A Novel PID Tunning Method for TISO (Two-Input Single-Output) System with Dead Time ⋆

Gustavo Catusso Balbinot [a], Rafael Oliveira Pedrisch [a], Sérgio Paulo Neves dos Santos [a], Kariston Dias Alves [a], José Diogo Forte de Oliveira Luna [a]

[a] *Federal Institute of Rondônia, Porto Velho, Brazil*

Abstract

Processes with multiple inputs and a single output are usual in industries. These processes present more than one actuator driving a single process variable. Recent literature has shown the possibility of developing new methodologies for controlling this class of processes, including dead time compensation for Two-Input Single-Output (TISO) systems. However, such techniques require an explicit prediction model, increasing computational cost and limiting its deployment in less flexible hardware. That said, the present work seeks to develop a Single-Input Two-Output (SITO) control strategy for TISO system, based on the proportional integral derivative (PID) control, with similar properties to those found in the literature, allowing its implementation in less flexible hardware. To this end, the mathematical deduction is presented and the proposed controller is validated to computer simulation. The results shown the proposed controller has similar behaviour to the original controller at low frequencies and poses itself as a promising alternative due to its lower computational cost. For ease of usage, a tuning table of the novel controller is delivered.

Key words: SITO processes; Dead-Time Compensation; Automatic Control.

1 Introduction

In industrial processes, it is relatively common to find systems that have multiple inputs and a single output. For example, in the case where you have many actuators operating together to drive a single process variable. For TISO (two-input single-output) systems, where there are two inputs, pairs of valves or heat exchangers are frequent examples. In the case of two valves, the maximum flow rate is provided when both valves are fully open. In the case of an intermediate flow rate, it is possible to have a fully open valve or both partially open [1].

A usual technique for MISO systems is split-range control. This technique comprises a single controller capable of commanding all actuators. The most common use of split-range control is to extend the steady-state range by switching to another actuator when the primary one saturates. The split-range control consists of a single SISO controller whose manipulated variable is processed by a splitter. This splitter is usually a set of affine functions which, in turn, outputs the driving signal for each of the actuators. Split-range control has been known since the 1940s [2] but has received renewed attention in the literature in the last decade. In 2019, Reyes-Lúa and Skogestad [3] presented a systematic design method of the split-range controller that can grant the most economical control action. Again in 2019, the same authors, in their work on advanced control for active constraint switching [4], noted that split-range control is a favorable choice for situations where the active constraint switches between two constraints on manipulated variables. Then after a year, Reyes-Lúa and Skogestad [5] proposed a generalized split-range control using a baton-passing strategy.

Another recent study is the paper of Fatani [6], where they discuss a split-range variation with a SITO-PI controller as the primary controller. In this approach, pure

⋆ The authors would like to thank IFRO - Campus Calama, CNPq, and CAPES for funding through the public notice Nº 5/2021/REIT - PROPESP/IFRO.

Email addresses: gu.academico@gmail.com (Gustavo Catusso Balbinot), rafaeloliveira01239@gmail.com (Rafael Oliveira Pedrisch), sergio.santos@estudante.ifro.edu.br (Sérgio Paulo Neves dos Santos), kariston.alves@ifro.edu.br (Kariston Dias Alves), jose.luna@ifro.edu.br (José Diogo Forte de Oliveira Luna).

proportional action is used in one actuator and pure integral in the other so the latter performs fine control and the first one executes a coarse control. The work shows that for the case of similar transport delays and time constants between the actuators, the PI design is the same as for an equivalent actuator. This result implies simplified design and robustness or stability analyses for the proposed control. The work overlooks, however, the application of dead-time compensation.

In turn, dead time, or transport delay, is a characteristic of a dynamic system that describes the time interval between the application of a stimulus and the perception of a reaction to that stimulus. This phenomenon can occur due to a delay in the transport of energy, mass, or information [7]. In control terms, a transport delay adds increasing phase delay as frequency increases, reducing the phase margin of the system and making it easier for it to become unstable. When faced with significant delays, in classical control, it is normal to opt for a conservative tuning that guarantees a robust, albeit slow, response [8].

Ideally, one would want to eliminate dead time by immediately sensing the system's reaction to the given stimulus. With this perspective, Smith (1959) [9] proposed the controller that came to be called Smith's Predictor, an internal model control strategy that uses a fast model (without delay), commanded by a primary controller, while feedbacking the error between the measured output and the output predicted by the complete model (with delay) of the process.

In this regard, the work of Balbinot [10] addressed the use of dead-time compensation, by an internal model strategy analogous to the classical Smith's Predictor, in a PI-SITO control structure like that of the work of Fatani [6], thus filling the latter's gap. However, this solution has the drawback of requiring a controller with more flexibility to implement the prediction model. Using the explicit prediction model requires both the implementation of the difference equations and the use of memory to store the previous samples. Its realization becomes challenging in less flexible hardware present in the industry, which relies on mostly programming blocks. Nevertheless, in Normey-Rico [7] a PID tuning based on Smith's Predictor was shown to deliver equivalent performance for first-order systems with dead-time, which allows implementation in basic hardware, as long as they have PID blocks.

Building upon these previous works, this paper addresses the drawback of the Balbinot [10] controller by presenting a novel control strategy for TISO systems with dead-time based on Normey-Rico's approximation. Therefore, the text is structured as follows: in the second section a mathematical demonstration is delivered; the third section presents simulation results; finally in the fourth section brings some final considerations.

2 Mathematical Demonstration

This section develops the mathematical results for the implementation of control strategy.

2.1 PI-SITO Controller

Fatani [6] discusses a Proportional-Integral (PI) controller technique, which receives a single input (the error signal) and operates two manipulated variables simultaneously, thus being a SITO controller.

The authors discussed an example of a process that consists of two valves of different sizes. In this strategy, proportional action (P) is used for the smaller valve, enabling quick response to disturbances. Concurrently, integral action (I) commands the larger valve, allowing it to achieve setpoint tracking.

Fatani [6] states that this controller maintains the valves in a favorable control region, with the larger valve seeking to reject the slow disturbances of the permanent regime and the smaller valve rejecting the fast disturbances, ensuring coordination between the valves with a single control algorithm. Fig. 1 exemplifies Fatani's proposal.

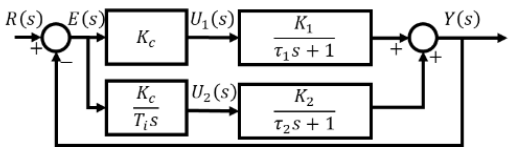

Fig. 1. The control structure proposed in [6]. $G_1(s) = \frac{K_1}{\tau_1 s+1}$ and $G_2(s) = \frac{K_2}{\tau_2 s+1}$ are the dynamics of the TISO process as driven by actuator 1 and actuator 2, respectively. K_c is the proportional gain and T_i is the time integral.

Consider that the process dynamics is described by:

$$G_1(s) = \frac{K_1}{\tau_1 s + 1}, G_2(s) = \frac{K_2}{\tau_2 s + 1}, \tag{1}$$

where K_1 and K_2 are the static gain, τ_1 and τ_2 are the positive time constants of the system.

Thus the output of each process is described by:

$$Y_1(s) = G_1(s)U_1(s), Y_2(s) = G_2(s)U_2(s), \tag{2}$$

where $Y_1(s)$ and $Y_2(s)$ are partial outputs of the system, $U_1(s)$ and $U_2(s)$ are the control signals.

Thus, the final output of the SITO process is:

$$Y(s) = Y_1(s) + Y_2(s), \tag{3}$$

where $Y(s)$ is the output of the system.

The error is defined as:

$$E(s) = R(s) - Y(s), \tag{4}$$

where $E(s)$ is the error between the setpoint and the process variable and $R(s)$ is the system setpoint.

The control signals are:

$$U_1(s) = K_c E(s), U_2(s) = \frac{K_c}{T_i s} E(s). \tag{5}$$

Applying (5) in (2) and (1) in (2), leads to:

$$Y_1(s) = \frac{K_1 K_c}{\tau_1 s + 1} E(s), Y_2(s) = \frac{K_2 K_c}{(\tau_2 s + 1) T_i s} E(s), \tag{6}$$

where K_c is the proportional gain and T_i is the integral time.

Adopting (6) in (3):

$$Y(s) = \left(\frac{K_1 K_c}{\tau_1 s + 1} + \frac{K_2 K_c}{(\tau_2 s + 1) T_i s} \right) E(s). \tag{7}$$

The relationship between the gains of the prosses can be simplified as, assuming that both time constants of the actuator are approximately the same:

$$\tau_1 = \tau_2 = \tau > 0, \ K_2 = \alpha K_1 \ \mid \ \alpha > 1,$$

where α is a ratio between the system static gains and τ is the positive equivalent time constant, therefore the system is assumed to be open-loop stable.

Then:

$$Y(s) = K_c \left(1 + \frac{\alpha}{T_i s} \right) \frac{K_1}{\tau s + 1} E(s). \tag{8}$$

Consider that:

$$T_i' = \frac{T_i}{\alpha},$$

where T_i' is the ratio between the integral time and α.

Finally, apply (4) in (8) in order to obtain the equation that represents the output:

$$Y(s) = \frac{K_c \left(1 + \frac{1}{T_i' s} \right) \frac{K_1}{\tau s + 1}}{1 + K_c \left(1 + \frac{1}{T_i' s} \right) \frac{K_1}{\tau s + 1}} R(s). \tag{9}$$

In (9) presents the closed-loop output equation. Thus, it can be said that the output will depend on an equivalent PI control action and an equivalent dynamics, so it is clear that this equation represents a SISO output equation, thus, rewriting (9):

$$Y(s) = \frac{C_{eq}(s) G_{eq}(s)}{1 + C_{eq} G_{eq}} R(s).$$

As it is shown that the output equation has been converted to a SISO output equation, all the SISO controller design and analysis techniques consolidated in the literature are valid for this controller.

2.2 PI-SITO with Dead-Time Compensation Controller

Building on Fatani's study [6], Balbinot [10] proposed dead-time compensation techniques for TISO process cases, where this delay is significant.

In particular, Smith's predictor (SP) [8] is a widely studied dead-time compensation technique. It can be seen in Figure 2, that the control structure is divided into two parts: a primary controller represented by $C(s)$ and the prediction structure. The prediction structure consists of two blocks: the first represents the process model without the dead time $G_m(s)$ which is also known as the fast model, and the second contains the delay model. The union of the two blocks represents the process output with dead time. The fast model is used to predict the open-loop response. Taking into account the modeling error, the difference between the process and model outputs including dead time is added to the predicted open-loop response. The primary controller is tuned to the fast model controller.

According [7] Smith's Predictor has three closed-loop characteristics under the nonexistence of modeling errors. The first feature of SP is the elimination of dead time in the closed-loop characteristic equation, that is: the closer the model is to the actual process delay the smaller the prediction error will be. The second feature is prediction, where the prediction signal produced by the SP anticipates the system output for changes in the setpoint. It is important to point out however that it is not valid in the presence of disturbance. And finally, the optimal dynamic compensation, SP structure implies billing the plant in two parts, i.e. the fast model and the dead-time model. As the delay is eliminated in the denominator in the close loop transfer function it is possible to design the controller for the fast model. These three features that SP brings allow the controller to perform better than PID for controlling processes with dead time in the absence of modeling error [11].

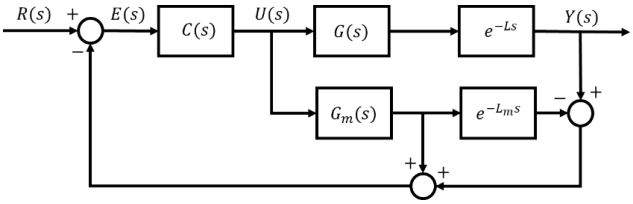

Fig. 2. SISO Smith's Predictor block diagram.

It is worth noting that the SP cant improve disturbance rejection, because it takes twice the dead time to the controller action to be perceived in the output.

Fig. 3 shows the proposed expansion of this work from Fatani's study [6] with the addition of dead-time compensation. For this work, the transport delays on the two channels are estimated to be identical or close enough to be considered equal.

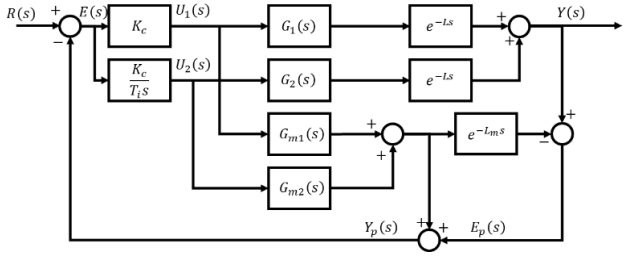

Fig. 3. Control structure with dead-time compensation. G_{m1} and G_{m2} are the fast models of the system, L is the dead time of the real system and finally L_m is the dead time model.

Taking the from [6] structure and adding the prediction and dead-time compensation capabilities of the SP, the obtained strategy is presented in Fig. 3.

$$Y(s) = G_1(s)U_1(s)e^{-Ls} + G_2(s)U_2(s)e^{-Ls}, \quad (10)$$

$$U_1(s) = E(s)C_1(s), \quad (11)$$

$$U_2(s) = E(s)C_2(s), \quad (12)$$

$$Y_p(s) = E_p(s) + (G_{m1}(s)U_1(s) + G_{m2}(s)U_2(s)), \quad (13)$$

$$E_p(s) = Y(s) - (G_{m1}(s)U_1(s) + G_{m2}(s)U_2(s))e^{-L_m s}, \quad (14)$$

$$E(s) = R(s) - Y_p(s), \quad (15)$$

where $C_1(s)$ and $C_2(s)$ represent the controllers, G_{m1} and G_{m2} are the fast models of the system, L is the dead time of the real system and finally L_m is the dead time model, $Y_p(s)$ is the sum of the fast model response and the prediction error, and $E_p(s)$ is the prediction error.

Equation (10) describes the effective plant output, (11) and (12) describe the controls signals, (13) describes the signal that will be subtracted from the reference, (14) describes the prediction error, and finally (15) describes the error.

Replacing (14) in (13):

$$Y_p(s) = Y(s) - (G_{m1}(s)U_1(s) + G_{m2}(s)U_2(s))e^{-L_m s} + (G_{m1}(s)U_1(s) + G_{m2}(s)U_2(s)). \quad (16)$$

Adopting (15) in (11) and (12):

$$U_1(s) = R(s)C_1(s) - Y_p(s)C_1(s). \quad (17)$$

$$U_2(s) = R(s)C_2(s) - Y_p(s)C_2(s). \quad (18)$$

Applying (17) and (18) in (10) and rearranging the equation:

$$Y(s) = (G_1(s)C_1(s) + G_2(s)C_2(s))e^{-Ls}R(s) - (G_1(s)C_1(s) + G_2(s)C_2(s))e^{-Ls}Y_p(s). \quad (19)$$

Substituting (17) and (18) in (16) and rearranging:

$$Y_p(s) - (e^{-L_m s} - 1)(G_{m1}(s)C_1(s) + G_{m2}C_2(s))Y_p(s) = Y(s) + (-e^{-L_m s} - 1)(G_{m1}(s)C_1(s) + G_{m2}C_2(s))R(s). \quad (20)$$

Considering perfect modeling, that is, that the models are identical to the process dynamics:

$$G_1(s) = G_{m1}(s), G_2(s) = G_{m2}(s).$$

If there is no model error, it can be rewritten. Substituting (19) into (20) and considering an exact model:

$$Y(s) = \frac{(G_1(s)C_1(s) + G_2(s)C_2(s))e^{-Ls}}{1 + (G_1(s)C_1(s) + G_2(s)C_2(s))}R(s). \quad (21)$$

Knowing that $C_1G_1 + C_2G_2$ is an equivalence equal to that presented by (9) one can rewrite (21) as:

$$Y(s) = \frac{G_{eq}(s)C_{eq}(s)e^{-Ls}}{1 + (G_{eq}(s)C_{eq}(s))}R(s), \quad (22)$$

where $G_{eq}(s)$ and $C_{eq}(s)$ are the plant and the equivalent controller, respectively, according to (9).

2.3 Smith's Predictor approach by PID

The equivalent controller of Smith's Predictor is given by:

$$C_e(s) = \frac{C(s)}{1 + C(s)[G_m(s) - P_m(s)]}. \quad (23)$$

Considering a first-order process with dead time, such that:

$$P_m(s) = \frac{K}{\tau s + 1} e^{-L_m s}, G_m(s) = \frac{K}{\tau s + 1}.$$

Assume further that the primary controller is a PI controller:

$$C(s) = \frac{K_c(T_i s + 1)}{T_i s},$$

making the choice of controller tuning by canceling the plant pole with the controller zero:

$$C_e(s) = \frac{\frac{K_c(\tau s+1)}{\tau s}}{1 + \frac{K_c K}{\tau s}(1 - e^{-L_m s})}, \quad (24)$$

rewriting (24):

$$C_e(s) = \frac{K_c(\tau s + 1)}{\tau s + K_c K(1 - e^{-L_m s})}, \quad (25)$$

by using a first-order Padé approximation for the delay, it is possible to rewrite the controller as a PID [7]. Given:

$$e^{-L_m s} = \frac{1 - \frac{L_m s}{2}}{1 + \frac{L_m s}{2}}, \quad (26)$$

Substituting (26) in (25):

$$C_e(s) = \frac{K_c(\tau s + 1)}{\tau s + K_c K \left(1 - \frac{1 - \frac{L_m s}{2}}{1 + \frac{L_m s}{2}}\right)}, \quad (27)$$

Reordering (27):

$$C_e(s) = \frac{K_c(\tau s + 1)(1 + 0.5 L_m s)}{\tau s \left(1 + \frac{0.5 L_m s}{1 + \frac{K_c K L_m}{\tau}}\right)\left(1 + \frac{K_c K L}{\tau}\right)}, \quad (28)$$

Note that in equation (28) the equivalent controller has a gain, two zeros, and two poles, having the same dimension as a filtered PID, so by similarity, it can be approximated as:

$$C_e(s) = \frac{K_{ceq}(T_i s + 1)(T_D s + 1)}{T_i s(\gamma T_D s + 1)}, \quad (29)$$

where:

$$K_{ceq} = \frac{\tau}{(L_m + \tau_c)K}, \ \gamma = \frac{1}{1 + \frac{L_m}{\tau_c}},$$

$$T_i = \tau, \ T_D = 0.5 L_m.$$

With:

$$\tau_c = \frac{\tau}{K_c K},$$

being the desired closed-loop time constant.

2.4 Novel Controller

Considering the result from the previous section which used an SP whose primary controller was a PI [10], added the filtered derivative action, obtains the equivalent controller shown in:

$$C_{eq}(s) = \frac{K_c(T_i' s + 1)(T_D s + 1)}{T_i' s(\gamma T_D s + 1)}, \quad (30)$$

Using the equivalence of the SISO system to the SITO system presented in (8):

$$Y(s) = K_c \left(1 + \frac{\alpha}{T_i s}\right) \left(\frac{T_D s + 1}{\gamma T_D s + 1}\right) \frac{K_1}{\tau s + 1} e^{-L_m s} E(s), \quad (31)$$

is equivalent to:

$$Y(s) = \left(K_c \frac{K_1}{\tau s + 1} + \frac{K_c}{T_i s} \frac{K_2}{\tau s + 1}\right) \left(\frac{T_D s + 1}{\gamma T_D s + 1}\right) e^{-L_m s} E(s), \quad (32)$$

performing the distrubitive multiplication:

$$Y(s) = K_c \frac{T_D s + 1}{\gamma T_D s + 1} \frac{K_1}{\tau s + 1} e^{-L_m s} E(s) + \frac{K_c}{T_i s} \frac{T_D s + 1}{\gamma T_D s + 1} \frac{K_2}{\tau s + 1} e^{-L_m s} E(s). \quad (33)$$

This result in the block diagram of presented in Fig. 4, where it can be seen that the proposed new structure

resembles Fatani's structure. For the proposed control structure it can be observed that in each channel a filtered derivative action has been added. Note that the derivative action is the same on both channels. Thus, the new control structure is composed of a proportional and filtered derivative (PD) controller, which will control the process with lower static gain, and an integral and filtered derivative (ID) controller that will control the process with higher static gain.

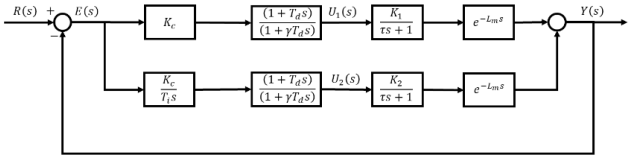

Fig. 4. The proposed control structure.

There for this is result can be synthesized in the tuning Table 1, where τ_c is the a tuning parameter, the desired close-loop time constant.

Table 1
Tuning table for proposed controller. K_c is the controller gain, K_1 is the static gain of G_1, K_2 is the static gain of G_2, τ stands for the process time constant, T_i represents the integral time, T_D is the derivative time, L_m is the dead time and γ is the filter of derivative action.

	K_c	T_i	T_D	γ
$C_1(s)$	$\frac{\tau}{K_1(L_m+\tau_c)}$	-	$\frac{L_m}{2}$	$\frac{1}{1+\frac{L_m}{\tau_c}}$
$C_2(s)$	$\frac{\tau}{K_1(L_m+\tau_c)}$	$\frac{K_2}{K_1}min(\tau, 4\tau_c)$	$\frac{L_m}{2}$	$\frac{1}{1+\frac{L_m}{\tau_c}}$

3 Simulation Results

For validation via computer simulation, consider the following process characteristics:

$$G_1(s) = \frac{K_1}{\tau_1 s + 1} e^{-Ls}, G_2(s) = \frac{K_2}{\tau_2 s + 1} e^{-Ls}.$$

Where:

$$K_1 = 14.7, K_2 = 18.67, L = 41, \tau_1 = \tau_2 = 189.$$

The model and control tuning are taking from Balbinot's study [10] which was done using the SIMC (Simple Internal Model Control) [12] technique. In the study the following gains are adopted for a temperature control system:

$$\tau_c = 65, \;\; K_c = 0.1978, \;\; T_i = 240.0429.$$

This means that the deployed controllers were:

$$C_1(s) = 0.1978, \;\; C_2(s) = \frac{0.1978}{240.0429s}.$$

On the other hand, the novel controller was designed according to Table 1 for $\tau_c = 65$:

$$K_c = 0.1213, \;\; T_i = 240.0429, \;\; T_D = 20.5, \;\; \gamma = 0.6132.$$

This leads to the following controllers:

$$C_1(s) = 0.1978 \left(\frac{20.5s + 1}{12.570s + 1} \right),$$
$$C_2(s) = \left(\frac{0.1213}{240.0429s} \right) \left(\frac{20.5s + 1}{12.570s + 1} \right).$$

Fig. 5 presents the simulation diagram for the Balbinot [10] controller, while Fig. 6 presents the simulation diagram for the approximated controller.

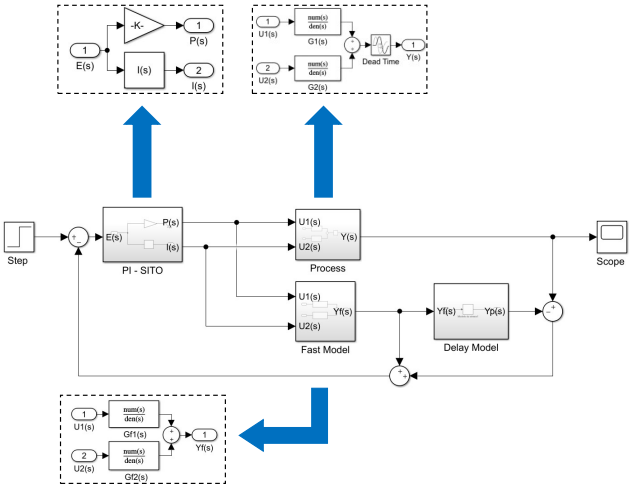

Fig. 5. Block diagram for simulation of the Balbinot [10].

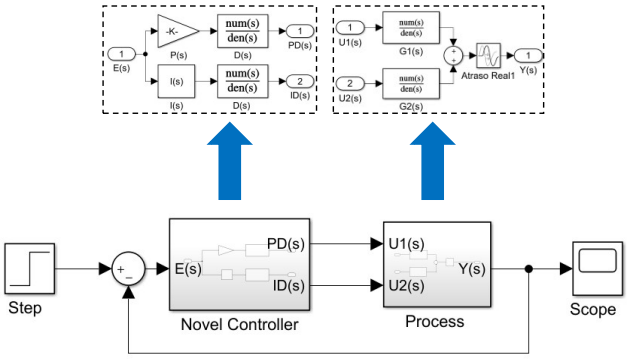

Fig. 6. Block diagram for simulation of the approximated controller.

Fig. 7 shows the system response for a series of variations in the setpoints, being 37 ºC, 39 ºC, 37 ºC, and finally 38 ºC. An ambient temperature was considered, this ambient temperature being 36 ºC. It is possible to

observe that the process output with the proposed controller performs similarly to the original controller proposed by Balbinot [10]. Thus, even though the proposed controller is an approximation, it can present similar characteristics to the original controller.

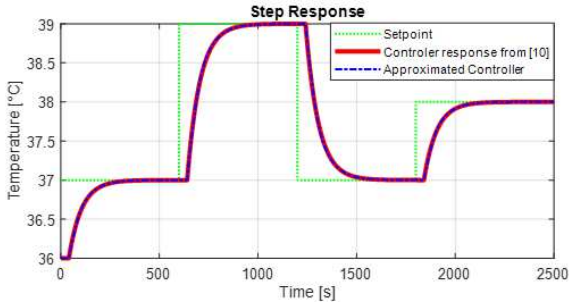

Fig. 7. Step response. Dotted green: setpoint. Solid red: original controller. Dashed blue: approximate controller.

In a more detailed graphical analysis consider Fig. 8 which demonstrates the process response for the initial 500 seconds, when the first step occurs. It can be seen that at the initial time the proposed controller has a slower behavior, subsequently, the proposed controller shows a faster behavior in the transient period. However, in the permanent regime, both controllers present the same behavior as seen in Fig. 7. This difference in the transient regime is due to the Padé approximation for the dead time.

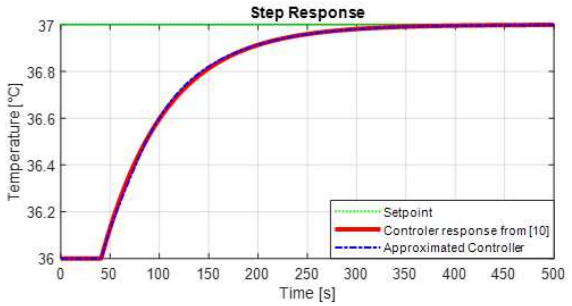

Fig. 8. Step response for the initial 500 seconds. Dotted green: setpoint. Solid red: original controller. Dashed blue: approximate controller.

In Fig.9 are shown the control signals for the two control strategies. Note that the integral action in both strategies presents the same behavior, however, when analyzing the proportional action it is possible to notice a difference in the signal. For the original controller, the proportional action presents a smooth slope because of the error prediction through Smith's Predictor. For the approximate controller, it is noticeable that the proportional action does not present this smooth slope, it starts to attenuate and after a while, it has a peak and then attenuates again. This occurs due to the derivative action.

To understand the effect of adding the derivative action, thus transforming $C_1(s)$, which was a P controller

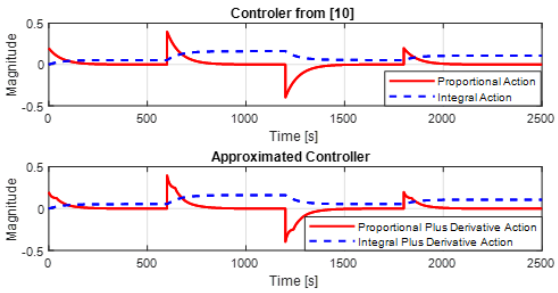

Fig. 9. From top to bottom. Top: Control signal from the original controller. In solid red: proportional action. Dashed blue: integral action. Bottom: Signal of the proposed controller. Solid red: proportional plus derivative action. Dashed blue: integral plus derivative action.

in [10], into a PD controller, a comparison of the system behavior with and without the derivative action was performed.

The derivative action causes the proportional action to attenuate in the dead time. Consider that if there were no derivative action the signal from the proportional action would remain constant until the system begins to react to the stimulus, that is, after the 41 seconds of dead time have passed. However, as the derivative action is present it presents a slope, which is measured in time would be 41 seconds, the same dead time identified. That said, the filtered derivative action is close to an estimate of prediction that the system will react, causing the control action to decrease. In Fig. 10 is shown graphically this comparison, of a pure proportional action and a proportional action plus the derivative.

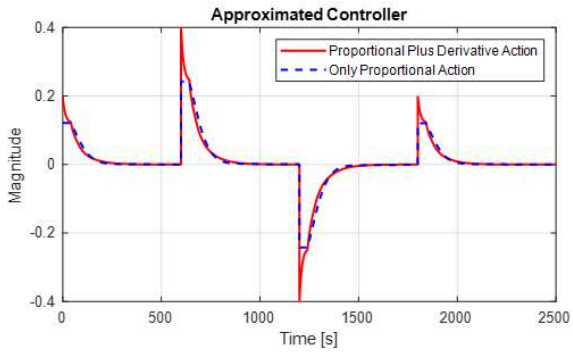

Fig. 10. Signal of the proposed controller. Solid red: proportional plus derivative action. Dashed blue: only proportional action.

Concerning computational cost, the new controller uses less memory space as it requires less value storage. For cost illustration, consider d the number of samples that should be delayed. For the [10] controller it was necessary to have $5 + 2d$ stored values and for the new controller, it is necessary to have a fixed value for the storage of values, being this quantity 5.

For a quantitative analysis of the control strategies, it

was decided to evaluate the error in each strategy, thus considering the order IAE (Intregal Absolute Error), ISE (Integral Square Error), ITAE (Integral Time from Absolute Error) and ITSE (Integral TIme from Square Error), according to the Table 2. It can be seen that for all metrics the performance of the approximate controller was very close to the original controller, any difference between the values are only marginal.

Table 2
Performance criteria by error for the first step (0s to 500s).

	Original Controller	Approximated Controller
IAE	105.90	105.90
ISE	73.50	73.89
ITAE	7699.00	7691.00
ITSE	3229.00	3235.00

For a performance analysis between the controller proposed by [10] and the new proposed strategy, consider Fig. 11 which presents a simulation case where no model error (nominal case) however the controller tuning is varied. As expected, the new strategy has similar performance for more conservative tuning but its performance degrades as the tuning becomes more aggressive. This behavior is explained because the Padé approximation is a faithful approximation to the dead time at low frequencies.

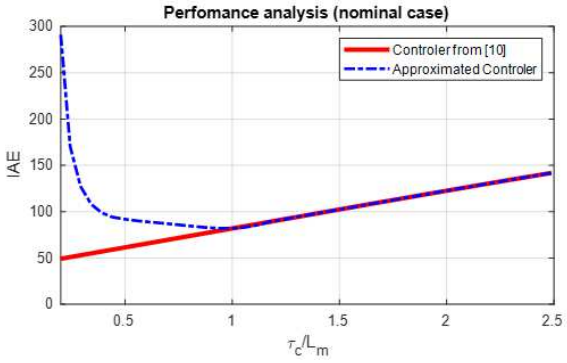

Fig. 11. Comparative analysis between controller from [10] and the novel controller for the nominal case: considering no modelling error. Solid red: Controller from [10]. Dashed blue: Approximate controller.

Consider now the case where there is a 5% error in the static gain, the time constant, and the dead time. Fig. 12 demonstrates the performance analysis for this scenario. It can be seen that for conservative tuning the performance of both controllers is still quite similar, however, the presence of modeling error causes the performance of both to diverge earlier. Again, this is expected, since the higher the modeling error the more conservative the tuning should be to ensure robustness.

This implies the approximate controller has limited applicability to cases where it is needed to significantly speed up the closed-loop response of the system. It is also worth noting that both controllers suffer from the same limitations as Smith's Predictor SISO concerning disturbance rejection.

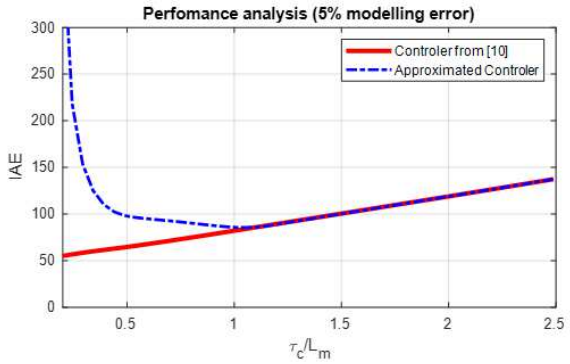

Fig. 12. Comparative analysis between controller from [10] and the novel controller for the non-nominal case: considering 5% modelling error. Solid red: Controller from [10]. Dashed blue: Approximate controller.

4 Conclusion

In the present paper, control proposals for a class of TISO systems with dead time were discussed. It was pointed out that in the literature there is a promising alternative to the split-range in such systems but with a gap regarding the presence of mean dead time. A way to solve this problem was recently presented in the literature, however, due to the need to use an explicit prediction model such a strategy presents a computational cost that scales with the transport delay. Therefore, we presented a low-frequency approximation for this controller that presents similar performance with a constant computational cost, structure based on two PID blocks, and being able to be implemented in less flexible hardware.

Given the results obtained through computer simulation, it is possible to confirm that the proposed controller has similar characteristics of the controller presented by [10]. It was possible to make a comparison with the performance criteria through the error of the controllers. From the comparison, it can be seen that the proposed controller presents a similar performance. When factoring in the advantage of the newer controller being simpler to implement and requiring less data storage, it poses itself as a promising alternative.

Also through this study, it was possible to define a tuning table for the new controller. This will facilitate the implementation of such controller, and the tuning of this control depends on the desired closed-loop time constant. It makes this new control approach as one viable strategy for implementation in SITO processes with dead time especially in less flexible hardware.

For future works, it is intended to systematize a way of choosing the generic time constant when the two identified time constants are different. Additionally, the proposal will be implemented in a practical experiment for physical validation. After validation in the practical experiment, a comparison with other consolidated control techniques in the literature, for example, split-range, will be performed. In addition, the goal is to generalize the characteristics of this controller to other types of control strategies for TISO systems. In particular, a valve position control scheme will be explored for the addition of a dead-time compensation capability.

Acknowledgements

The authors would like to thank IFRO - Campus Calama, CNPq, and CAPES for funding through the public notice Nº 5/2021/REIT - PROPESP/IFRO.

References

[1] G. C. Nunes, J. L. de Medeiros, and O. d. Q. F. Araújo, *Modelagem e controle da produção de petróleo: aplicações em Matlab*. Editora Blucher, 2010.

[2] D. P. Eckman, "Principles of industrial process control," 1945.

[3] A. Reyes-Lúa, C. Zotică, K. Forsman, and S. Skogestad, "Systematic design of split range controllers," *IFAC-PapersOnLine*, vol. 52, no. 1, pp. 898–903, 2019.

[4] A. Reyes-Lúa and S. Skogestad, "Systematic design of active constraint switching using classical advanced control structures," *Industrial & Engineering Chemistry Research*, vol. 59, no. 6, pp. 2229–2241, 2019.

[5] A. Reyes-Lúa and S. Skogestad, "Multi-input single-output control for extending the operating range: Generalized split range control using the baton strategy," *Journal of Process Control*, vol. 91, pp. 1–11, 2020.

[6] S. T. Fatani, R. S. Patwardhan, and M. A. L. Andreu, "A novel single-input two-output (sito) strategy for split range control," in *2017 6th International Symposium on Advanced Control of Industrial Processes (AdCONIP)*, pp. 348–353, IEEE, 2017.

[7] J. E. Normey-Rico, *Control of dead-time processes*. Springer Science & Business Media, 2007.

[8] L. R. da Silva, R. C. C. Flesch, and J. E. Normey-Rico, "Controlling industrial dead-time systems: When to use a PID or an advanced controller," *ISA transactions*, vol. 99, pp. 339–350, 2020.

[9] O. J. Smith, "A controller to overcome dead time," *ISA J.*, vol. 6, pp. 28–33, 1959.

[10] G. C. Balbinot, D. S. Sales, S. P. N. Santos, and J. D. F. d. O. Luna, "Uma estratégia de controle single-input two-output (sito) com capacidade de compensação de tempo morto," *Anais do 15o Simpósio Brasileiro de Automação Inteligente*, 2021.

[11] L. R. da Silva, R. C. C. Flesch, and J. E. Normey-Rico, "Controlling industrial dead-time systems: When to use a pid or an advanced controller," *ISA transactions*, vol. 99, pp. 339–350, 2020.

[12] S. Skogestad, "Simple analytic rules for model reduction and PID controller tuning," *Journal of Process Control*, vol. 13, no. 4, pp. 291–309, 2003.