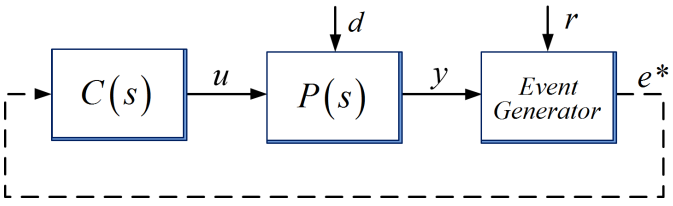

Fig. 1. General scheme of the event-based control system. Solid lines denote time-based signals, and dashed lines event-based signal transmission.

Note that, although such an approach has been effectively applied to self-regulating processes with delay [23]–[25], it is not inherently suitable to be applied to integral processes.

In this sense, a new implementation is proposed in this paper. This results from the redesign of the architecture proposed in [24] in order to obtain a unified approach able to cope with the most common (both self- and non self-regulating) industrial processes. The new scheme is based on the structure of the filtered Smith predictor (FSP) [26]. The advantages of this structure in comparison with the original Smith predictor one (implemented in [24]) rely on the two additional filters considered, which allow the control system to use a common design to simultaneously handle the performance and the robustness properties for both kind of processes. Under this scenario, the performance and robustness of the proposed event-based structure is analyzed for a significant set of self-regulating and integral processes by providing a tuning methodology. In particular, the Describing Function (DF) approach [27] is used to determine the design conditions of the filters that robustly guarantee the avoidance of limit cycles for each particular process.

In this context, some theoretical analyses are provided, which are supported by simulations, and experimental results.

The paper is organized as follows. In Section 2, the event-based control system is presented and its behavior is described. In Section 3, the robust stability and limit cycles problems are analyzed. In Section 4, the tuning procedure is described. In Section 5, practical issues on the compensation of load disturbances are introduced. The results from simulation are presented in Section 6 while experimental results are shown in Section 7. Finally, Section 8 presents the main conclusions.

II. Control architecture

The scheme of the event-based control system (EBCS) analyzed in this work is shown in Fig. 1. In line with the general model [21], the architecture considers three functional/main blocks: the controller, the process and the event generator. Their tasks are described in the following part.

The dynamic behavior of the stable and integral processes under study can be described (or approximated), respectively, by a first order plus time delay (FOPTD) transfer function as:

$$P_s(s) = \frac{K}{\tau s + 1} e^{-Ls} = G_s(s)\, e^{-Ls} \tag{1}$$

or by an integrator plus time delay (IPTD) transfer function as:

$$P_u(s) = \frac{K}{s} e^{-Ls} = G_u(s)\, e^{-Ls} \tag{2}$$

where K is the static gain (velocity gain in the case of P_u), τ is the equivalent time constant of the stable process, and L is the apparent dead time. The process is assumed to be affected by non-modeled step disturbances $d(t)$ at the input $u(t)$ and a constant set-point value $r(t)$ is applied. In both cases $G(s)$ or $G_u(s)$ represent the dead-time free part of the process.

Remark 1. Simple models are essential in the process industry. There are several claims that in practice low-order models coupled with dead times are sufficient for most purposes in control system design and analyses. For this reason, a relevant range of processes can be well represented as FOPTD and IPTD models [2].

The structure of the controller block is different depending on the process considered. In the stable case (1), it is implemented by a discretized version of a continuous PI controller (since $G_s(s)$ is a first order transfer function, a PI controller achieves a good compromise between performance and simplicity):

$$C(s) = K_P \left(1 + \frac{1}{T_i s}\right) \tag{3}$$

In the case of (2), a standard P controller is chosen:

$$C(s) = K_P \tag{4}$$

The block of the event generator implements the event-based sampling technique, and therefore characterizes the overall control loop as an event-based control system. According to the approach presented in [24], its structure can be given by the

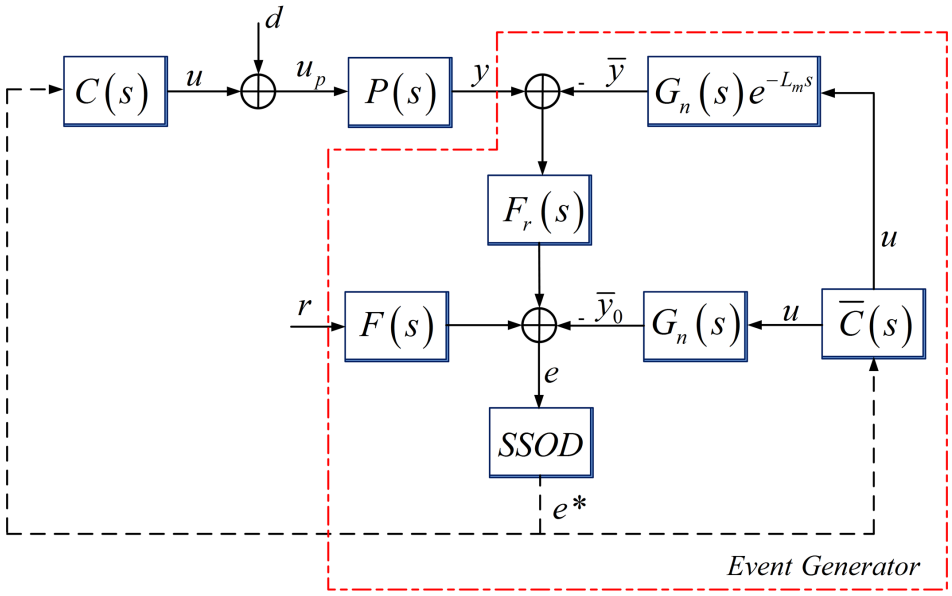

Fig. 2. Event-based control scheme including the signals and the details of the event generator.

combination of a dead-time compensator (DTC) and an event-based sampler, which act as two complementary functional parts. The DTC aims at reducing the effect of the process delay such that the relationship between the timing of events and the process behavior is more predictable. In this work, the DTC is implemented with the structure of the Filtered Smith predictor [26]. A detailed description of the event generator is shown in Fig. 2. Here, $G_n(s)e^{-L_m s}$ is the model of the process, $G_n(s)$ represents the dead-time-free part and $\overline{C}(s)$ is a model of the primary controller ($\overline{C}(s) = C(s)$). The structure considers two filters, known respectively, as the set-point filter, $F(s)$, and the predictor filter $F_r(s)$. To be consistent with the robustness results of [26], the transfer function of the set-point filter is assumed to be the unity (because unstable processes are not considered) and the predictor filter is defined as

$$F_r(s) = \frac{(T_r s + 1)(\beta_1 s + 1)}{(T_0 s + 1)^2} \tag{5}$$

where the parameter β_1 is dependent on the structure of the controller, to be particularized as $\beta_1 = T_m\left[1 - (1 - T_0/T_m)^2 e^{-L_m/T_m}\right]$ in the case of (1) and as $\beta_1 = 2T_0 + L_m$ in the case of (2). Similarly, the controller parameters are established from the model and filter parameters as (6) and (7), for the case of the self-regulating process (1) and the integer one (2), respectively.

$$T_i = T_m \quad K_P = T/K_m T_r \tag{6}$$

$$K_P = 1/K_m T_r \tag{7}$$

As pointed out in [26], this structure provides a good trade-off between the robustness and load disturbance properties, while simplifying the design using only T_0 and T_r as adjustable parameters.

Remark 2: The event generator considers a model of the controller to obtain information about the control action $u(t)$ without the need of extra-communications between units. Thus, under the assumption that communication time delays can be neglected, the control signal determined by both controllers (process controller and model controller) can be considered to be synchronized. This can be assumed because they have the same transfer function and they are activated by the same event-based error samples (which are sent at the same triggering instant to both controllers).

The event-based sampler block receives the measure of the error signal from the previous entity and computes the variable error samples that are sent to the controller when the event condition is satisfied. As observed, the error signal is calculated as the result of the difference between the filtered prediction error, the delay-free model output and a constant set-point value r. In this work the triggering algorithm is based on the SSOD sampling technique [28], which is described in the next section.

From a practical point of view, the EBCS functional blocks could be implemented in one or more physical entities, for instance, intelligent on board devices, microcontrollers, automatons, computer-based systems, etc. In the case of several

entities, the data has to be sent from one to each other on a network which implies more efforts than the data exchanging into a single machine, especially when they are battery powered. For this reason, it is recommended to use event-triggered data exchanging for all the signals which are sent between two machines and normal time-driven sampling for the data which are computed by an unique machine (for instance to link the blocks that compose the event-generator). It has to be noted that this is an appealing architecture for networked control systems, because the sensor unit (in this case, the event-generator represents this function) can be battery powered (thus, a reduction of the communications can increase the battery life), whereas the control unit normally requires an external power supply so that the power consumption reduction is not as critical as in the other unit.

A. Event-based sampling

The event-triggered data exchanging is performed using the SSOD event-based sampling algorithm, which represents a special case of the send-on-delta (SOD) sampling technique (see [18]). According to the standard algorithm, the output is computed based on two predefined parameters: an event threshold $\Delta \in \mathbb{R}^+$, and the internal state of the algorithm $j \in \mathbb{Z}^+$. Thus, if $e(t)$ is the input of the algorithm, the output, denoted as $e^*(t)$, is forced to satisfy the relation $e^*(t) = j\Delta$, that is, a multiple integer j of Δ. In this way, the events are triggered when consecutive levels are crossed by the error signal, which means that the sampled signal changes its value to the upper or lower quantization level when the input signal $e(t)$ increases or decreases more than Δ. This behavior can be mathematically described as

$$e_j^*(t) = SSOD\left(e_j(t);\Delta\right) = \begin{cases} (j+1)\Delta & \text{if} \quad e(t) \geq (j+1)\Delta \quad \text{and} \quad e^*\left(t^-\right) = j\Delta \\ j\Delta & \text{if} \quad e(t) \in [(j-1)\Delta, (j+1)\Delta] \quad \text{and} \quad e^*\left(t^-\right) = j\Delta \\ (j-1)\Delta & \text{if} \quad e(t) \leq (j-1)\Delta \quad \text{and} \quad e^*\left(t^-\right) = j\Delta \end{cases} \tag{8}$$

and a graphical representation is shown in Fig. 3.

Remark 3: The behavior of the algorithm can be considered as a generalization of a relay with hysteresis, where there are an infinite number of thresholds and admitting a simple representation as an hybrid automaton described by the set of parameters (j, Δ) [19].

B. Closed-loop response

The closed-loop response of the event-based system is mainly affected by the occurrence and the timing of events. As shown in Fig. 2, the control loop has a time-driven behavior between the controller and the event generator, whereas the transmissions between the event-generator and the controller are performed using the event-based sampling technique. Thus, a new sample is not triggered and sent to the controller if the event condition is not satisfied, which interrupts the continuity of the feedback loop and characterizes the overall scheme as an event-based control system. Therefore, in the nominal case, the behavior of the system can be described as a sequence of open-loop responses composed by the controller and the process plant since the feedback loop is only closed through the event generator at event triggering instants.

With this background, the behavior of the event-based system can be summarized as follow: each time the instantaneous error signal $e(t)$ exceeds a specific threshold defined by (5), a new event is triggered and therefore, a new sample $e^*(t)$ is sent to the controller. The value of the sample is defined by the SSOD algorithm in accordance with the current state j. According to Fig. 2, the response of the instantaneous error signal is elaborated from the difference between the process and the model after filtering, and the delay-free model with respect to the value of the constant set-point r. Thus, under the assumption of an accurate model of the process, the delay produced in its response can be compensated by the predictor and the event samples are triggered at deterministic instants. In case of plant-model mismatches, events are triggered with a different timeline because the dynamic of the error signal is altered and, consequently, the process response cannot be predicted with precision. Thus, the key point in the closed-loop system behavior is represented by the dynamic of the error signals between events, which is characterized by the effect of the predictor filter over the modeling errors and its interaction with the open loop response of the controller $C(s)$.

III. Robust stabilization of the EBCS

The stability properties of an EBCS can be derived from the analysis of the behavior of its equilibrium points [10]. The analysis is based on the properties of the describing function (DF) of the nonlinearity introduced by the SSOD algorithm and it will be exploited for the tuning of the controller parameters.

The describing function is an approximate method, classically used to analyze nonlinear systems by characterizing the limit cycles experimented in the presence of nonlinear elements [29]–[31]. Alternatively, many authors have used the DF as a frequency-domain-based modeling method for autotuner systems and, particularly, in the field of event-based control, only a few works have considered this methodology for analyses [23], [32]. The relevance of this method is founded on the adaptation of the Nyquist stability test to detect the presence of limit cycles in a nonlinear system, providing a prediction of its approximate amplitude and frequency.

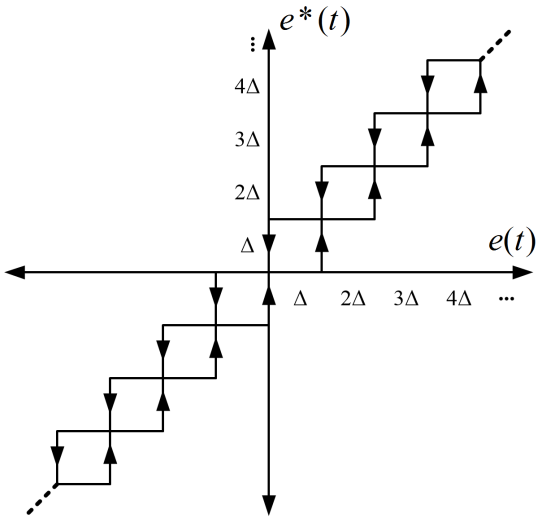

Fig. 3. Quantization relationship between $e(t)$ and $e^*(t)$.

In the context of this work, it is clear that the DF method can become a valuable approach to analyze the EBCS since its stability properties are related to the behavior of the limit cycles. It has to be noted that the method of the DF represents an approximation which is based on some assumptions, concretely, that the controlled system has low-pass filtering properties and responds principally to the fundamental Fourier component [27]. This implies that the estimation error depends on how the low-pass filtering assumption is verified, being smaller for very low-pass responses. As it will be explained later on, the filtered predictor will introduce in the design the low-pass component so that this hypothesis will be verified. In the following section, the basics of DF and their application to the tuning of the proposed scheme are described.

A. The Describing function analysis

In order to consider the describing function analysis, the control system in Fig. 2 is modified to obtain the equivalent scheme in Fig. 4, where the nonlinear and the linear parts are described by separated blocks interconnected through the feedback loop. Here, $G_{ol}(s)$ represents the frequency domain open-loop dynamics defined by the combination of the controller, the process and the elements of the event-generator excluding the SSOD sampling block, which defines the nonlinear element. For the particular case of a standard parametric transfer function of the process as (1) or (2) subject to uncertainty, the equivalent transfer function could be obtained by considering the following expression:

$$G_{ol}(s) = C(s)\,G_{eq}(s) = C(s)\,F_r(s)\left[P(s) - P_n(s) + G_n(s)/F_r(s)\right] \tag{9}$$

where $P_n(s) = G_n(s)\,e^{-L_m s}$

As detailed in [27], the describing function of a nonlinear element can be defined as the ratio of the fundamental component of the output with respect to the magnitude of a sinusoidal input signal applied to the nonlinear element. Thus, considering the input $x(t) = A\sin(wt)$ to the nonlinear element, the output can be expressed using Fourier series as:

$$y(t) = \frac{c_0}{2} + \sum_{k=1}^{\infty}\left[c_n\cos(nwt) + d_n\sin(nwt)\right] \tag{10}$$

where

$$c_n = \frac{1}{\pi}\int_0^{2\pi} y(t)\cos(nwt)\,d(wt) \tag{11}$$

$$d_n = \frac{1}{\pi}\int_0^{2\pi} y(t)\sin(nwt)\,d(wt) \tag{12}$$

For an odd nonlinearity c_0 is equal to zero. Then, using the fundamental component of $y(t)$, that is, $y_1(t) = c_1\cos(wt) + d_1\sin(wt)$, the nonlinearity is represented with the describing function as

$$N(A,w) = \frac{d_1 + jc_1}{A} \tag{13}$$

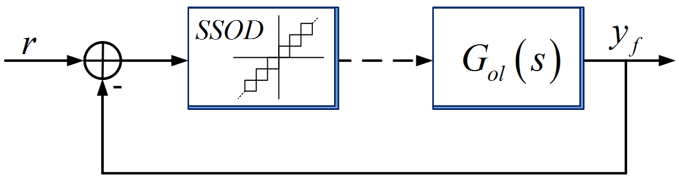

Fig. 4. Equivalent closed-loop system.

Thus, from (9), the characteristic equation of the system shown in Fig. 4 can be written as

$$1 + N(A)\,G_{ol}(s) = 0 \tag{14}$$

where the parameter w is dropped from notation. Note that, in practice, the nonlinearity $N(A,w)$ does not involve energy storage or dissipation so that it can be expressed only as a function of the amplitude in the form $N(A)$. From (14), the following equation can be obtained:

$$G_{ol}(jw) = -\frac{1}{N(A)} \tag{15}$$

The solutions of (14) describe all the feasible characteristic limit cycles. This equation can be solved with graphical methods by finding the intersections of $G(jw)$ and $-1/N(A)$ in the complex plane. Additionally, by analyzing (15), it can be easily understood that the problem of the existence limit cycle solutions can be translated into a Nyquist stability problem, where there is not a single critical point $(-1,0)$, but a locus of critical points given by the complex values of $-1/N(A)$.

Under this scenario, the DF of the nonlinear SSOD block can be computed by following the procedure of [27] as:

$$N(A) = \frac{2\Delta}{\pi A}\left[1 + \sqrt{1-\left(m\frac{\Delta}{A}\right)^2} + 2\sum_{k=1}^{m-1}\sqrt{1-\left(k\frac{\Delta}{A}\right)^2}\right] - jm\frac{2\Delta^2}{\pi A^2} \tag{16}$$

where $m = \left\lfloor \frac{A}{\Delta} \right\rfloor$ represents the number of output levels for a sinusoidal input with amplitude A. The polar plot of $\frac{1}{N(A)}$ is shown in Fig. 5, where the equation (16) is evaluated in the interval $\frac{\Delta}{A} \in [0,1]$.

From Fig. 5, it can be easily understood that a characteristic limit cycle will appear depending on the level and the point of crossing with the polar plot of the open loop response (9). Therefore, the closed-loop stability, in the sense of the existence of limit cycles, can be graphically analyzed by computing such graphics.

In next section, the problem of the robust stabilization of the EBCS is considered for the particular case of the processes (1) and (2).

B. DF-based Robust Stability

A standard stability problem in robust control theory consists in finding a constraint on the infinity norm of the weighted complementary sensitivity function that guarantees stability [33]. For the particular case of the proposed EBCS, the constraint is represented in the Nyquist diagram by the bound of the uncertainties that avoid the existence of solutions that intersect with the characteristic describing function of the SSOD block (limit cycle solutions).

Consider that the process is represented by a family of plants $P(s)$ described by a multiplicative unstructured uncertainty, such that $P(s) = P_n(s)[1 + \delta P(s)\Lambda(s)]$, where $P_n(s)$ is the nominal model, $\delta P(s)$ is the uncertainty weighting frequency function and $\Lambda(s)$ is an unknown stable transfer function with $\|\Lambda\|_\infty < 1$. As stated in [33], the complementary sensitivity function of a system can be invoked to test the robust stability conditions, which from Fig. 4 can be determined for the nominal model $P_n(s)$ as:

$$T(s) = \frac{F_r(s)C(s)P_n(s)}{1 + C(s)G_n(s)} \tag{17}$$

In this sense, considering that the nominal system is stable, the necessary and sufficient condition for robust stability of the standard FSP is:

$$\overline{\delta P}(\omega) < \frac{|1 + C(j\omega)G_n(j\omega)|}{|F_r(j\omega)C(j\omega)P_n(j\omega)|}, \quad \forall\,\omega > 0 \tag{18}$$

where $\overline{\delta P}(\omega)$ is the multiplicative norm-bound uncertainty that verifies $|\delta P(j\omega)| \le \overline{\delta P}(\omega)$, which finally yields

$$\|T(s)\,\delta P(s)\|_\infty < 1 \tag{19}$$

From (19), a sufficient condition to obtain robust stability without limit cycles can be obtained by constraining the infinity norm as it is stated in the following proposition.

Proposition 1: An EBCS according to the proposed scheme in Fig. 2 is robustly stable without limit cycles if the following sufficient condition is satisfied:

$$\left\| \tilde{T}(s)\,\delta P(s) \right\|_{\infty} < 1 \tag{20}$$

where

$$\tilde{T}(s) = \frac{F_r(s)\,C(s)\,P_n(s)}{N_s + C(s)\,P_n(s)} \tag{21}$$

and where N_s represents the farthest point away from the critical point, that is $N_s = N_a + jN_b \in N(A)$, such that $|N_s| = \sup|1 + N(A)|$.

Proof: If we develop the equation (9) and we introduce the module condition, it can be easily demonstrated that intersections do not exist if $|G_{ol}(s)| < \inf\left|-\frac{1}{N(A)}\right| = |-N_s(s)|$ is satisfied for all frequencies ω. From this, we have:

$$|P(s) - P_n(s)| < \frac{|N_s(s)+C(s)G_n(s)|}{|F_r(s)C(s)|}$$

Dividing both sides by the nominal model, we obtain:

$$|\delta P(s)| < \frac{|N_s(s)+C(s)G_n(s)|}{|F_r(s)C(s)G_n(s)|}$$

that is equivalent to

$$\frac{|F_r(j\omega)C(j\omega)G_n(j\omega)|}{|N_s(j\omega)+C(j\omega)G_n(j\omega)|}\,|\delta P(\omega)| < 1, \quad \forall\,\omega > 0$$

thus obtaining the infinity norm constraint

$$\left\| \delta P(s)\,\tilde{T}(s) \right\|_{\infty} < 1$$

that demonstrates the validity of the Proposition 1.

Hence, from the DF analysis, the original problem of stability for the proposed event-based controller is transformed into a nonlinear robust control problem that could be accomplished by finding the needed constraint on the infinity norm of the complementary sensitivity function as illustrated. It is worth remembering that, in order to be consistent with robustness results of [26], the adjustable tuning parameters used to cope with the problem of limit cycles and robustness will be represented by the coefficients of the filter $F_r(s)$, T_0 and T_r (instead of the conventional tuning parameters focused on the controller, which are specified according to (6)-(7)). In the next section, the conditions for tuning these parameters in order to obtain asymptotic robust stability are addressed and the tuning framework is introduced.

IV. Tuning procedure

As for all industrial control systems, in addition to the achievement of a satisfactory performance, it is important to ensure that the controller is easy to design and for this reason a clear (robust) tuning procedure has to be sought.

Because of the nonlinear nature of the EBCS, the tuning of the filter can make the system unstable, to reach an equilibrium point without limit cycles, or admit a stable limit cycle. For this reason, the proper knowledge of system properties is essential to adequately tuning the filters in order to avoid limit cycles with a certain degree of robust performance. In this section, we propose a simple method to compute the space of solutions that provide the characteristic behaviors of the EBCS, that is, the instability region, the stable limit cycle region or the asymptotic stability region.

According to the formulation in the previous sections, the three characteristic regions that describe the behavior of the equilibrium point for the EBCS can be represented as a function of the uncertainty bounds and of the filter parameters. They can be derived in two steps by using the conditions (19)-(20). Firstly, the stable feasible region (namely, the region concerning to stable limit cycles and asymptotic solutions) can be separated from the unstable solution region via the standard robust stability problem derived from (19). Secondly, this information can be used to constrain the infinity norm in order to adequately delimit the space of solutions for T_0 and T_r that avoid a limit cycle for the family of uncertain plants (according to condition (20)). In this way the three regions can be clearly delimited for tuning purposes. The procedure is described hereafter, but for the sake of clarity, only the process plant (1) is considered in the description. It has to be emphasized, however, that the same procedure can be trivially particularized for the process plant (2).

Define a nominal process described by the set of parameters (K, τ, L) and a set of structured uncertainty bounds $K \in \left[\underline{K}, \overline{K}\right]$, $\tau \in \left[\underline{\tau}, \overline{\tau}\right]$ and $L \in \left[\underline{L}, \overline{L}\right]$ that determine all the family of process plants (only (K, L) in the case of (2)). The model of the process $P_n(s) = G_n(s)e^{-L_m s}$ is assumed to be described by the values of the nominal plant, namely $(K_m, \tau_m, L_m) = (K, \tau, L)$. Now, consider the limits of interest for the parameters of the filter to be used in the computation of the tuning regions as $T_0 \in \left[T_0^{min}, T_0^{max}\right]$ and $T_r \in \left[T_r^{min}, T_r^{max}\right]$.

With this information, by suitably gridding the search space of T_0 and T_r, the robustly stable solutions can be computed by evaluating the condition (19) for each value of the controller parameters. To this aim, it is necessary to obtain the transfer function that represents the upper bound of the relative error δP (the uncertain function), which can be derived using a representative family of uncertain plants. As remarked in [33], a reasonable bound approximation is obtained by using all the combinations of the values of the vertex of the uncertainties, that is all possible combinations in the process plant structure that define the family $\mathcal{F} = \left\{\forall P(s, K, \tau, L) | K \in \left\{\underline{K}, \overline{K}\right\}, \tau \in \left\{\underline{\tau}, \overline{\tau}\right\}, L \in \left\{\underline{L}, \overline{L}\right\}\right\}$. Note that the bound is independent from the filter setting so that it has to be computed once in order to evaluate all the space of parameters for the filter values. In this way, the solution space delimited by $T_0 \in \left[T_0^{min}, T_0^{max}\right]$ and $T_r \in \left[T_r^{min} T_r^{max}\right]$ is reduced to the set of values that provides robust stability according to the standard condition (19).

Next, the stable limit cycle region can be extracted from the global stable region by detecting the possible solutions that produce limit cycles. This task is performed by imposing the infinity norm constraint derived in (20). The less conservative constraint which guarantees the robust stability can be obtained by defining N_s with the complex coordinates $-0.78 - 0.78j$. Note that this point represents the point belonging to the DF with the highest radio-vector as can be observed in the polar plot introduced in the graphical example of Fig. 6. Thus, solutions that satisfy condition (20) will not produce crossings between $G_{ol}(j\omega)$ and the DF, avoiding the generation of limit cycles. In this sense, a second iterative procedure can be carried out by checking condition (20) on the set of stable solutions, where only the solutions that satisfy it belong to the asymptotic stable region, whereas the rest of solutions belongs the stable limit cycle region.

Remark 4: It can be easily understood that the region of asymptotically stable solutions can be computed directly without first computing the overall stable feasible region. However, distinguishing all the three characteristic regions can be interesting for the tuning procedure as in this way the designer can evaluate explicitly the relative stability of the system (that is, the distance of the selected tuning from the unstable region). This is especially relevant if limit cycles are admissible for a given application.

From a practical point of view, this method provides a simple map of the characteristic behavior of the EBCS, where it is only necessary to specify the uncertainty bound. As it will be seen from the results of this approach, under certain prior knowledge of uncertainties bound, the level of robustness to limit cycles can be increased arbitrarily using the information provided by the characteristic tuning regions. This is very useful because the values of the filter can be easily derived to handle the trade-off between the rejection of load disturbances and robust stability by selecting the adequate set of values. It is worth stressing the simplicity of the proposed method which can be precomputed for each particular case without the need of complex calculations or subroutines in order to develop the tuning procedure. As already mentioned, the extension

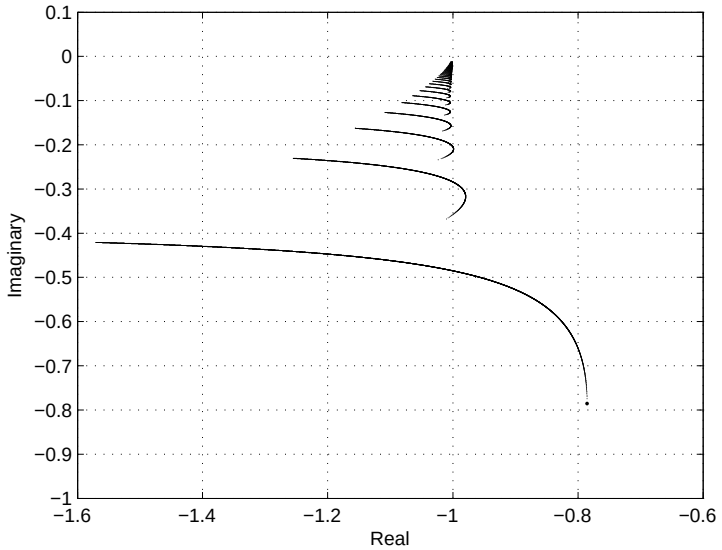

Fig. 5. Polar plot of $-1/N(A)$.

to the case of integral processes (2) can be trivially derived using the same approach.

V. Disturbance compensation/estimation algorithm

The event-based control of integral processes as proposed in this work requires an additional (feedforward) compensation action to deal with step load disturbances. Due to the quantized nature of control actions derived from the use of a P controller in the scheme of Fig. 2, the unquantized part of a load disturbance cannot be compensated and, consequently, a bimodal limit cycle surely arises. To clarify this point, we can consider the simplified version of the scheme in Fig. 4 where it is assumed that the process plant behaves as the nominal case. In this situation, the simplification condition $P(s) = P_n(s)$ is satisfied and, therefore, the feedback signal is given by the output of the dead-time-free model $G_{ol}(s) = G_n(s)$.

By analyzing the scheme and by remembering that control actions are computed as $u(t) = K_P e^*(t) = K_P j\Delta$, it can be easily seen that an equilibrium exist only if $r = K_P j\Delta + D$. Then, if an unquantized part exists, because of the integral behavior of the process, a limit cycle with period T appears, which involves two consecutive states denoted as $j_l + 1$ and j_l. Denoting as t^* the time interval where $j = j_l + 1$ is retained, the control action assumes only the values $K_P j_l$ and $K_P(j_l + 1)$, that is,

$$u(t) = \begin{cases} K_P(j_l + 1)\Delta & \text{if} \quad t \in [0, t^*] \\ K_P j_l \Delta & \text{if} \quad t \in [t^*, T] \end{cases} \tag{22}$$

Under an established limit cycle, by definition, the trajectories of the process and of the controller are periodic. If we consider now the integrated error $IE(t)$, its Laplace transform $IE(s)$ can be calculated as

$$IE(s) = \frac{R(s)}{s} - \frac{K}{s^2}(U(s) + d) \tag{23}$$

which corresponds to the following differential equation:

$$\dot{IE}(t) = \dot{r}(t) - K(u(t) + d) \tag{24}$$

By integrating (24) over a period and remembering that, by hypothesis, r and d are constant and $IE(t)$ is periodic, we obtain:

$$\dot{IE}(T) - \dot{IE}(0) = -KdT - K\int_0^T u(t)\,dt = 0 \tag{25}$$

and finally:

$$\overline{u} = \frac{\Delta t^*}{T} + j_l\Delta = d \tag{26}$$

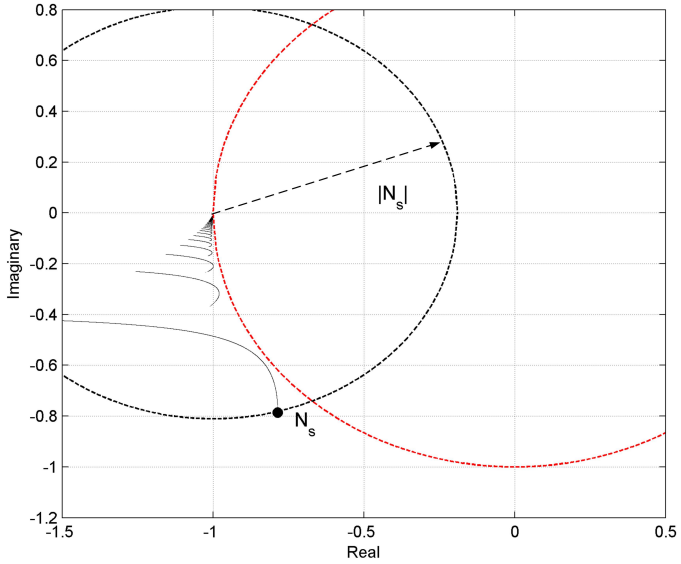

Fig. 6. Point with highest radio-vector of the SSOD describing function.

where $\overline{u}$ is the mean value of $u(t)$ during a period and $\frac{\Delta t^*}{T}$ is the "unquantized" part of the control action necessary to compensate the load disturbance. Thus, a constant feedforward action equal to $\overline{u}$ is added to the event-based feedback control action for integral processes in order to achieve a zero steady-state error when a constant load disturbance occurs. However, as it was pointed out in [10] for the PI-SSOD control scheme, when the controller is quantized, the exact compensation is virtually impossible in practice. In real plants, the trajectory tends to the limit cycles but does not reach it. Hence, the hypothesis of periodicity is not further preserved (it is only an approximation), therefore the exact compensation of the disturbances is not possible. For this reason, it it recommendable to include a small deadband in order to take into account the numerical errors in the calculus of t^* and T and the presence of noise, considering it is overbounded by Δ.

Remark 5: Limit cycles are defined as periodic oscillations of the unperturbed system, while stability analysis (in terms of convergence in a neighborhood around the equilibrium point) is still valid also in presence of bounded disturbances. However, for slow-varying piecewise constant input disturbances (namely, multiples of the limit cycle period) the limit cycles analysis can be applied without significant approximation errors. It is worth stressing that, in presence of a fast-varying disturbance signal (compared with the event-generator threshold), the number of events increases and the event-based control strategies become more and more similar to their equivalent time-driven control strategies.

VI. Simulation Results

In this section, simulation results are provided. In particular, the examples illustrate the robust design of the filter for a significant range of different process dynamics (both self-regulating and integral) in order to avoid the presence of limit cycles and the compensation of exogenous signals.

A. Illustrative Example 1

In the first simulation case, a series of experiments is proposed with the FOPTD process

$$P_s(s) = \frac{1}{s+1}e^{-4s} \tag{27}$$

where for each parameter (K, τ, L) an uncertainty of ± 20 % is considered, that is, $\underline{K} = 0.8$, $\overline{K} = 1.2$, $\underline{\tau} = 0.8$, $\overline{\tau} = 1.2$, $\underline{L} = 3.2$, $\overline{L} = 4.8$.

A.1. Computation of the stability regions: Assuming that the upper bound of the uncertainties is known, the methodology proposed in Section IV can be used to compute the robust solutions for the filter parameters (T_0, T_r).

The first step is to obtain the norm bound function $\overline{\delta P}(\omega)$. According to the analysis of Sections 3 and 4, $\overline{\delta P}(\omega)$ can be computed as $\sup \left| \frac{P_i(j\omega) - P_n(j\omega)}{P_n(j\omega)} \right|$, where P_i represents each one of the combinations that defines the family of uncertain process plants $\mathcal{F} = \left\{ \forall P(s,K,\tau,L) | K \in \left\{\underline{K}, \overline{K}\right\}, \tau \in \left\{\underline{\tau}, \overline{\tau}\right\}, L \in \left\{\underline{L}, \overline{L}\right\} \right\}$. The Bode magnitude plot of the transfer function employed is shown in Fig. 7 where it has been approximated by a second-order transfer function.

The graphic with the feasible solutions is shown in top-left plot of Fig. 8. Here, the three areas that characterize the behavior of the EBCS can be distinguished (as pointed out in Section III, it can be unstable, reach an equilibrium point, or admit a limit cycle). As a result of the proposed approach, the region for unstable solutions can be separated from the stable one (condition (19)), which, at the same time, can be divided into the stable limit cycle and the asymptotically stable regions (by using the proposition with the condition (20)). Note that, from this kind of graphic, a solution can be conveniently chosen from the stable regions in order to fulfill some closed-loop performance properties (as it will be discussed in the following examples). In this sense, an example of a stable design point is chosen for analysis.

The Nyquist plot of the family of plants $\mathcal{F}$ is also shown in Fig. 8 for the selected design point. As it can be observed, the Nyquist plots satisfy the condition of no limit cycles imposed. Right-side plots of Fig. 8 show the time domain responses for such a design point. Here, three cases of structured uncertainty are simulated such that the values of the parameters are $\left\{\overline{K}, \overline{\tau}, \overline{L}\right\}$, $\left\{\underline{K}, \underline{\tau}, \underline{L}\right\}$ and $\left\{\underline{K}, \underline{\tau}, \overline{L}\right\}$, with $P_n(s)$ given by (27). The value of Δ is set equal to 0.1. A unit set-point step is simulated at time $t = 0$ and a unit load disturbance step is applied at time $t = 40$. The proposition of robust avoidance of limit cycles can be verified from the obtained results and it can be also seen that the used robust design point describes a balanced solution in handling the set-point following and the load disturbance rejection tasks considering the uncertainty bounds.

A.2. Influence of filter tuning parameters: Next example illustrates the influence of the tuning parameters, considering the aforementioned process (27) with the same structured uncertainty. Note that T_r, T_0 and Δ represent the only three tuning parameters, because the parameters of the controller are characterized by the parameters of the model and by the coefficient T_r. The effects of the parameters T_r and T_0 are derived from the structure of the FSP. While T_r determines the performance with respect to the set-point step response, T_0 handles the trade-off between performance (in the load disturbance rejection task) and robustness of the control system. The parameter Δ concerns to the system precision and the number of events. The influence of this parameter will be discussed later on. As shown in Fig. 9, six sets of parameters belonging to the stable area are analyzed, which are respectively denoted by using a circle or a triangle with an specific index. In the case of the sets denoted with circles, they introduce increments in the parameter T_0, while T_r is kept constant. Conversely, the sets

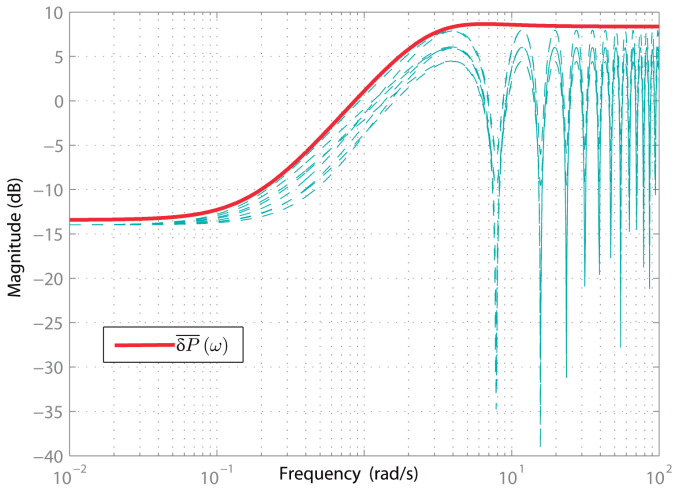

Fig. 7. Magnitude of the norm-bound transfer function $\overline{\delta P}(s) = (2.621s^2 + 13.27s + 2.318)/(s^2 + 6.21s + 10.87)$ for the process (27) considering an uncertainty bound of $\pm 20\%$. Dashed lines represent the Bode magnitude of the function $(P_i(j\omega) - P_n(j\omega))/P_n(j\omega)$ for each one of the considered uncertain plants $P_i \in \mathcal{F}$.

TABLE I

PERFORMANCE OF THE CONTROLLERS IN EXAMPLES OF FIG. 9-10. *OV*: OVERSHOOT. IAE_{SP}: INTEGRATED ABSOLUTE ERROR (COMPUTED ONLY FOR THE SET-POINT CHANGE). IAE_{LD}: INTEGRATED ABSOLUTE ERROR (COMPUTED ONLY AFTER THE APPLICATION OF THE LOAD DISTURBANCE). S: NUMBER OF EVENTS.

	T_0	T_r	Δ	$OV\%$	IAE_{SP}	IAE_{LD}	S
Nominal	1.25	2	0.1	0%	5.77	5.68	32
Set 1 ●▲	1.25	2	0.1	37%	8.59	8.42	50
Set 2 ●	1.5	2	0.1	36%	8.11	8.62	39
Set 3 ●	2	2	0.1	34%	8.52	8.57	30
Set 2 ▲	1.25	3	0.1	25%	8.28	9.23	52
Set 3 ▲	1.25	4	0.1	15%	7.98	8.78	57

denoted with triangles illustrate the influence of the parameter T_r. The Nyquist plots and the response of the controllers are shown in Fig. 10. Left-side plots of Fig. 10 describe the results to variations of T_0. In qualitative terms, when T_r is fixed and T_0 is progressively increased, the disturbance rejection response slows down, which reduces the impact of the unmodeled dynamics (this results in an increment of the robustness of the closed-loop control system as remarked in [26]) while the set-point following performance is practically preserved. Specifically, the response of the feedback signal is more sluggish, which reduces the oscillations of the error signal such that it remains inside the deadband (see Section II).

Right-side plots show the results with respect to changes in T_r. As mentioned, results reveal that lower values of T_r increase the speed of the set-point response while the load disturbance is not significantly affected. The importance of making a balanced design has to be noted since excessive low values of T_r turns into a higher overshooting which in general increases the number of events without improving substantially the performance. Similarly, very low values of T_0 reduces the robustness of the design, reducing the relative stability of the equilibrium point under load disturbances.

For comparison with the nominal case (without structured uncertainty), Table I reflects some of the commonest performance indexes computed from simulations. As observed, the number of events generated is, in general, higher when parameters are close to the limit-cycle region, that is, when the relative stability of the system is lower. As expected, the resulting overshoot after set-point changes mainly depends on the value of T_r (being higher for lower values of T_r), and each one of the IAE indexes is particularly affected by the filter settings in a different way, where the IAE_{SP} may be regulated by the parameter T_r and the IAE_{LD} by T_0.

A.3. System behavior in the stability regions: The following experiment discusses the importance of satisfying the bounds for stability defined by the limits of the stability regions. In Fig. 11, the response of the control system with three sets of parameters is simulated, each one belonging to one of the characteristic stability areas for process (27). It can be observed that the relative stability is degraded as T_0 is decreased, which, in the case of the set 3, causes instability. It is worth stressing that, despite the set 2 belongs to the area of limit cycle solutions, an equilibrium point without limit cycles is achieved, similarly to design 1. This aspect is clarified in Remark 6. The bad performance of the design 3 is also verified in the

Nyquist plot. As observed, despite the reaction to load disturbances is slower than the case of the stable design (design 1), the equilibrium point is achieved faster than the design 2. Additionally, the oscillations and peaks (because of the small value of T_0) make design 2 close to destabilize the EBCS.

Remark 6: Regarding the stability region graphics, it should be noted that the solutions enclosed in areas denoted as "Stable limit cycle area" or "Instability area" represent solutions where at least one combination of the modeling errors that produce limit cycle or instability, respectively, exists inside the range of the considered uncertainties. In practice, this means that such limit of the areas could be violated for particular cases of modeling errors (which consequently supposes relaxing the robustness condition presented in Section 3) without achieving limit cycles or instability, respectively.

A.4. Influence of the event-triggering threshold: In last example, the influence of the event-based threshold Δ is analyzed. It can be easily understood, by analyzing the properties of the describing function, that the amplitude of the limit cycle, in terms of the numbers of levels Δ involved, is only determined by the points where the Nyquist plot crosses the DF [27]. This means that, when the design allows establishing a limit cycle, the value of Δ not only has influence on the number of events, but it determines also the precision of the system (in terms of amplitude of the oscillations). This aspect is highlighted in Fig. 12, where a filter that produces a limit cycle is tuned using two different values of Δ, that is, 0.05 and 0.1, respectively. It can be seen that the limit cycle has a different amplitude in each case. By contrast, in the area with absence of limit cycles, the event-threshold Δ has influence on the number of resulting events and on the bounds for the expected steady-state error value.

B. Illustrative Example 2

In the second simulation case, the event-based control of the IPTD process

$$P_u(s) = \frac{1}{s}e^{-4s} \tag{28}$$

is investigated, by considering a structured modeling error of ± 10 % in the set of parameters (K, L), that is, $\underline{K} = 0.9$, $\overline{K} = 1.1$,$\underline{L} = 3.6$ and $\overline{L} = 4.4$.

The corresponding stability regions are shown in Fig. 13. This example is also used to illustrate the compensation algorithm for step disturbances proposed in Section V. Here, two cases of structured uncertainty are simulated for the values $\left\{\overline{K}, \overline{L}\right\}$ and $\left\{\underline{K}, \underline{L}\right\}$, with $P_n(s)$ given by (28). The simulation results are shown in right-side of Fig. 13. In this case, the even-based threshold is set as 0.1. A unity set-point value is applied at the time instant $t = 0$ and a step input disturbance of 0.51 is introduced at the time instant $t = 50$. Note that with this value of Δ it is not possible to quantize the disturbance input and, therefore, to compensate it perfectly. Under this scenario, the compensation algorithm is introduced at the instant $t = 120$ (after the occurrence of some limit cycles). It is possible to note that, at this time instant, the control variable takes the value necessary to compensate the disturbance achieving a stable equilibrium point for both cases of the structured uncertainty.

Remark 7: Under ideal conditions, the exact compensation can be achieved so that the equilibrium point is preserved for future instants and new events are not triggered. In a real setup, because the noisy conditions and the unmodelled dynamics, the exact compensation is virtually impossible, but in practice a value very close to the ideal one can be determined so that it is possible to enlarge the period of the limit cycle reducing substantially the number of events.

C. Illustrative Example 3

In this example, the performance of the proposed event-based strategy is evaluated for the self-regulating high-order system

$$P_s(s) = \frac{(-0.3s+1)(0.08s+1)}{(2s+1)(s+1)(0.4s+1)(0.2s+1)(0.05s+1)^3} \tag{29}$$

In this case, the event generator is defined by the FOPDT approximation

$$P_n(s) = \frac{1}{2.5s+1}e^{-1.47s} \tag{30}$$

which has been obtained by using the Skogestad's half rule [34].

The case of unstructured uncertainties can be managed by increasing the level of robustness of the stability regions, which means considering in the computation a normalized dead time reasonably higher than the one obtained with the approximation, as well as a suitable bound for the uncertainties. For the process (29), in order to compute the stability regions, the dead time value is enlarged to consider a value of 2 (in comparison with the nominal one equal to 1.47) and a bound for the parameters of $\pm 20\%$ (see Fig. 14). With this setting, the validity of the regions is checked by simulating three designs (one for each characteristic behavior of the equilibrium point). It appears that each set reflects the expected behavior from each region. The correctness of the uncertainty bound is also verified with the Nyquist plots. Note that oversizing the stability regions contributes positively to delimit the real stable solutions, but obviously results in more conservative responses.

D. Illustrative Example 4

Last example is related to the control of a non self-process subject to unstructured uncertainties. The following transfer function is considered

$$P_u(s) = \frac{(-0.17s+1)^2}{s(s+1)(0.02s+1)} \tag{31}$$

and it is approximated as

$$P_n(s) = \frac{1}{s}e^{-1.69s} \tag{32}$$

In this case the stability regions are computed by considering a dead time of 2 seconds and a uncertainty bound of $\pm 10\%$ (see Fig. 15). The set-point and load disturbance step responses are shown in Fig. 15. The simulation conditions were configured like in example 2 and the compensation was introduced again at the time instant t = 120. The results demonstrate the stability of the solution and the effectiveness of the algorithm. It should be noted that the compensation algorithm works well both for structured and unstructured uncertainty cases because the method is independent on the process structure.

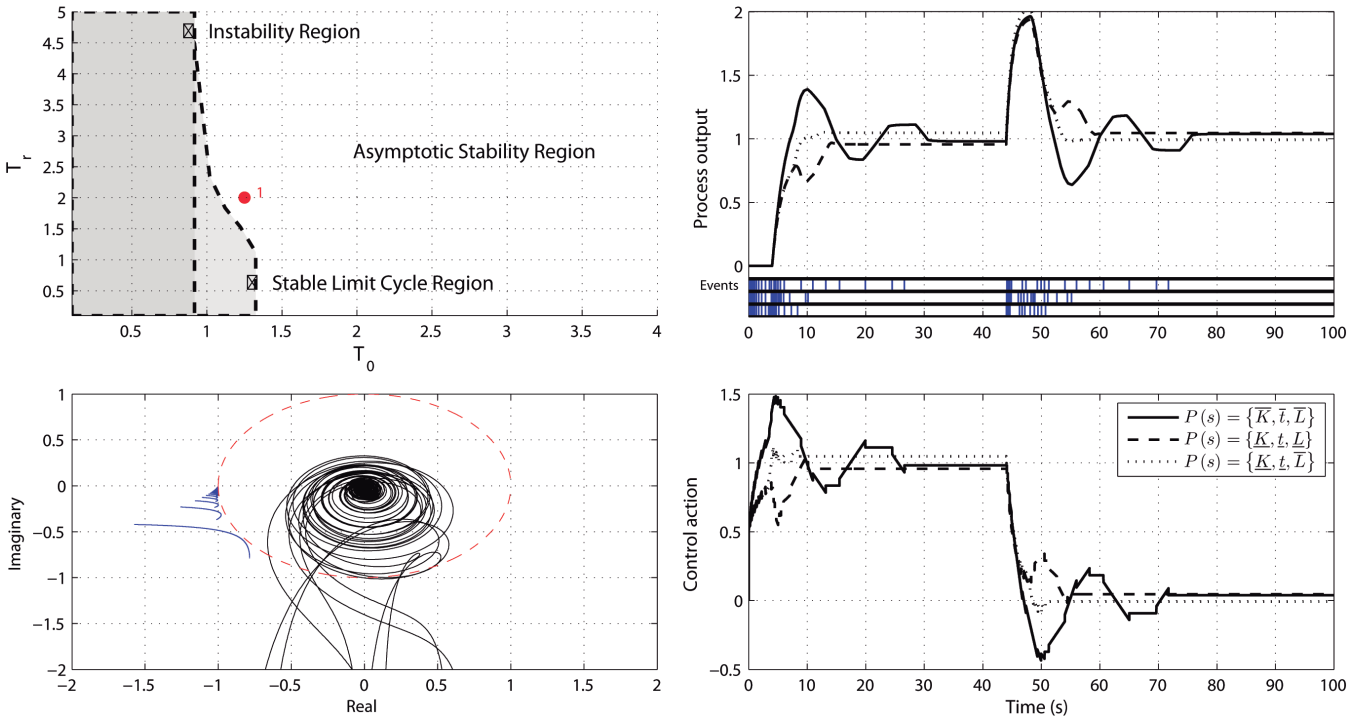

Fig. 8. Example of design with the proposed methodology for the process (27). Top-left plot: stability regions considering an uncertainty bound of $\pm 20\%$ for the process parameters. Bottom-left plot: Nyquist plots of the family of uncertain plants derived from the design point 1. Top-right plot: process outputs and number of events considering the design point 1 subjected to three cases of structured uncertainty given by $\{\overline{K}, \overline{\tau}, \overline{L}\}$, $\{\underline{K}, \underline{\tau}, \underline{L}\}$ and $\{\underline{K}, \underline{\tau}, \overline{L}\}$, respectively. Bottom-right plot: control variables for the three cases.

VII. Experimental Results

The proposed approach has also been implemented in a real application. For this purpose, a laboratory-scale setup of the Computer Science Department of the University of Cordoba has been used. In particular, the experimental setup consists in the speed control of a brushless DC motor as described in [24]. Since the system has an apparent dead time that is very small in comparison with the dominant time constant, a time delay of 4 s has been added via software at the process output. The estimated FOPDT model, obtained by applying the area method to the open-loop response of an input step from 1 to 1.5 V is

$$P(s) = \frac{0.53}{0.54s+1}e^{-4s} \tag{33}$$

The percentage of the output variation that is explained by the model is approximately 91%. To provide significant experiments, the process delay is assumed to be estimated with modeling errors. Particularly, the DTC is computed with

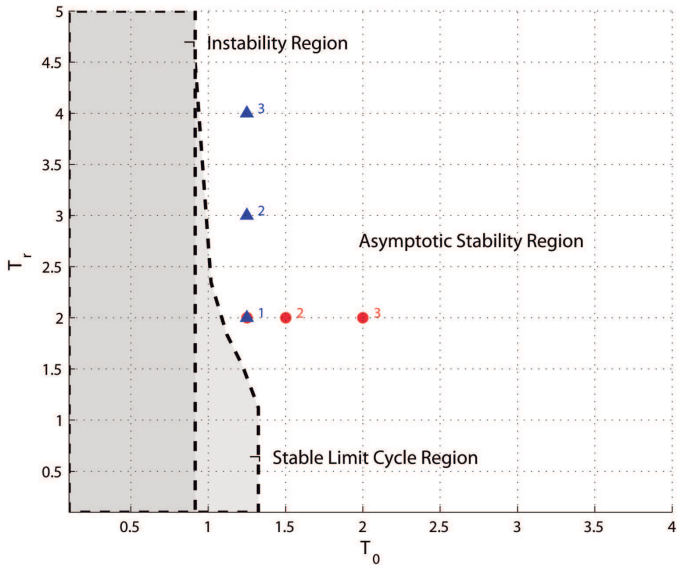

Fig. 9. Stability regions for the process (27) considering an uncertainty bound of ±20%. Six sets of stable controllers are defined to evaluate the tuning properties under changes in T_0 and T_r (denoted with circles and triangles, respectively).

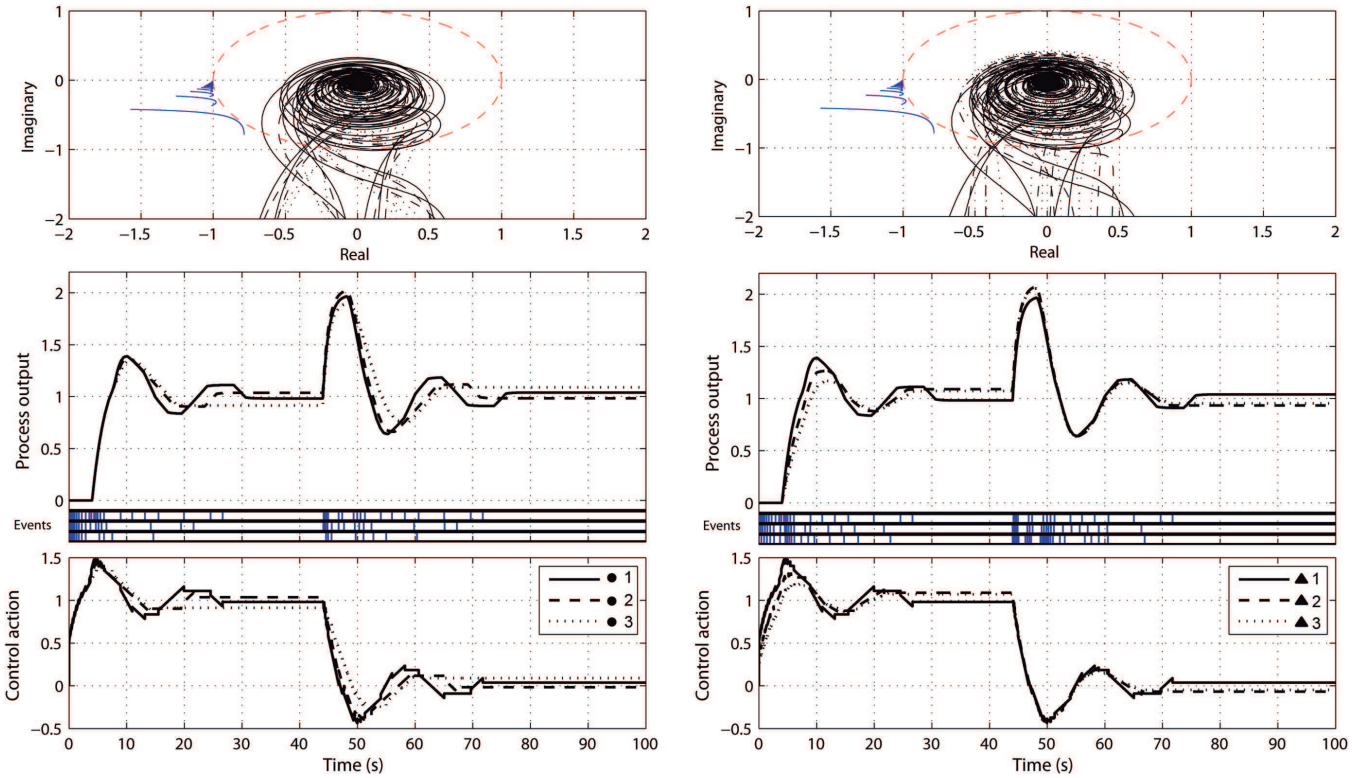

Fig. 10. Nyquist plots and control system responses for the six sets of stable controllers defined in Fig. 9 (denoted as sets 1 to 3 with circles on left-side plots, and with triangles on right-side plots, respectively). Top plots: Nyquist plots of the family of uncertain plants derived from the six design points for the process (27). Middle plots: process outputs and number of events for the six design point considered. Bottom plots: control variables.

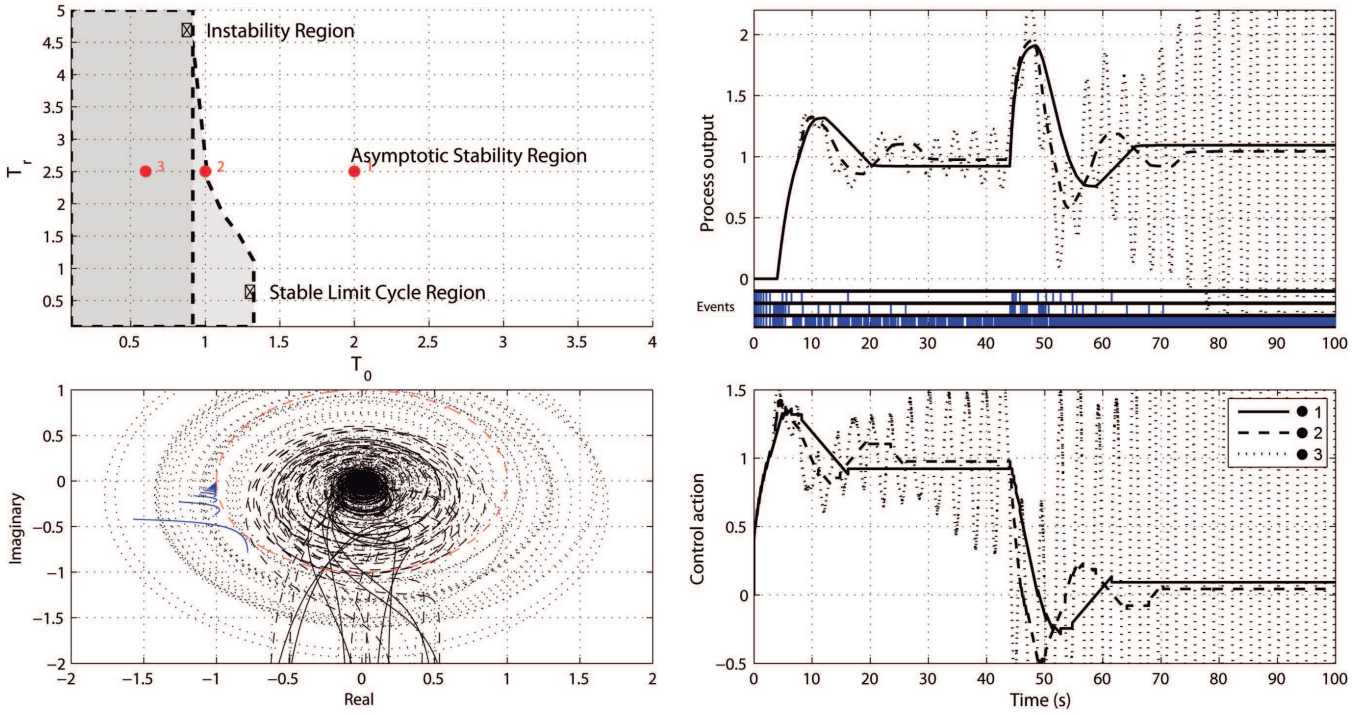

Fig. 11. Illustration of the response of some characteristic equilibrium points for the process (27). Top-left plot: stability regions considering an uncertainty bound of ±20% for the process parameters. Bottom-left plot: Nyquist plots of the family of uncertain plants derived from the three design points. Top-right plot: process outputs and number of events for the three design points with structured uncertainty given by $\{\underline{K}, \underline{\tau}, \underline{L}\}$. Bottom-right plot: control variables.

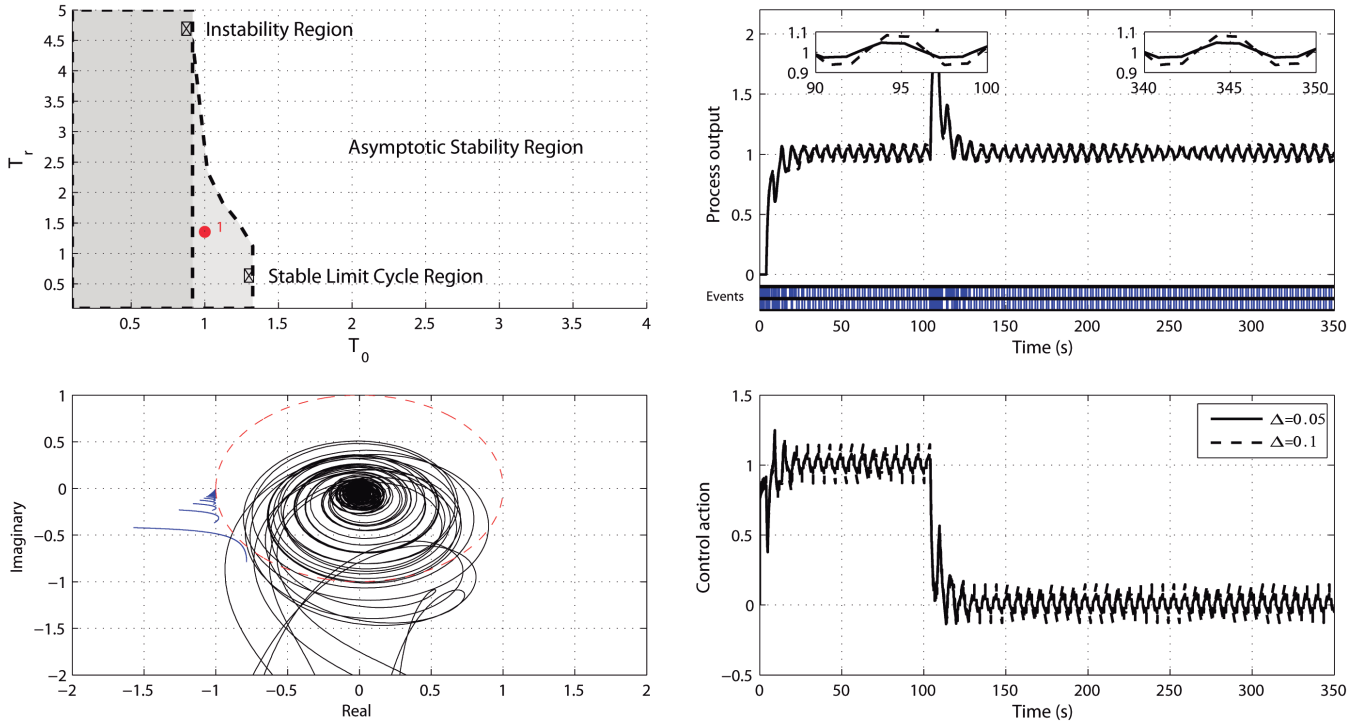

Fig. 12. Evaluation of the influence of the event-based threshold Δ inside the limit cycle region. Top-left plot: stability regions considering an uncertainty bound of ±20% for the process parameters. Bottom-left plot: Nyquist plots of the family of uncertain plants derived from the the design point 1. Top-right plot: process outputs and number of events for the design point 1 with structured uncertainty given by $\{\underline{K}, \underline{\tau}, \underline{L}\}$ and using $\Delta = 0.05$ and $\Delta = 0.1$. Bottom-right plot: control variables.

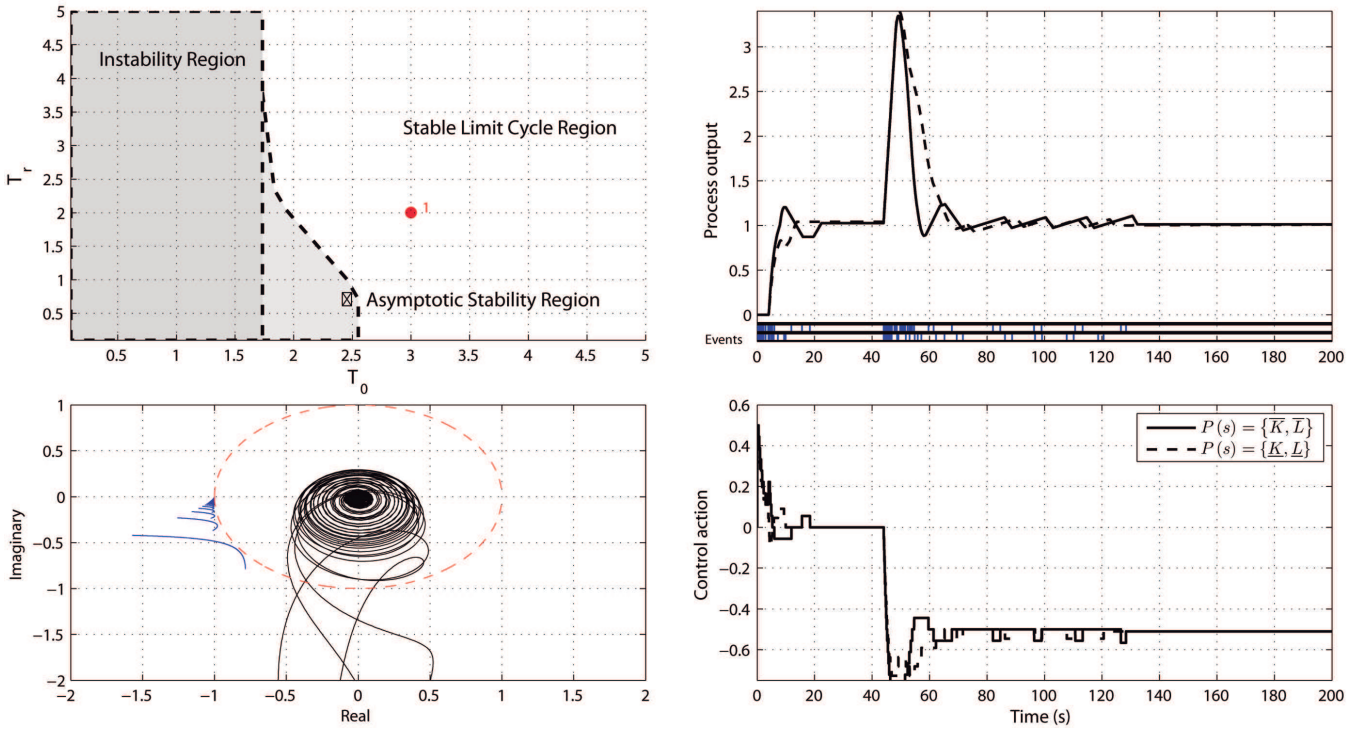

Fig. 13. Evaluation of the input disturbance compensation algorithm for the IPTD process (28). Top-left plot: stability regions considering an uncertainty bound of $\pm 10\%$ for the process parameters (K, L). Bottom-left plot: Nyquist plots of the family of uncertain plants derived from the design point 1. Top-right plot: process outputs and number of events for the design point 1 with structured uncertainty given by $\{\overline{K}, \overline{L}\}$ and $\{\underline{K}, \underline{L}\}$. Bottom-right plot: control responses for the three cases of structured uncertainty.

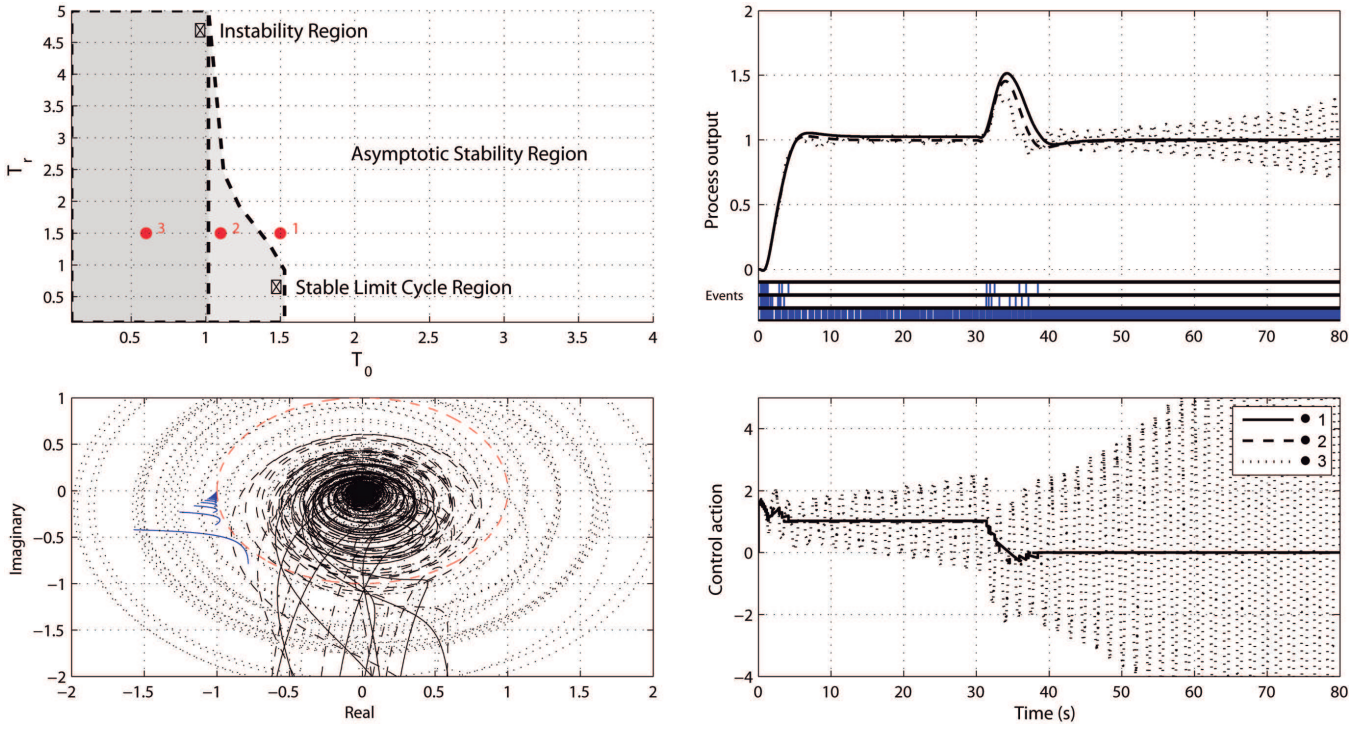

Fig. 14. Control of the self-regulating process (29) subjected to unstructured uncertainty. Top-left plot: stability regions considering an uncertainty bound of a $\pm 20\%$ for the modeled process parameters K_m, τ_m, L_m. Bottom-left plot: Nyquist plots of the family of uncertain plants derived from the three design points. Top-right plot: process outputs and number of events for the three design points. Bottom-right plot: control variables.

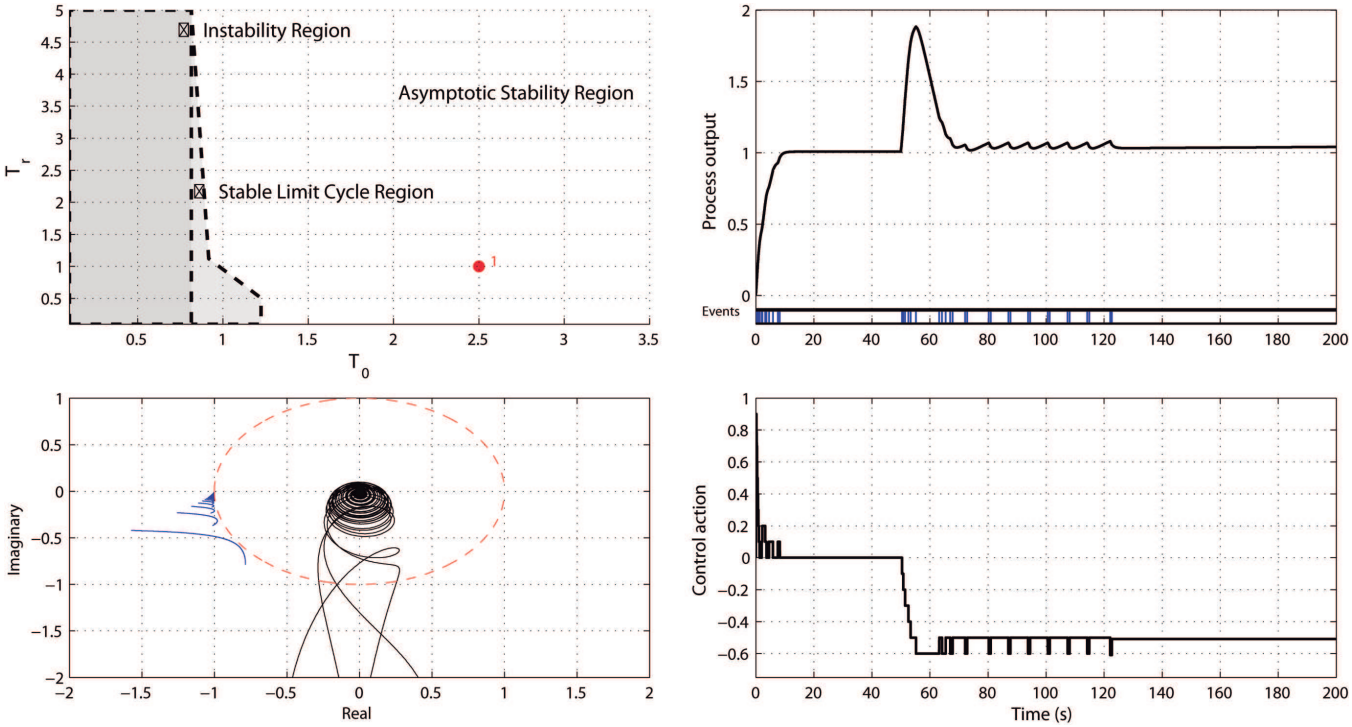

Fig. 15. Control of the integrating process (31) subjected to unstructured uncertainty. Top-left plot: stability regions considering an uncertainty bound of ±20% for the modeled process parameters K_m, L_m. Bottom-left plot: Nyquist plots of the family of uncertain plants derived from the three design points for process (31). Top-right plot: process outputs and number of events. Bottom-right plot: control variable.

the model $P_n(s) = (0.53/(0.54s+1))e^{-3.2s}$, which introduces a plant/model mismatch of -20% in the dead time. With this information, the stability regions are computed (see Fig. 16).

Four cases are developed (denoted with a circle and an index, respectively). The first three designs were performed to obtain asymptotic stable responses, whereas the last one considers the stable limit cycle operation region. The responses are shown in Fig. 17. Table II shows the results of the performance indexes considered in this work. Such experiments aim at demonstrating experimentally the properties of the approach and of the tuning methodology.

A set-point change of magnitude 0.5 V is demanded at the initial time instant and a step input disturbance of 0.5 V is applied at the time instant 15 s. The event threshold was selected as $\Delta = 0.1$, which is approximately 3-4 times higher than the noise band coupled to the output signal. As expected, the experiments derived from the designs 1 to 3 (belonging to the stable region) provide asymptotic stable responses (without limit cycles), whereas the design 4 achieves a limit cycle after the set-point change. If we compare the responses, it can be observed that the general properties of the tuning parameters are preserved. The designs 1,2 and 4 produce very similar fast transitions after the set-point change, which occurs because of a similar tuning for T_r. In the case of the design 4, it establishes a three-state limit cycle as a consequence of the low robustness caused by the small value of T_0. Conversely, the design 1 is able to reject the input disturbance with a smaller settling time in comparison with the design 2 (which has a higher level of robustness) but it is more sensitive to plant-model mismatches as proven from the oscillation produced after the set-point change. This is also reflected in a higher number of transitions of the event-based error signal and therefore, a higher number of events, as shown in Table II. In contrast, the designs 1 and 2 produce faster set-point transitions since the value of T_r is lower. It turns out that the stability properties as well as the robustness properties can be characterized with the developed tuning method. Such results are also summarized in the values of the performance indexes shown in Table II.

VIII. Conclusions

In this paper, a unified structure for an event-based control system has been analyzed. The methodology is based on the integration of the filtered Smith predictor in the event generator.

The robust stability has been addressed from the perspective of the limit cycles introduced by the nonlinear element. A describing function approach has been proposed to this aim, where two of the most common process structures of industrial relevance has been considered for this analysis, namely, FOPTD and IPTD systems.

A simple method to obtain the characteristic stability map of the EBCS has been provided, where only some prior knowledge of the uncertainties bound is necessary. Two sufficient conditions have been derived in Sections III and IV. It

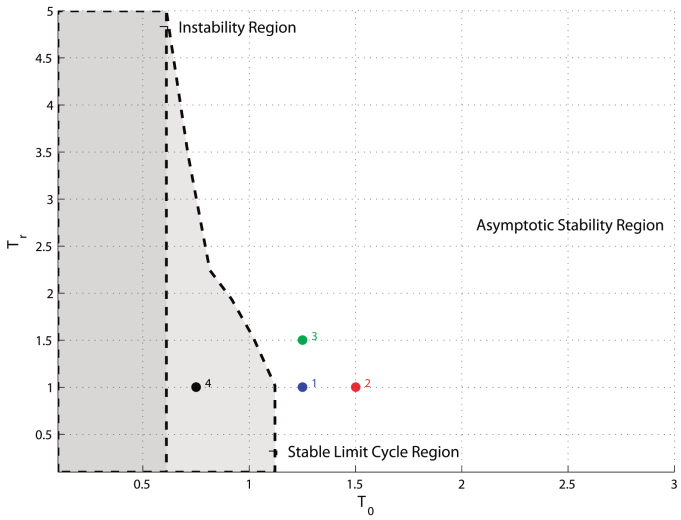

Fig. 16. Stability regions and design points for the experimental process plant.

TABLE II

PERFORMANCE OF THE CONTROLLERS IN THE EXPERIMENTAL SETUP. OV: OVERSHOOT. IAE_{SP}: INTEGRATED ABSOLUTE ERROR (COMPUTED ONLY FOR THE SET-POINT CHANGE). IAE_{LD}: INTEGRATED ABSOLUTE ERROR (COMPUTED ONLY AFTER THE APPLICATION OF THE LOAD DISTURBANCE). S: NUMBER OF EVENTS.

	T_0	T_r	Δ	$OV\%$	IAE_{SP}	IAE_{LD}	S
Set 1 ●	1.25	1	0.1	7.9%	1.1	0.42	19
Set 2 ●	1.5	1	0.1	7.6%	1.31	0.4	18
Set 3 ●	1.25	1.5	0.1	6.8%	1.5	0.74	13
Set 4 ●	0.75	1	0.1	10.6%	1.24	0.61	38

has been shown that both the stability and limit cycle properties do not depend on the event-triggering strategy, so that it can be tuned in order to handle the trade-off between the number of events and the system precision. The advantage of this method relies on the simplicity of the framework to handle the system properties in term of robustness and load disturbance rejection performance.

Simulations and experimental results confirmed the effectiveness of the approach in the framework of event-based control architectures and robust tuning methodologies.

In future works, the extension of the proposed approach to unstable processes will be investigated.

ACKNOWLEDGEMENT

This work has been supported in part by Project DPI2012-31303 (financed by the Spanish Ministry of Economy and Competitiveness).

REFERENCES

[1] A. O'Dwyer, *Handbook of PI and PID Controller Tuning Rules*. Imperial College Press, 2009.
[2] K. Åström and T. Hägglund, *Advanced PID Control*. ISA Press, Research Triangle Park (USA), 2005.
[3] R. Gupta and M.-Y. Chow, "Networked control system: Overview and research trends," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 7, pp. 2527–2535, 2010.
[4] T. Blevins, "PID advances in industrial control," in *Proceedings IFAC Conference on Advances in PID Control*, Brescia (I), 2012, pp. 23–28.
[5] A. Willig, "Recent and emerging topics in wireless industrial communications: A selection," *IEEE Transactions on Industrial Informatics*, vol. 4, no. 2, pp. 102–122, 2008.
[6] S. Dormido, J. Sánchez, and E. Kofman, "Muestreo, control y comunicacin basados en eventos," *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 5, no. 1, pp. 5 – 26, 2008.
[7] A. Ruiz, J. Jiménez, J. Sánchez, and S. Dormido, "Control basado en eventos de sistemas de primer orden con retardo," *Revista Iberoamericana de Automática e Informática Industrial RIAI*, vol. 10, no. 3, pp. 302 – 312, 2013.
[8] J. Sánchez, A. Visioli, and S. Dormido, "A two-degree-of-freedom PI controller based on events," *Journal of Process Control*, vol. 21, no. 4, pp. 639–651, 2011.
[9] J. Lunze and D. Lehmann, "A state-feedback approach to event-based control," *Automatica*, vol. 46, no. 1, pp. 211 – 215, 2010.
[10] M. Beschi, S. Dormido, J. Sánchez, and A. Visioli, "Characterization of symmetric send-on-delta PI controllers," *Journal of Process Control*, vol. 22, no. 10, pp. 1930 – 1945, 2012.

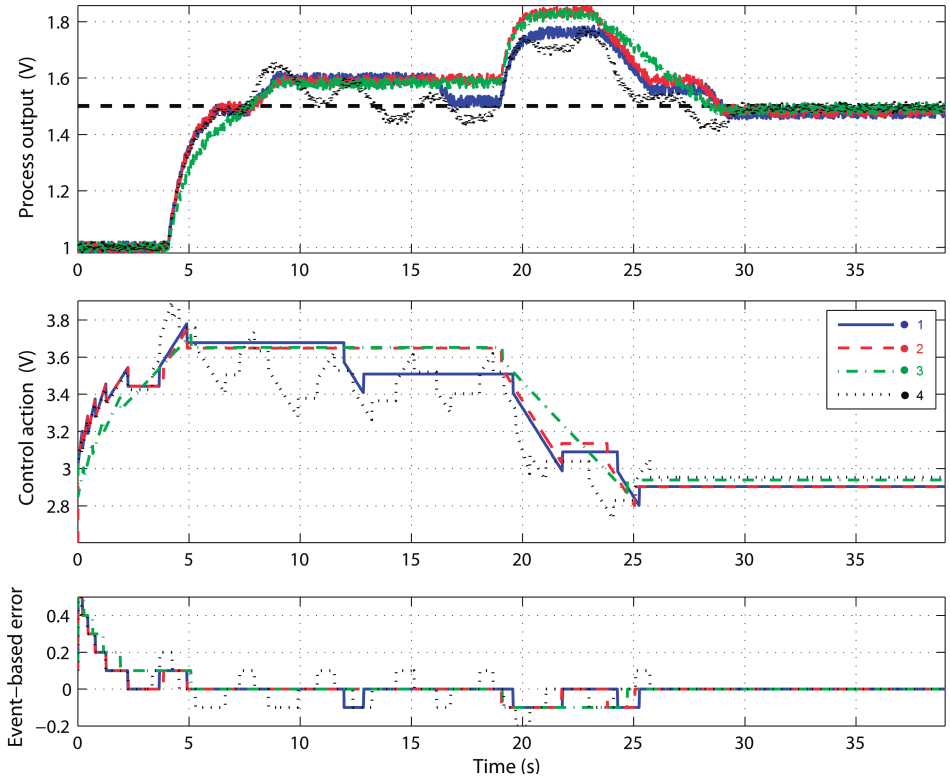

Fig. 17. Experimental results for the speed control of the DC motor. Top plots: process outputs for the four design points considered. Middle plots: control responses. Bottom plots: event-based error signals.

[11] A. Leva and A. Papadopoulos, "Tuning of event-based industrial controllers with simple stability guarantees," *Journal of Process Control*, vol. 23, no. 9, pp. 1251 – 1260, 2013.

[12] T. Blevins, M. Nixon, and W. Wojsznis, "Event-based control applied to wireless throttling valves," in *Proceedings First International Conference on Event-based Control, Communication, and Signal Processing*, 2015, pp. 1–6.

[13] J. Sánchez, A. Visioli, and S. Dormido, "Event-based pid control," in *PID Control in the Third Millennium*, R. Vilanova and A. Visioli, Eds. London (UK): Springer, 2012, pp. 495–526.

[14] J. Sánchez, M. Guarnes, and S. Dormido, "On the application of different event-based sampling strategies to the control of a simple industrial process," *Sensors*, vol. 9, no. 9, pp. 6795–6818, 2009.

[15] V. Vasyutynskyy and K. Kabitzsch, "Event-based control: Overview and generic model," in *Proceedings 8th IEEE International Workshop on Factory Communication Systems*, Nancy (F), May 2010, pp. 271–279.

[16] J. Chacón, J. Sánchez, A. Visioli, L. Yebra, and S. Dormido, "Characterization of limit cycles for self-regulating and integral processes with PI control and send-on-delta sampling," *Journal of Process Control*, vol. 23, no. 6, pp. 826 – 838, 2013.

[17] G. Anastasi, M. Conti, M. Di Francesco, and A. Passarella, "Energy conservation in wireless sensor networks: A survey," *Ad Hoc Networks*, vol. 7, no. 3, pp. 537 – 568, 2009.

[18] M. Miskowicz, "Send-on-delta concept: An event-based data reporting strategy," *Sensors*, vol. 6, no. 1, pp. 49–63, 2006.

[19] E. Kofman and J. Braslavsky, "Level crossing sampling in feedback stabilization under data-rate constraints," in *Proceedings 45th IEEE Conference on Decision and Control*, San Diego (USA), 2006, pp. 4423–4428.

[20] V. Vasyutynskyy and K. Kabitzsch, "Towards comparison of deadband sampling types," in *Proceedings IEEE International Symposium on Industrial Electronics*, Vigo (E), 2007, pp. 2899–2904.

[21] K. J. Åström, "Event based control," in *Analysis and Design of Nonlinear Control Systems: In Honor of Alberto Isidori*, A. Astolfi and L. Marconi, Eds. London (UK): Springer, 2008, pp. 495–526.

[22] A. Ruiz, J. E. Jimenez, J. Sanchez, and S. Dormido, "Characterization and tuning of predictive SSOD-PI controllers," in *Proceedings 23th Mediterranean Conference on Control and Automation*, Torremolinos (E), 2015, pp. 551–557.

[23] M. Beschi, S. Dormido, J. Sanchez, A. Visioli, and L. Yebra, "Event-based PI plus feedforward control strategies for a distributed solar collector field," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1615–1622, 2014.

[24] Á. Ruiz, J. Jiménez, J. Sánchez, and S. Dormido, "A practical tuning methodology for event-based PI control," *Journal of Process Control*, vol. 24, no. 1, pp. 278 – 295, 2014.

[25] M. Beschi, S. Dormido, J. Sánchez, and A. Visioli, "Closed-loop automatic tuning technique for an event-based pi controller," *Industrial & Engineering Chemistry Research*, vol. 54, no. 24, pp. 6362–6370, 2015.

[26] J. Normey-Rico and E. Camacho, "Unified approach for robust dead-time compensator design," *Journal of Process Control*, vol. 19, no. 1, pp. 38–47, 2009.

[27] A. Gelb and W. Vander Velde, *Multiple-input describing functions and nonlinear system design*. McGraw-Hill, New York (USA), 1968.

[28] M. Beschi, S. Dormido, J. Sanchez, and A. Visioli, "Tuning of symmetric send-on-delta Proportional-Integral controllers," *IET Control Theory Applications*, vol. 8, no. 4, pp. 248–259, 2014.

[29] B. Armstrong-Hélouvry, P. Dupont, and C. C. De Wit, "A survey of models, analysis tools and compensation methods for the control of machines with friction," *Automatica*, vol. 30, no. 7, pp. 1083–1138, 1994.

[30] B. Armstrong and B. Amin, "PID control in the presence of static friction: A comparison of algebraic and describing function analysis," *Automatica*, vol. 32, no. 5, pp. 679–692, 1996.
[31] J.-J. E. Slotine and W. Li, *Applied nonlinear control*. Prentice-Hall, Englewood Cliffs (USA), 1991.
[32] J. A. R. Perez and R. S. Llopis, "A new method for tuning PI controllers with symmetric send-on-delta sampling strategy," *ISA Transactions*, vol. in press.
[33] K. Zhou and J. C. Doyle, *Essentials of robust control*. Prentice Hall, Upper Saddle River (USA), 1998.
[34] S. Skogestad, "Simple analytic rules for model reduction and PID controller tuning," *Journal of Process Control*, vol. 13, no. 4, pp. 291 – 309, 2003.