Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

Event-based predictive control triggered by input and output deadband conditions

A. Pawlowski * **J. L. Guzmán** ** **M. Berenguel** ** **S. Dormido** *
I. Fernández **

* *Dep. Informática y Automática, ETSI Informática, UNED, 28040 Madrid, Spain (e-mail:a.pawlowski@dia.uned.es, sdormido@dia.uned.es)*
** *Dep. Informática, University of Almería, ceiA3, 04120 Almería, Spain, (joseluis.guzman@ual.es, beren@ual.es, ifernandez@ual.es)*

Abstract: This paper describes a predictive event-based control algorithm focused on practical issues. The Generalized Predictive Control scheme is used as predictive controller, and sensor and actuator deadbands are included in the design procedure to provide event-based capabilities. The first configuration adapts a control structure to deal with asynchronous measurements; the second approach is used for asynchronous process updates. The presented ideas, for an event-based control scheme, conserve all of the well-known advantages of the adaptive sampling techniques applied to process sampling and updating. The objective of this combination is to reduce the control effort while the control system precision is maintained at an acceptable level. The presented algorithm allows to obtain a direct tradeoff between performance and number of actuator events. The diurnal greenhouse temperature control problem is used as test bench to evaluate the different control algorithms configurations.

Keywords: Generalized Predictive Control, Events, Deadband, Actuators, Sensors.

1. INTRODUCTION

The principal features of event-based control are specially important from a practical point of view. First of them is the minimization of the actuation load, which extends actuator lifespan for electromechanical or mechanical systems. For the event-triggered controller, the control signal will be applied to the controller process in an asynchronous mode, only when it is necessary. On the contrary, classical control systems, apply new control actions every sampling instant even for insignificant errors. In such a way, constant position changes of the actuator are required. Moreover, the event-based control is beneficial for distributed control systems (commonly used in industry), where important reduction in the information exchange between control system agents can be achieved (Årzén, 1999; Pawlowski et al., 2012). This feature is even more advantageous if the control system is implemented through computer networks, that share bandwidth with other tasks (Pawlowski et al., 2012).

Another important benefit which characterizes the event-based control systems is the relaxed requirements on the sampling period. This problem is very important in all computer-based control systems (Anta and Tabuada, 2010). The correct selection of the best sampling period for a digital control system is a compromise between many design aspects. Lower sampling rates are usually characterized by their elevated cost; a slow sampling time directly reduces the hardware requirements and the overall cost of process automation. A slower sample rate makes it possible for a slower computer to achieve a given control function, or provides greater capability from a given computer. Factors which may require an increase in the sample rate are: command signal tracking effectiveness, sensitivity to plant disturbance rejection, and sensitivity to plant parameter variations (Sánchez et al., 2011).

As mentioned above, in event-based control systems the signal samples are triggered in an asynchronous way. Usually, special requirements, or limitations on sampling-rate issues, appear in the distributed control system. The reason is that typical implementation of modern control systems involves communication networks connecting different control system agents (Can et al., 2006; Hu et al., 2007; Varutti et al., 2009; Chen et al., 2008; Varutti et al., 2010). However, most of presented event-based and networked control systems are focused on the events and communications between the sensor and controller nodes. As a consequence of this assumption, the communication rates between the controller and actuator nodes are determined by the controller input events. However, the issue related to actuators has received less attention. Recently, some problems related to controller-actuator links were considered in certain research works (Beschi et al., 2012; Pawlowski et al., 2014). Nowadays, there are many active research topics related to event-based control, where Model Predictive Control (MPC) techniques are frequently used (Gawthrop and Wang, 2007; Hu et al., 2007; Ploennigs et al., 2010). Most attention has been given to networked control systems where event-based MPC controllers have been used to reduce the amount of information exchanged through a network of control system nodes (Quevedo et al., 2011; Jugo and Eguiraun, 2010; Bemporad et al., 2009). Moreover, the MPC properties are used to compensate for typical phenomena related

to computer networks, such as variable delay, packet loss, etc.

The present work introduces a complete event-based GPC control scheme that considers both sensor and actuator deadband at the same time. To realize this event-based control structure, it is necessary to build the control structure for the sensor deadband approach (Pawlowski et al., 2012). Additionally, the developed controllers consider the actuator deadband in the optimization procedure (Pawlowski et al., 2014). In presented control scheme, an event-based GPC controller manages two sources of events, related to process output and input, respectively. Thus, the complete event-based GPC control structure has two additional tuning parameters, β_y and β_u, which determine the deadband for the sensor and the actuator, respectively. Each of these tuning parameters can be used to obtain the desired compromise between controller performance and the number of events for the sensor and the actuator, independently. The general ideas of the analyzed event-based algorithm and simulation results on the greenhouse diurnal temperature control problem are described in the subsequent sections.

2. EVENT-BASED GPC CONTROL STRUCTURE

The event-based control structure used in this work is shown in Figure 1, where C represents an event-based controller and $P(s)$ the controlled process. In this configuration, two types of events can be generated from *u-based* and *y-based* conditions. In the developed application, the actuator possesses a ZOH (Zero-Order Hold), thus the current control signal is kept until a new one is computerd.

The *u-based* criterion is used to trigger the input-side event, E_u, consisting of the transmission of a new control action, $u(t_k)$, when differ enough (bigger than a threshold β_u) with respect to the previous control action. On the other hand, the *y-based* condition will trigger the output side event, E_y, when the difference between the reference $w(t)$ and the process output is out of the limit β_y. The adaptive sampling technique with deadband sampling/updating is used for *y-based* and *u-based* conditions (Ploennigs et al., 2010).

Since the *u-based* condition is related to the actuator and the *y-based* condition with the sensor, three approaches to perform event-based control strategies can be established:

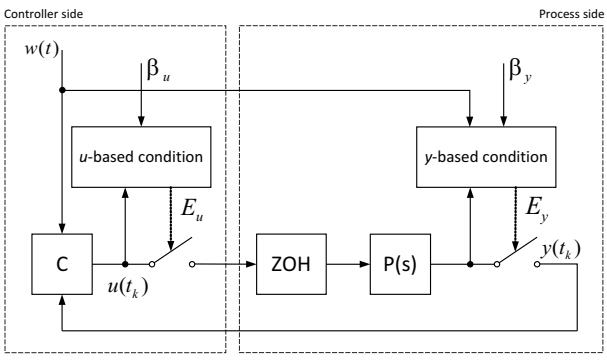

Fig. 1. Event-based control approach. t_k refers to time of events and t to discrete time (samples).

sensor deadband approach, actuator deadband approach, and complete event-based approach.

2.1 Sensor Deadband Approach for Event-Based Control

Generally, an event-based controller is composed of: an event detector and a controller (Åström, 2007). The first component indicates to the controller when a new control signal must be calculated and is triggered by the occurrence of a new input event. In presented scheme, the controller node is formed by a set of GPC controllers. In such a way, corresponding controller is selected according to a new event (time instant), that determine current sampling interval. The event-based with sensor deadband operates using the following ideas (Pawlowski et al., 2012):

- The controlled variable is sampled with T_{base} sampling time at the event generator block. On the contrary, the control action is calculated and send to the process with T_f (variable sampling time) which is determined by an control system input events.
- The T_f is set up as a multiple of T_{base} ($T_f = fT_{base}$, $f \in [1, n_{max}]$) and satisfies $T_f \leq T_{max}$, where $T_{max} = n_{max}T_{base}$ the maximum sampling time value to keep a minimum performance.
- The T_{base} and T_{max} sampling times are determined using process data and closed-loop specifications, according to classical methods for sampling time selection.
- When a new control signal is applied (at time t), the process variable is continuously supervised by the event generator block with T_{base} and verified if the process output satisfies established conditions. If one of the condition is meet, a new event is generated with a variable sampling period T_f. In consequence, a new control action is computed. Otherwise, in absence of events from controlled variable, system is driven by a time events (at $t_k = t + T_{max}$).
- Taking into account presented working principle, the control signal will be calculated with variable sampling time, T_f. Due to this feature, a set of GPC controllers is used. In this case each GPC controller is designed for a specific sampling time $T_f = fT_{base}, f \in [1, n_{max}]$. Moreover, it is necessary application of resampling technique to avoid undesirable jumps in the control action at each change between controllers.

GPC algorithm - In accordance with introduced scheme, the proposed control structure use the GPC algorithm as the feedback controller. Thus, a set of GPC controllers is implemented, one for each sampling time T_f, $f \in [1, n_{max}]$. Everyone of them is implemented using classical GPC formulation. The objective of the GPC controller is to obtains a control sequence that minimizes a multistage cost function of the following form (Pawlowski et al., 2012):

$$J = \sum_{j=N_1^f}^{N_2^f} \delta^f[\hat{y}^f(t+j|t) - w(t+j)]^2 + \sum_{j=1}^{N_u^f} \lambda^f[\Delta u^f(t+j-1)]^2 \quad (1)$$

where $\hat{y}^f(t+j|t)$ is an optimum j time instants ahead prediction of the process output using data up to time t. The $\Delta u^f(t+j-1)$ is a future control sequence and $w(t+j)$ is the future reference. All those signals are considered for

a T_f ($t = kT_f, k \in Z^+$) sampling time. Additionally, all controllers are tuned using: the minimum and maximum prediction horizons, N_1^f and N_2^f, the control horizon, N_u^f, and the future error and control weighting factors, δ^f and λ^f, respectively (Pawlowski et al., 2012). The objective of GPC is accomplished by minimizing J, in order to compute the future control sequence $u^f(t), u^f(t+1), ..., u^f(t+N_u^f-1)$.

Signal sampling and resampling technique - Such as described above, the computation of a new control action is done with a variable sampling period T_f. So, in order to implement the GPC control algorithm, the past values of the process variables and of the control signals must be available sampled with that sampling period T_f. Therefore, a resampling of the corresponding signals is required, such as described in the following.

Resampling of process output - In accordance with introduced working principles, the controller node receives a new state of the plant when an event is triggered. Required past information is stored in the controller block and is saved with T_{base} sampling frequency. The resampling process is performed using a linear approximation between two consecutive events. Later, approximated data are sampled with the T_{base} sampling period, forming $y^b(k)$ with $k = 0, T_{base}, 2T_{base}, 3T_{base}, \ldots$. Once the controlled variable is resampled, the necessary past information is obtained in accordance with new sampling frequency T_f, resulting in a new discrete time signal, y^f, containing information of the controlled variable sampled with T_f.

In such a way, the y^f vector is obtained, containing the past information with current sampling period T_f. Afterwards, this vector is used in the calculation of the control action.

Reconstruction of past control signals - The algorithm is quite similar to that presented for the resampling of the controlled variable. The discrete time control signal, u^b, stores the past control signal sampled with T_{base}. On the contrary to the process output reasmpling, firstly past information is computed and later the u^b vector is actualized. Considering that a new event is triggered, and a new sampling time $T_f = fT_{base}$ is determined. The past information (for T_f), is obtained from the u^b vector and stored in a temporary variable u_p^f. Subsequently, the u_p^f and the y^f, are used to compute a new control signal, represented by $u^f(T_f) = u^b(k)$. When a new control signal has been obtained, $u^f(T_f) = u^b(k)$, the u^b vector is actualized, maintaining constant signal value between the two successive events.

2.2 Actuator Deadband Approach for Event-Based Control

The main idea of this approach is to develop a control structure where the control signal is updated in an asynchronous manner. The main goal is to reduce the number of control signal updates, saving system resources, retaining acceptable control performance. Therefore, this section will focus on the actuator deadband approach, which tries to face these drawbacks regarding control signal changes.

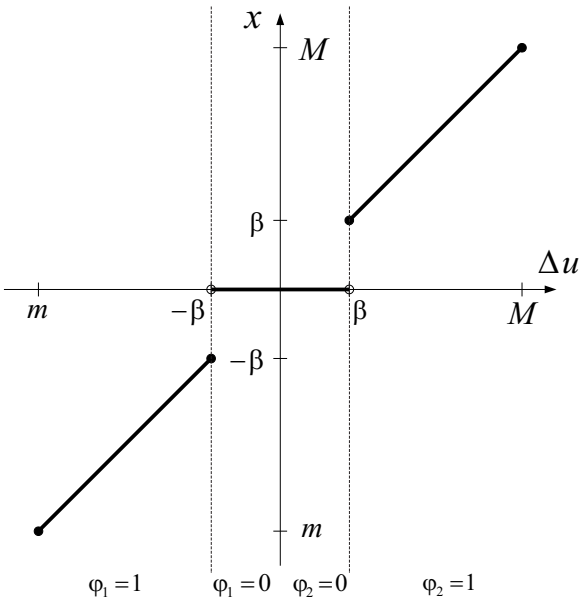

Fig. 2. Control signal increments with deadband.

The actuator deadband can be understood as a constraint on control signal increments $\Delta u(t)$:

$$|\Delta u(t)| = |u(t-1) - u(t)| \geqslant \beta_u \tag{2}$$

where β_u is the proposed deadband.

The introduced deadband, β_u, will be used as an additional tuning parameter for control system design, to adjust the number of actuator events (transmissions from controller to actuator). In this work, the presence of the actuator deadband was included into the controller design procedure in order to improve the event-based control system performance.

The methodology presented in this work consists of including the actuator virtual deadband into the GPC design framework (Pawlowski et al., 2014). The deadband nonlinearity can be handled together with other constraints on controlled process. The deadband can be expressed mathematically with a hybrid design framework developed by (Bemporad and Morari, 1999), that allows to translate discrete rules into a set of linear logical constraints. The resulting formulation consists in a system containing continuous and discrete components, which is known as a Mixed Logical Dynamic (MLD) system (Bemporad and Morari, 1999).

Let us introduce two logical variables, φ_1 and φ_2, to determine a condition on control signal increments, $\Delta u(t)$. So, these logical variables are used to describe the different stages of the control signal with respect to the deadband, as following:

$$x(t) = \begin{cases} \Delta u(t) : \Delta u(t) \geqslant \beta & \varphi_2 = 1 \\ 0 : \Delta u(t) \leqslant \beta & \varphi_2 = 0 \\ 0 : \Delta u(t) \geqslant -\beta & \varphi_1 = 0 \\ \Delta u(t) : \Delta u(t) \leqslant -\beta & \varphi_1 = 1 \end{cases} \tag{3}$$

To make this solution more general, minimal m and maximal M values for control signal increments are included into the control system design procedure, resulting in $M = max\{\Delta u(t)\}$ and $m = min\{\Delta u(t)\}$. In this way,

it is possible to determine the solution region based on binary variables. Figure 2 shows the virtual deadband in a graphical form, where each region can be distinguished. Thus, the proposed logic determined by equation (3) can be translated into a set of *mixed-integer linear inequalities* involving both continuous variables, $\Delta u \in \mathbb{R}$, and logical variables $\varphi_i \in \{0, 1\}$. Finally, a set of mixed-integer linear inequalities constraints for the actuator deadband are established as:

$$\begin{aligned} \Delta u - (M-\beta)\varphi_2 &\leqslant \beta \\ \Delta u + (M+\beta)\varphi_1 &\leqslant M \\ \Delta u - M\varphi_2 &\leqslant 0 \\ -\Delta u + (m+\beta)\varphi_1 &\leqslant \beta \\ -\Delta u - (m-\beta)\varphi_2 &\leqslant -m \\ -\Delta u + m\varphi_1 &\leqslant 0 \\ \varphi_1 + \varphi_2 &\leqslant 1 \end{aligned}$$

MIQP based design for control signal deadband - The reformulated hybrid input constraints presented above are integrated in the GPC optimization problem, where the resulting formulation belongs to an MIQP optimization problem.

In the case where the control horizon is $N_u > 1$, the corresponding matrix becomes:

$$\underbrace{\begin{bmatrix} 1\underline{\mathbf{D}} & 0\underline{\mathbf{D}} & -(M-\beta)\underline{\mathbf{D}} \\ 1\underline{\mathbf{D}} & (M+\beta)\underline{\mathbf{D}} & 0\underline{\mathbf{D}} \\ 1\underline{\mathbf{D}} & 0\underline{\mathbf{D}} & -M\underline{\mathbf{D}} \\ -1\underline{\mathbf{D}} & (m+\beta)\underline{\mathbf{D}} & 0\underline{\mathbf{D}} \\ -1\underline{\mathbf{D}} & 0\underline{\mathbf{D}} & -(m-\beta)\underline{\mathbf{D}} \\ -1\underline{\mathbf{D}} & m\underline{\mathbf{D}} & 0\underline{\mathbf{D}} \\ 0\underline{\mathbf{D}} & 1\underline{\mathbf{D}} & 1\underline{\mathbf{D}} \end{bmatrix}}_{\boldsymbol{C}} \underbrace{\begin{bmatrix} \Delta u\underline{\mathbf{d}} \\ \varphi_1\underline{\mathbf{d}} \\ \varphi_2\underline{\mathbf{d}} \end{bmatrix}}_{\boldsymbol{x}} \leqslant \underbrace{\begin{bmatrix} \beta\underline{\mathbf{d}} \\ M\underline{\mathbf{d}} \\ 0\underline{\mathbf{d}} \\ \beta\underline{\mathbf{d}} \\ -m\underline{\mathbf{d}} \\ 0\underline{\mathbf{d}} \\ 1\underline{\mathbf{d}} \end{bmatrix}}_{\boldsymbol{\rho}}$$

where $\underline{\mathbf{D}}$ is a matrix $(N_u \times N_u)$ of ones and $\underline{\mathbf{d}}$ is a vector of ones with size $(N_u \times 1)$.

The previous matrices that contain linear inequality constraints can be expressed in a general form as

$$\mathbf{C}\boldsymbol{x} \leqslant \boldsymbol{\rho} \tag{4}$$

with $\boldsymbol{x} = [x_c, x_d]^T$, where x_c represents the continuous variables Δu, and x_d are those of the logical variables φ_i. Introducing the matrix $\mathbf{Q}_{(3N_u \times 3N_u)}$ and $\mathbf{l}_{(3N_u \times 1)}$ defined as:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{H} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \\ \underline{\mathbf{0}} & \underline{\mathbf{0}} & \underline{\mathbf{0}} \end{bmatrix}; \mathbf{l} = \begin{bmatrix} \mathbf{b} \\ \hat{\mathbf{0}} \\ \hat{\mathbf{0}} \end{bmatrix} \tag{5}$$

where $\underline{\mathbf{0}} = N_u \times N_u$, $\hat{\mathbf{0}} = N_u \times 1$ both of zeros, $\mathbf{H}$ and $\mathbf{b}$ are a matrices used in classical QP optimization, the GPC optimization problem is expressed as:

$$\min_{\boldsymbol{x}} \boldsymbol{x}^T \mathbf{Q}\boldsymbol{x} + \mathbf{l}^T \boldsymbol{x} \tag{6}$$

subject to (4), which is a *Mixed Integer Quadratic Programming* (MIQP) optimization problem (Bemporad and Morari, 1999). The optimization problem involves a quadratic objective function and a set of mixed linear inequalities. Moreover, the classical set of constraints, $\mathbf{R}\Delta u \leq \mathbf{r}$ can also be included into the optimization procedure, introducing an auxiliary matrix $\hat{\boldsymbol{R}}$ of the form $[\boldsymbol{R}\ \overline{\mathbf{0}}\ \overline{\mathbf{0}}]$, where $\overline{\mathbf{0}}$ is a matrix of zeros with the same dimensions that $\boldsymbol{R}$. Finally, all constraints that must be considered into the optimization procedure are grouped in:

$$\begin{bmatrix} \boldsymbol{C} \\ \hat{\boldsymbol{R}} \end{bmatrix} \boldsymbol{x} \leqslant \begin{bmatrix} \boldsymbol{\rho} \\ \boldsymbol{r} \end{bmatrix}$$

In such a way, the event-based GPC with actuator deadband obtains optimal control signal values considering established deadband and classical constraints.

2.3 Complete event-based GPC scheme

The complete event-based GPC control scheme considers both sensor and actuator deadband at the same time. To realize such an event-based control structure, it is necessary to build the control structure introduced for the sensor deadband approach. Additionally, the developed controllers consider the actuator deadband in the optimization procedure. In this way, an event-based GPC controller manages two sources of events, related to process output and input, respectively. Thus, the complete event-based GPC control structure has two additional tuning parameters, β_y and β_u, which determine the deadband for the sensor and the actuator, respectively. Each of these tuning parameters, can be used to obtain the desired trade-off between control performance and the number of events for the sensor and the actuator, independently.

In this configuration, the process output is sampled using the intelligent sensor, where the deadband sampling logic is implemented. When one of the conditions becomes true, the event generator transmits the current process output to the controller node. The usage of the sensor deadband allows one to reduce the process output events, E_y. Afterwards, the received information in the event-based controller node triggers the event detector to calculate the time elapsed since the last event. The obtained time value is used as the current sampling time and the corresponding controller is selected to calculate a new control signal. Because the virtual actuator deadband is also used in such a configuration, the corresponding constraints on the control signal are active for all controllers from the set. In this way, the resulting control signal takes into account the deadband and makes the reduction of process input events E_u possible.

3. RESULTS

This section shows the simulation results of the event-based control strategy applied to the greenhouse inside temperature control problem. The simulations presented in this section have been performed using the TrueTime MATLAB/Simulink toolbox. The TrueTime is a tool developed to simulate real-time systems, networked control systems, communication models, and wireless sensor networks (Cervin et al., 2010).

To compare classical time-based (TB) and event-based (EB) configuration a specific performance indexes for this type of control strategies were considered (Ploennigs et al., 2010). As a first measure, the Integrated Absolute Error (IAE) is used to evaluate the control accuracy $IAE = \int_0^\infty |e(t)|dt$. The IAEP compares the event-based control with the time-based control used as a reference $IAEP = \int_0^\infty |y_{TB}(t) - y_{EB}(t)|dt$, where $y_{TB}(t)$ is the response of the time-based classical GPC. An efficiency measure index for event-based control systems can be defined as $NE = \frac{IAEP}{IAE}$. Additionally, E_y and E_u measurements are

considered to show the number of events for the process output and input, respectively.

The greenhouse inside temperature problem, used as the test bench, can be considered as a MISO (Multi-Input Single-Output) system [1], where the soil temperature, $v_1(t)$, the solar radiation, $v_2(t)$, the wind velocity, $v_3(t)$, the outside temperature, $v_4(t)$, are the input disturbances; the vents opening percentage, $u(t)$, control varialbe; and the inside temperature, $y(t)$, is the output variable. Considering the previous description, the CARIMA model for this system is given by:

$$A(z^{-1})y(t) = z^{-d}B(z^{-1})u(t-1) + \sum_{i=1}^{4} z^{-d_{D_i}} D_i(z^{-1})v_i(t) + \frac{\varepsilon(t)}{\Delta} \tag{7}$$

where $z^{-d_{D_i}}$ and $A(z^{-1})$, $B(z^{-1})$, $D_i(z^{-1})$ are time delays and polynomials used to describe the dynamics between the input, the disturbances, and the process output:

$$\begin{aligned}
A(z^{-1}) &= 1 - 0.3682z^{-1} + 0.0001z^{-2} \\
B(z^{-1})z^{-d} &= (-0.0402 - 0.0027z^{-1})z^{-1} \\
D_1(z^{-1})z^{-d_{D_1}} &= (0.1989 + 0.0924z^{-1} + 0.1614z^{-2})z^{-2} \\
D_2(z^{-1})z^{-d_{D_2}} &= (0.0001 + 0.0067z^{-1} + 0.0002z^{-2})z^{-1} \\
D_3(z^{-1})z^{-d_{D_3}} &= (-0.0002 - 0.3618z^{-1} + 0.0175z^{-2})z^{-1} \\
D_4(z^{-1})z^{-d_{D_4}} &= (0.0525 + 0.3306z^{-1} + 0.0058z^{-2})z^{-1}
\end{aligned}$$

The event-based GPC control structures were implemented with the following parameters. The sensor deadband configuration was implemented with $T_{base} = 1$ min, $T_{max} = 4$ min, $n_{max} = 4$, and thus $T_f \in [1, 2, 3, 4]$. The control horizon was selected to $N_u^{n_{max}} = 5$ samples. The prediction horizon was set to $N_2^c = 20$ min in continuous time, and the control weighting factor to $\lambda^f = 1$.

For the actuator deadband configuration, simulations were performed for the following system parameters: $N_2 = 10$, $N_u = 5$, and $\lambda = 1$. The minimum and maximum control signal increments of the vents opening percentage were set to $m = -20$ % and $M = 20$ %, respectively. The actuator virtual deadband was set to $\beta_u = [0.1, 0.5, 1, 2]$ in order to check its influence on the control performance.

The complete event-based GPC considers actuator and sensor deadbands at the same time. In this case, the control system configuration is as follows: $T_{base} = 1$ min, $T_{max} = 4$min, $n_{max} = 4$, and thus $T_f \in [1, 2, 3, 4]$. The control horizon was selected to $N_u^{n_{max}} = 5$ samples, the prediction horizon was set to $N_2^c = 20$, and the control signal weighting factor was adjusted to $\lambda^f = 1$. In this configuration, the actuator virtual deadband was set to $\beta_u = [0.1, 0.5, 1]$ and the sensor deadband $\beta_y = [0.2, 0.5]$.

Due to the physical limitation of the actuator, all controllers consider constraints on the control signal $0 \leq u(k) \leq 100\%$.

Table 1 collects performance indexes for all analyzed control configurations for 19 simulation days (using real data for disturbances) and Figure 3 shows graphical results for a representative day. As can be observed, the deadband values have a direct influence on the control performance obtained for the different configurations. The event-based GPC with sensor deadband is characterized by an important reduction for process output events E_y, where the number of events depends directly on the deadband value.

[1] http://aer.ual.es/CJPROS/engindex.php

For this event-based configuration, the sensor deadband value allows to establish a desired tradeoff between control performance and number of events (see Figure 3). The event-based GPC with actuator deadband is characterized by acceptable control performance for most of the tested deadband values, what is confirmed by the performance indexes from Table 1. Interesting results for this configuration are obtained for $\beta_u = 1$, where IAE increases about 15.9%, an event reduction of the order of 74% is obtained. Likewise in the previous configuration, the deadband value can be tuned to obtain a desired process input event reduction at an acceptable control accuracy. On the other hand, even the smaller actuator deadband, $\beta_u = 0.1$, results in an important event reduction, E_u, where savings of 9.9% are achieved compared to the time-based GPC.

The complete event-based configuration merges the advantages of both previously introduced methods. In the resulting configuration, the process input and output events can be tuned independently using the actuator or the sensor deadband, respectively. The analyzed scheme is characterized by acceptable control performance and obtains minimum and maximum IAE values between 1.4% and 33.6 % higher than the TB configuration. For the configuration with $[\beta_y = 0.5, \beta_u = 0.5]$, the best tradeoff between control performance and the number of events was obtained. In this case, IAE increases 10.6%, while E_y and E_u were reduced by about 41.3% and 46%, respectively. The bottom graph on Figure 3 shows how events are generated for this configuration.

4. CONCLUSIONS

This work presented a straightforward algorithm to implement a GPC controller with event-based capabilities. The core idea of the proposed control scheme was focused on the deadbands application for predictive algorithm. Three configurations for an event-based GPC were presented considering a sensor deadband approach, an actuator deadband approach, as well as the complete event-based scheme. In the developed event-based GPC configurations, the deadband values were used as additional tuning parameters to achieve the desired performance.

The main benefits of the event-based GPC has been highlighted using simulation study, where the event-based

Table 1. Numerical results for 19 days.

	Deadband		Performance indexes				
	β_u	β_y	IAE	E_u	E_y	IAEP	NE
TB	-	-	2275	5173	5173	-	-
EB	-	0.1	2292	4383	4383	715	0.31
	-	0.2	2343	3829	3829	727	0.31
	-	0.5	2799	2656	2656	1013	0.36
	-	0.75	3283	2161	2161	1457	0.44
	0.1	-	2277	4665	5173	12	0.01
	0.5	-	2352	2694	5173	300	0.13
	1	-	2637	1317	5173	845	0.32
	2	-	4457	453	5173	2894	0.65
	0.1	0.2	2307	4411	4530	489	0.21
	0.1	0.5	2525	2991	3049	486	0.19
	0.5	0.2	2416	3435	4271	717	0.29
	0.5	0.5	2516	2792	3005	492	0.19
	1	0.2	2597	2028	4957	939	0.36
	1	0.5	2998	2165	2693	800	0.26

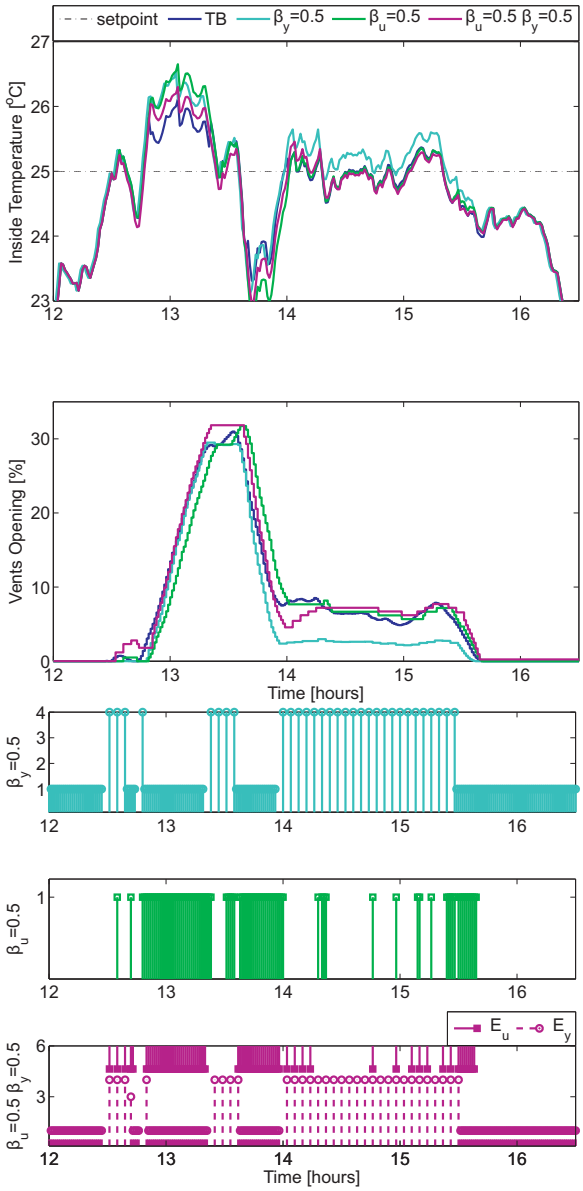

Fig. 3. Comparison of control results for different GPC configurations for a representative day.

control schemes were evaluated for the greenhouse temperature control problem. For each evaluated event-based GPC configuration, it was possible to reduce the control signal changes, which are associated to actuator wastage and economic costs, maintaining acceptable performance results.

ACKNOWLEDGEMENTS

This work has been partially funded by the following projects: DPI2011-27818-C02-01 and DPI2012-31303 (financed by the Spanish Ministry of Economy and Competitiveness and ERDF funds) and Controlcrop PIO-TEP-6174 (financed by the Consejería de Economía, Innovación y Ciencia de la Junta de Andalucía).

REFERENCES

Anta, A. and Tabuada, P. (2010). To sample or not to sample: Self-triggered control for nonlinear systems. *IEEE T Automat Contr*, 55, 2030–2042.

Årzén, K.E. (1999). A simple event-based PID controller. *Proc. of 14th IFAC World Congress*. Beijing, China.

Åström, K.J. (2007). Event-based control. *In* Analysis and Design of Nonlinear Control Systems, 127–147. Springer-Verlag.

Bemporad, A., Cairano, S.D., Henriksson, E., and Johansson, K.H. (2009). Hybrid model predictive control based on wireless sensor feedback: an experimental study. *Int J Robust Nonlin Contr*, 20, 209–225.

Bemporad, A. and Morari, M. (1999). Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35, 407–427.

Beschi, M., Dormido, S., Sánchez, J., and Visioli, A. (2012). Characterization of symmetric send-on-delta PI controllers. *J Process Contr*, 22, 1930–1945.

Can, Y., Shan-an, Z., Wan-Zeng, K., and Li-ming, L. (2006). Application of generalized predictive control in networked control system. *J. Zhejiang Univ.*, 7, 225–233.

Cervin, A., Henriksson, D., and Ohlin, M. (2010). Truetime 2.0 beta 5 - reference manual. *Department of Automatic Control, Lund University*. Lund, Sweden.

Chen, D., Xi, N., Wang, Y., Li, H., and Tang, X. (2008). Event based predictive control strategy for teleoperation via internet. *Proc. Conf. on Advanced Inteligent Mechatronics*. Xi'an, China.

Gawthrop, P.J. and Wang, L. (2007). Intermittent model predictive control. *J Syst Contr Eng*, 221, 1007–1018.

Hu, W., Liu, G., and Rees, D. (2007). Event-driven networked predictive control. *IEEE T Ind Electron*, 54, 1603–1613.

Jugo, J. and Eguiraun, M. (2010). Experimental implementation of a networked input-output sporadic control system. *Proc.of the IEEE Int. Conf. on Control Applications*. Yokohama, Japan.

Pawlowski, A., Cervin, A., Guzmán, J.L., and Berenguel, M. (2014). Generalized predictive control with actuator deadband for event-based approaches. *IEEE T Ind Inform*, 10, 523–537.

Pawlowski, A., Guzmán, J.L., Normey-Rico, J.E., and Berenguel, M. (2012). A practical approach for Generalized Predictive Control within an event-based framework. *Comput Chem Eng*, 41, 52–66.

Ploennigs, J., Vasyutynskyy, V., and Kabitzsch, K. (2010). Comparative study of energy-efficient sampling approaches for wireless control networks. *IEEE T Ind Inform*, 6, 416–424.

Quevedo, D.E., Østergaard, J., and Nešić, D. (2011). Packetized predictive control of stochastic systems over bit-rate limited channels with packet loss. *IEEE T Automat Contr*, 56, 2855–2868.

Sánchez, J., Visioli, A., and Dormido, S. (2011). A two-degree-of-freedom PI controller based on events. *J Process Contr*, 21, 639–651.

Varutti, P., Faulwasser, T., Kern, B., Kogel, M., and Findeisen, R. (2010). Event-based reduced attention predictive control for nonlinear uncertain systems. *IEEE Int. Symp. on Computer-Aided Control System Design*. Yokohama, Japan.

Varutti, P., Kern, B., Faulwasser, T., and Findeisen, R. (2009). Event-based model predictive control for networked control systems. *Proc. of the 48th IEEE Conf. on Decision and Control*. Shanghai, China.