Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

CSPS: an interactive tool for control design and analysis of processes with industrial characteristics ⋆

Lucian R. da Silva * **Rodolfo C. C. Flesch** **
Julio E. Normey-Rico ** **Paulo R. C. Mendes** ***

* *Programa de Pós-Graduação em Engenharia de Automação e Sistemas, Universidade Federal de Santa Catarina, 88040-900, Florianópolis, SC, Brazil (e-mail: lucian.ribeiro@posgrad.ufsc.br)*
** *Departamento de Automação e Sistemas, Universidade Federal de Santa Catarina, 88040-900, Florianópolis, SC, Brazil (e-mail: rodolfo.flesch@ufsc.br, julio.normey@ufsc.br)*
*** *Fraunhofer Institute for Industrial Mathematics, Kaiserslautern, Germany (e-mail: paulo.mendes@itwm.fraunhofer.de)*

Abstract: This work presents an user friendly interactive tool for control design, simulation and analysis of systems with characteristics commonly found in industry, such as dead time, constraints and measurement noise. The tool is able to validate and compare, in a simple and intuitive way, the performance and robustness of the three control structures most widely used in industrial applications: proportional-integral-derivative (PID), dead-time compensators (DTC), and model predictive control (MPC). Furthermore, the tool provides several options of techniques for handling input and output process constraints. A case study is used to illustrate some of the features of the tool.

Keywords: PID, dead-time compensator, model predictive control, constraints, control education.

1. INTRODUCTION

Most processes in industry present dead times, constraints and measurement noise. Processes with these characteristics are typically more difficult to control and therefore need more attention when choosing which control strategy to use (Normey-Rico and Camacho, 2007; Torrico et al., 2018).

The presence of dead time on processes makes the controller tuning a complex task, due to the reduction of stability margin caused by the dead time (Visioli, 2006). Proportional, integral and derivative (PID), dead-time compensators (DTC), and model predictive control (MPC) are the three of the most widely used control techniques in industry to deal with processes with dead time (Samad, 2017). Such control structures have different characteristics to deal with constraints and noise, in addition to different robustness characteristics.

To solve many practical problems that present this type of characteristics, advanced mathematics and control engineering concepts are generally involved. Many students in the field have difficulties to assimilate the mathematical concepts behind each control strategy, which leads to a decrease in student motivation (Méndez et al., 2006). In such cases, user friendly tools, which allow the students to simulate systems with different control structures and strategies considering characteristics commonly found in practical applications, can be useful in teaching or as supporting material. In the last decade, several interactive tools were proposed to facilitate the teaching of concepts in control engineering. In Guzman et al. (2005) a generalized predictive control interactive tool (GPCIT) is proposed to help students to understand basic and advanced concepts of generalized predictive control (GPC) strategy. An equivalent tool for teaching PID concepts was proposed in Guzmán et al. (2006). Some tools also support the comparison between different control structures, such as the web-based tool for analysis and simulation of automatic control systems using PID or state feedback controllers presented in Méndez et al. (2006) or the interactive tool to facilitate the design of PID, DTC and MPC controllers for processes with dead time proposed in da Costa Filho and Normey-Rico (2009).

In this work an user friendly interactive tool with graphical user interface for control design, simulation and analysis of closed-loop systems, considering PID, DTC and MPC controllers, is proposed. The tool provides to the user the capability to simulate and analyze single input single output (SISO) processes with dead time, considering measurement noise and constraints. Furthermore, the user can analyze the robustness of the system using robustness indices provided by the tool. Several anti-windup techniques are implemented, as well as tuning rules for the controllers which consider dead-time plants. Thus, in

⋆ This work was supported by the Brazilian National Council for Scientific and Technological Development (CNPq) under Grants 142342/2019-0, 309244/2018-8 and 305785/2015-0.

addition to providing valuable support for lecturers and students, it is an excellent tool to help engineers to decide when to use a simple PID controller in a certain process instead of more complex solutions, such as DTC or MPC strategies.

The paper is organized as follows. Section 2 briefly presents the control strategies for dead-time processes which are considered in the tool. In Section 3 the proposed tool is presented and its features are described. Section 4 presents a case study to illustrates some of the functionalities of the tool. The last section presents the conclusions.

2. CONTROL OF PROCESSES WITH DEAD TIME AND CONSTRAINTS

In this section the structures of the controllers available in the proposed tool for controlling dead-time processes with constraints are briefly described.

2.1 Dead time compensators

The Smith predictor (SP), shown in discrete-time domain in Fig. 1 if $F_r(z) = 1$, proposed in Smith (1957), is probably one of the most widely used dead-time compensation techniques (Normey-Rico and Camacho, 2007). In Fig. 1, $C_{sp}(z)$ is the primary controller, ZOH is a zero-order hold, $P(s)$ is the plant, T_s is the sampling time, $P_n(z) = G_n(z)z^{-d_n}$ is the nominal model, $G_n(z)$ is the dead-time-free model, d_n is the nominal dead time, k is the discrete time in samples, $r(k)$ is the reference, $u(k)$ is the control signal, $y(k)$ is the plant output, $e(k)$ is the error, $q(t)$ is the load disturbance, $\hat{y}(k)$ is the model output, $e_p(k)$ is the prediction error, and $F_r(z)$ is a robustness filter. The original SP structure presents two main drawbacks: it cannot eliminate the open-loop poles from disturbance rejection response and it cannot be used to control integrating and unstable processes. One of the solutions for these drawbacks, known as filtered Smith predictor (FSP), was described in Normey-Rico and Camacho (2008). The idea of this technique is to consider a filter, $F_r(z)$, in the prediction error of the original SP structure, as shown in Fig. 1. By using a properly tuning of $F_r(z)$ it is possible either to improve the dynamics of disturbance rejection or the closed-loop robustness. In addition, it can be used to control integrating and unstable processes.

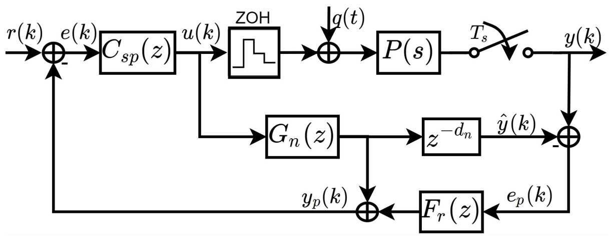

Fig. 1. Filtered Smith predictor (equivalent to SP if $F_r(z) = 1$)

2.2 PID approximation of FSP

In Normey-Rico and Guzmán (2013), a PID tuning rule based on a low frequency approximation of FSP is proposed. This tuning rule can be used to control processes modeled as first order plus dead time (FOPDT), integrating plus dead time (IPDT) or unstable first order plus dead time (UFOPDT). The idea of this rule is to obtain an equivalent controller of the FSP in a two-degree-of-freedom (2DOF) structure by using a Padé approximation for the dead time, which results in a PID series controller given by

$$C_{pid}(s) = \frac{K_c(sT_i + 1)(sT_d + 1)}{sT_i(s\alpha T_d + 1)}. \quad (1)$$

The PID parameters (K_c, T_i, T_d and α) are computed based on the chosen value for the desired closed-loop time constant, T_0, and also on the parameters of the model used to represent the process dynamics (see Normey-Rico and Guzmán (2013) for details). The parameter T_0 is the only tuning parameter and also defines a balance in the trade-off between performance and robustness.

2.3 MPC

MPC is a control strategy which uses the process model to predict future outputs and calculate an optimal control action (Camacho and Bordons, 2013). The GPC, proposed in Clarke et al. (1987) is an MPC strategy that became widely used in both industry and academia. It uses the discrete-time model

$$A(z^{-1})y(k) = z^{-d}B(z^{-1})u(k-1) + \frac{T(z^{-1})\eta(k)}{\Delta} \quad (2)$$

to predict the plant future outputs which are used to calculate, by minimizing a cost function, the control action (Camacho and Bordons, 2013). In (2), $T(z^{-1})$ is a polynomial in the backshift operator z^{-1} that represents the stochastic characteristics of the noise, d is the dead time, $\eta(k)$ is a zero-mean white noise, $\Delta = (1 - z^{-1})$, and $A(z^{-1})$ and $B(z^{-1})$ are polynomials in z^{-1}. The function to be minimized is given by

$$J = \sum_{j=d_n+1}^{d_n+N} [\hat{y}(k+j|k) - r(k+j)]^2 + \sum_{j=1}^{N_u} \lambda[\Delta u(k+j-1)]^2, \quad (3)$$

where N is the prediction horizon, N_u is the control horizon, λ is the control increment weight, $\hat{y}(k+j|k)$ is the predicted output for $k+j$ at time instant k, $r(k+j)$ is the future reference, and $\Delta u(k)$ is the control increment. If the process is subjected to constraints, the minimization of (3) does not have analytical solution, and needs to be solved by using numerical methods or other techniques (Camacho and Bordons, 2013).

In Normey-Rico and Camacho (2007) it is shown that in the unconstrained case GPC can be represented as a 2DOF FSP in the discrete-time domain with a reference filter, $F(z)$, a primary controller, $C_{sp}(z)$, and a predictor filter, $F_r(z)$. This representation can also be used to analyze robustness properties of GPC using classical robustness techniques. Furthermore, in Normey-Rico and Camacho (2007) it is demonstrated that it is possible to compute the output predictions up to $k + d_n$ using an FSP structure and from $k + d_n + 1$ to $k + d_n + N$ using the normal GPC procedure. With this formulation, it is possible either to improve the dynamics of disturbance rejection or the closed-loop robustness of the original GPC by choosing a different format for $F_r(z)$. This approach became known as dead time compensator generalized predictive controller (DTC-GPC).

3. TOOL DESCRIPTION

The constrained SISO-process simulator (CSPS) tool, shown in Fig. 2, was developed using MATLAB Graphic User Interface Design (GUIDE). CSPS tool allows the user to simulate SISO dead-time processes using PID, DTC and MPC controllers. Also, the tool considers three types of process constraints: saturation in magnitude and rate of change of control signal and in the output of the process. For dealing with process constraints the tool provides two options. The first one is to use an MPC strategy which handles process constraints by using an optimization procedure to find an optimal control action. The other option is to use anti-windup techniques, which are able to reduce the degradation of the closed-loop performance caused by the windup phenomenon (Hippe, 2006).

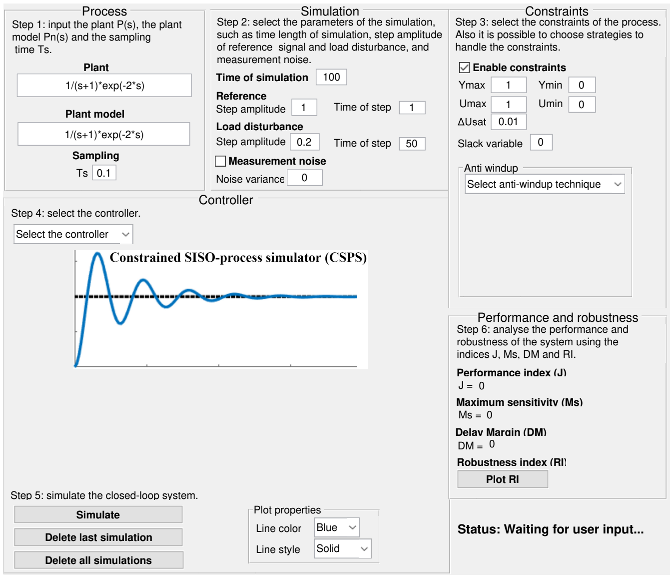

Fig. 2. Constrained SISO-process simulator (CSPS)

The main properties of the CSPS tool are:

- user friendly graphical user interface for simulation and analysis of SISO processes with characteristics commonly found in industrial applications;
- performance analysis considering IAE performance index for setpoint tracking and load disturbance rejection;
- easy comparative analysis between different control strategies widely used in practice, such as PID, DTC, and MPC;
- performance analysis of different techniques used to handle process constraints;
- robustness analysis including important robustness measures.

The graphical interface is subdivided in five panels: *Process*, *Simulation*, *Constraints*, *Controller* and *Performance and robustness*. The functionalities of each panel are described in the next sections.

3.1 Process description

Firstly, in *Process* panel, the user must input a model which is the representation of the plant to be controlled, $P(s)$, and the model of the process, $P_n(s)$. The former is used as plant in all simulations, while the latter is used as plant model in the model-based approaches. Both models are represented as continuous-time transfer functions using variable s. For example, the plant model

$$P(s) = \frac{1}{s+1} e^{-2s}, \tag{4}$$

is input as `1/(s+1)*exp(-2*s)`.

The user must also input the sampling time, T_s, used for discretization of the models and controllers. All the process models are transformed into their discrete-time equivalents using the zero-order hold method, which assumes that the control signals are kept constant between two sampling instants. Despite some of the tuning rules available in the tool are defined in the continuous-time domain, all the controllers are discretized using the Tustin approximation technique and implemented in the discrete-time domain.

In *Simulation* panel, the user must input the duration of the simulation and the amplitude of the step used as reference signal. The tool also provides the option for considering a step load disturbance and measurement noise with normal distribution and variance specified by the user.

3.2 Handling of constraints

CSPS tool provides the option to consider process constraints. For this purpose, it is necessary to mark the check box *enable constraints*, in *Constraints* panel, which then allows the user to define limits for the magnitude of control action (being $U_{\min}$ and $U_{\max}$ the minimum and maximum values, respectively), rate of change of control action (being ΔU_{sat} the saturation limit) and magnitude of the output of the process (being $Y_{\min}$ and $Y_{\max}$ the minimum and maximum values, respectively).

For dealing with input constraints the tool provides three AW techniques widely used in practice for controllers which do not consider the constraints a priori (PID and DTC). The first one is the incremental algorithm (or velocity algorithm), which consists of calculating a control increment at each sampling period and adding to the previous control signal only the amount that does not saturate the actuator (Åström and Wittenmark, 1984). This technique is widely used in industry for its simplicity of implementation in digital controllers. The second one is the back-calculation technique, proposed by Fertik and Ross (1967), which consists in adding an extra feedback signal to the input of the integrator, which is composed of the error between the output signal of the controller and the signal that is applied to the plant multiplied by a constant gain, T_t, known as tracking time parameter. The last one is the error recalculation (ER) technique, proposed in (Bruciapaglia and Apolônio, 1986; Flesch et al., 2017), which consists in modifying the current control signal and the current error signal to maintain the consistency between the control signal calculated by the controller and the input signal that is effectively applied to the plant. The main advantages of this technique are: it does not need an additional tuning parameter and it has good performance when applied to processes with measurement noise (da Silva et al., 2018).

For dealing with output constraints, the tool provides the option *constraints mapping*, which uses an approach based on the *clipping* technique, used in MPC strategies. The main idea of this technique is to calculate future output predictions using the process model, considering a prediction horizon, N, and compute a control action which guarantees that all output predictions are inside the region delimited by the constraints (see da Silva et al. (2019) for details).

3.3 Available controllers

CSPS tool provides three different control strategies for simulation: PID, DTC and MPC. These controllers can be select in *Controllers* panel. For the PID strategy, it is possible to use two different tunings, being the first one the approach proposed in Normey-Rico and Guzmán (2013), which uses only one tuning parameter, T_0. The second one is a manual tuning in which the user can freely choose the parameters of a series PID with the structure defined in (1).

For the DTC option, which is implemented as an FSP, there are three tuning options. The first one is also based on the approach presented in Normey-Rico and Guzmán (2013) and uses T_0 as tuning parameter. The second tuning is an FSP in the discrete-time domain based on the GPC strategy, which can provide the same performance as the GPC for the unconstrained case (see Normey-Rico and Camacho (2007) for details). For this tuning, the user must specify three tuning parameters: prediction horizon, N, control horizon, N_u, and control increment weight, λ (tracking error weighting factor is assumed as $\delta = 1$). The last one is a manual tuning in which the user can freely set the parameters of a discrete FSP with reference filter, $F(z)$, primary controller, $C_{sp}(z)$, and robustness filter, $F_r(z)$.

For the MPC options, the tool provides two strategies: GPC (Clarke et al., 1987) and DTC-GPC (Normey-Rico and Camacho, 2007). For the two strategies, the user must specify the tuning parameters N, N_u and λ. For the DTC-GPC option, the user must also specify the robustness filter, $F_r(z)$, which is used in the predictor structure. Both strategies use a quadratic programming solver provided by MATLAB, to find the optimal control action which satisfies all the constraints. If the optimization procedure results in an unfeasible solution, the tool will show a message indicating that a new tuning of the controller is necessary.

3.4 Performance and robustness evaluation

To quantify the performance of the closed-loop system CSPS tool uses a cost function, J, which considers the integral of absolute error index (IAE) for setpoint tracking and load disturbance rejection. Just periods of time where the control signal can affect the process output due to the delay are considered in the cost function, which is given by

$$J = \frac{1}{2}\left[\int_{t_s+L_n}^{t_d} |r(t)-y(t)|\mathrm{d}t + \int_{t_d+2L_n}^{\infty} |r(t)-y(t)|\mathrm{d}t\right], \quad (5)$$

where t_s is the time at which the reference change is commanded, L_n is the dead time and t_d is the time at which the disturbance is applied (da Silva et al., 2019).

CSPS tool provides three important indices for robustness analysis of the system: D_M, M_s and R_I. The first one is given by

$$D_M = \frac{P_M}{\omega_c}, \quad (6)$$

where P_M is the phase margin (given in rad) and ω_c is the crossover frequency (given in rad/s). This index is used to evaluate the robustness against uncertainties in dead time. D_M represents the smallest amount of time delay which causes the closed-loop system to become unstable (Palmor, 1980).

M_s index is typically used to measure the robustness of a system when modeling errors are not estimated and is given by (Åström and Hägglund, 1995)

$$M_s = \max_{\omega} |1 + C(j\omega)P_n(j\omega)|^{-1}, \quad (7)$$

where $C(j\omega)$ is the controller and $P_n(j\omega)$ is the process model used for tuning $C(j\omega)$. According to Åström and Hägglund (1995), the values of M_s typically used in industry are between 1.2 and 2.0.

$R_I(\omega)$ is used to check the robust stability condition $R_I(\omega) > \overline{\delta P}(\omega)$, $\forall\, \omega \geq 0$, where $\overline{\delta P}(\omega)$ is the multiplicative uncertainty, $P(\omega) = P_n(j\omega)[1 + \delta P(j\omega)]$, $\overline{\delta P}(\omega) \geq |\delta P(j\omega)|$, $\forall \omega \geq 0$. For a generalized control structure, without any transfer function in the feedback loop, $R_I(\omega)$ can be computed as

$$R_I(\omega) = \frac{|1 + C(j\omega)P_n(j\omega)|}{|C(j\omega)P_n(j\omega)|} \quad \forall\, \omega \geq 0 \quad (8)$$

where $C(s)$ is the controller and $P_n(s)$ is the process model used to tune $C(s)$.

The J, M_s and D_M indices are shown in *Performance and robustness* panel after the system is simulated, and the R_I index can be visualized by pressing the button *Plot RI* at the same panel.

4. CASE STUDY

In this section, an analysis of performance and robustness for a case study is presented to better illustrate the use of CSPS tool. The case study aims to explore the features of the proposed tool, not focusing on the performance or robustness of the controllers.

The plant considered in this section is a boiler, presented in Normey-Rico and Camacho (2007). The process is described by the linear model

$$P(s) = \frac{2e^{-5s}}{s(s+1)(0.5s+1)(0.1s+1)}. \quad (9)$$

For tuning the controllers, the dynamics of the process were approximated by an IPDT model given by

$$P_n(s) = \frac{2}{s}e^{-6.5s}, \quad (10)$$

with time given in minutes. The sampling time used for simulating the process and discretizing the controllers is

$T_s = 0.5\,\text{min}$. The first simulation does not take into account measurement noise and considers constraints in magnitude and rate of change of the control signal, $U_{\min} = -0.05$, $U_{\max} = 0.05$, $\Delta U_{\text{sat}} = 0.01$, and in the output of the process, $Y_{\min} = 0$ and $Y_{\max} = 1.1$. In this case a PID controller is used and it is tuned for fast performance with $M_s = 4.6$ using the rule presented in Normey-Rico and Guzmán (2013), with a closed loop time constant of $T_0 = 4\,\text{min}$, resulting in a controller $C_{pid}(s)$ and a reference filter $F(s)$ given by

$$C_{pid}(s) = \frac{0.45(s+0.30)(s+0.06)}{s(s+1.71)}, \tag{11}$$

$$F(s) = \frac{0.27(s+0.25)}{s+0.06}. \tag{12}$$

Fig. 3 shows the simulation of the closed-loop system for a unit step reference at $t = 1\,\text{min}$ and a load disturbance of amplitude -0.04 at $t = 100\,\text{min}$, considering three cases: without AW; with ER AW; and with ER AW and constraints mapping (CM), which is used to handle output constraints, with a prediction horizon of $N = 6$.

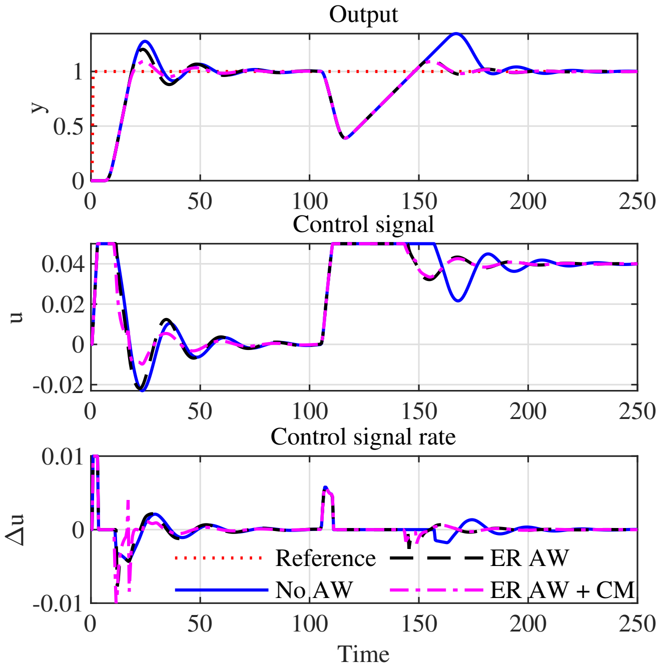

Fig. 3. Closed-loop performance of the integrating case without measurement noise

As can be seen in Fig. 3, the case without AW presented high overshoot and oscillations, caused by saturation constraints, resulting in a performance index of $J_{NOAW} = 32.65$. The PID controller with ER AW was able to reduce overshoot and oscillations, providing a better performance when compared to the case without AW and resulting in $J_{ER} = 26.65$. The last case presented the best performance, being able to reduce the overshoot and also able to deal with all the constraints considered, resulting in a performance index of $J_{ER+CM} = 24.42$. The obtained value for D_M in this case is $D_{M_{PID}} = 1.87\,\text{min}$.

The second simulation of the system considers measurement noise with normal distribution and variance of 0.03. In this case, the performance of FSP without AW, FSP with incremental algorithm (IA) AW and DTC-GPC are compared. Both controllers are tuned for a robust solution with $M_s = 2.0$. The FSP was tuned considering $T_0 = 8\,\text{s}$, resulting in a primary controller, $C_{sp}(s) = 0.06$, and robustness filter

$$F_r(s) = \frac{2.81(s+0.04)}{s+0.12}. \tag{13}$$

The DTC-GPC was tuned with $N = 40$, $N_u = 12$, $\lambda = 15 \times 10^3$ and a discrete-time robustness filter

$$F_r(z) = \frac{1.13z - 1.12}{z - 0.99}. \tag{14}$$

Furthermore, in this case, the output constraints were relaxed in DTC-GPC to avoid infeasibility of the optimization procedure, due to noisy measurements.

Fig. 4 shows the simulation of the system considering measurement noise. As can be seen, the FSP with IA presented the best performance when compared to the other two cases, with no overshoot and fast disturbance rejection response. The performance indices obtained for this case for FSP without AW, FSP with AW and DTC-GPC are $J_{FSP-NOAW} = 35.1$, $J_{FSP-IA} = 32.1$ and $J_{DTC-GPC} = 52.7$, respectively. Both controllers presented similar robustness properties in terms of dead time uncertainties, presenting $D_{M_{FSP}} = 5.24\,\text{min}$ and $D_{M_{DTC-GPC}} = 5.71\,\text{min}$.

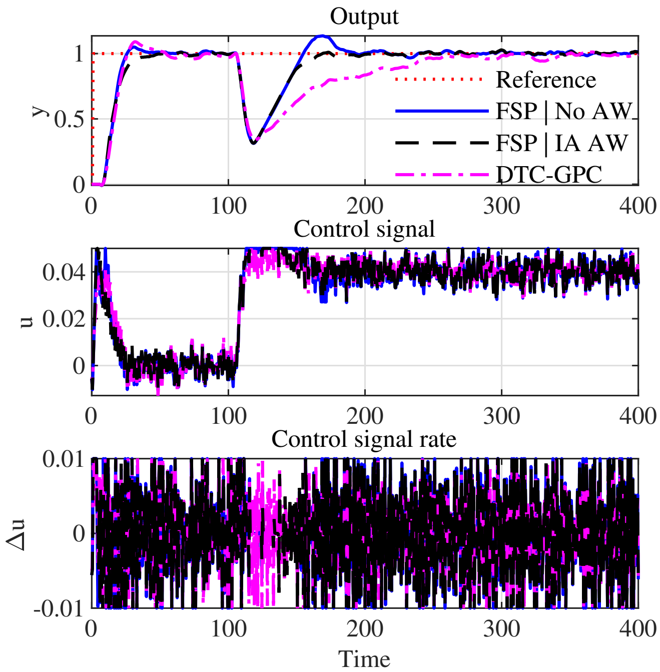

Fig. 4. Closed-loop performance of the integrating case with measurement noise

In Fig. 5 the robustness index, $R_I(\omega)$, of the three controllers and the modeling error, $\delta P(\omega)$, of the process are shown. As can be seen, the robustness properties of FSP and DTC-GPC were very similar. On the other hand, as expected, the PID tuned for fast performance was considerably less robust when compared to the other two strategies.

All the results of the performance and robustness analysis presented in this case study were easily obtained by using

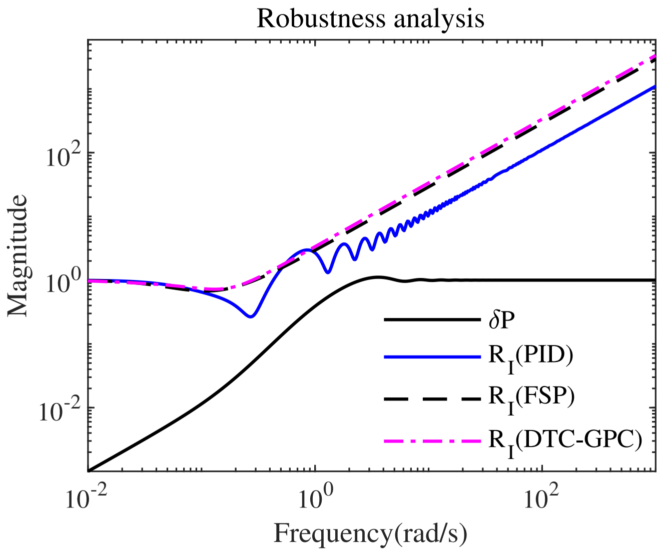

Fig. 5. Robustness analysis

the proposed tool. In addition, the tuning procedure of the controllers is easy and intuitive, and the performance and robustness comparison between different control structures can be done in a simple way.

5. CONCLUSIONS

In this work, an user friendly interactive tool with graphical interface for simulation and analysis of SISO processes with dead time and constraints is described. The main features of the tool include: easy tuning of controllers widely used in industry; performance analysis of the closed-loop response of processes including characteristics commonly found in practical applications; possibility to include anti-windup action in the structure of the controllers; robustness analysis including important robustness indices. These characteristics make the proposed tool a good option for teaching important concepts of control engineering. Furthermore, the tool can be used to decide the best control strategy to be used based on the characteristics of the process. A case study considering an integrating process was presented for a better illustration of the features of the tool. The CSPS tool is available for download at `http://rodolfoflesch.prof.ufsc.br/cspstool`.

REFERENCES

Åström, K.J. and Hägglund, T. (1995). PID controllers: theory, design, and tuning.

Åström, K.J. and Wittenmark, B. (1984). *Computer Controlled Systems: Theory and Design*. Prentice Hall.

Bruciapaglia, A. and Apolônio, R. (1986). Uma estratégia de eliminação da sobrecarga da ação integral para controladores PID discretos: Aplicação no controle de velocidade de um motor de corrente contínua. In *II Congresso Latino Americano de Controle Automático. Buenos Aires, Argentina: AADECA*, 519–524, (*in Portuguese*).

Camacho, E.F. and Bordons, C. (2013). *Model Predictive Control*. Springer, London.

Clarke, D.W., Mohtadi, C., and Tuffs, P. (1987). Generalized predictive control—Part i. The basic algorithm. *Automatica*, 23, 137–148.

da Costa Filho, M.V.A. and Normey-Rico, J.E. (2009). An interactive tool to design controllers for processes with dead time. *IFAC Proceedings Volumes*, 42(14), 189–194.

da Silva, L.R., Flesch, R.C.C., and Normey-Rico, J.E. (2018). Analysis of anti-windup techniques in PID control of processes with measurement noise. In *3rd IFAC Conference on Advances in Proportional-Integral-Derivative Control*, 948–953. Ghent, Belgium.

da Silva, L.R., Flesch, R.C.C., and Normey-Rico, J.E. (2019). Controlling industrial dead-time systems: When to use a PID or an advanced controller. *ISA Transactions*. doi:10.1016/j.isatra.2019.09.008.

Fertik, H.A. and Ross, C.W. (1967). Direct digital control algorithm with anti-windup feature. *ISA Transactions*, 6(4), 317.

Flesch, R.C., Normey-Rico, J.E., and Flesch, C.A. (2017). A unified anti-windup strategy for SISO discrete dead-time compensators. *Control Engineering Practice*, 69, 50–60.

Guzmán, J.L., Åström, K.J., Dormido, S., Hägglund, T., and Piguet, Y. (2006). Interactive learning modules for PID control. *IFAC Proceedings Volumes*, 39(6), 7–12.

Guzman, J.L., Berenguel, M., and Dormido, S. (2005). Interactive teaching of constrained generalized predictive control. *IEEE Control Systems Magazine*, 25(2), 52–66.

Hippe, P. (2006). *Windup in Control: Its Effects and Their Prevention.* Springer Science & Business Media.

Méndez, J.A., Lorenzo, C., Acosta, L., Torres, S., and González, E. (2006). A web-based tool for control engineering teaching. *Computer Applications in Engineering Education*, 14(3), 178–187.

Normey-Rico, J.E. and Camacho, E.F. (2007). *Control of Dead-time Processes.* Springer, London.

Normey-Rico, J.E. and Camacho, E.F. (2008). Dead-time compensators: A survey. *Control Engineering Practice*, 16(4), 407–428.

Normey-Rico, J.E. and Guzmán, J.L. (2013). Unified PID tuning approach for stable, integrative, and unstable dead-time processes. *Industrial & Engineering Chemistry Research*, 52, 16811–16819.

Palmor, Z. (1980). Stability properties of Smith dead-time compensator controllers. *International Journal of Control*, 32, 937–949.

Samad, T. (2017). A survey on industry impact and challenges thereof. *IEEE Control Systems*, 37, 17–18.

Smith, J.O. (1957). Closed control of loops with dead time. *Chemical Engineering Progress*, 53, 217–219.

Torrico, B.C., de Almeida Filho, M.P., Lima, T.A., Forte, M.D.d.N., Sá, R.C., and Nogueira, F.G. (2018). Tuning of a dead-time compensator focusing on industrial processes. *ISA Transactions*, 83, 189–198.

Visioli, A. (2006). *Practical PID control.* Springer Science & Business Media.