ISSN-e: 1856-9811

OPTIMAL TUNING OF PID CONTROLLERS USING MATLAB/SIMULINK

PEDRO TEPPA - GARRAN[1]
pteppa@unimet.edu.ve
Universidad Metropolitana de Caracas (Venezuela)

FABIAN ARZOLA[2]
a.fabian@unimet.edu.ve
Universidad Metropolitana de Caracas (Venezuela)

ELIAS ELYAS[3]
elias.elyas@unimet.edu.ve
University of South Florida (USA)

SUMMARY

A PID controller tuning method based on an optimal design is developed using the Matlab/Simulink platform. This allows, by using the powerful tools provided by Matlab/Simulink, to transform many optimal control problems into a conventional control problem. To then perform the optimal tuning of the parameters of a PID controller, in a simple way, through the numerical methods provided by Matlab. These methods, while not providing elegant analytical solutions, can be extremely powerful in providing practical solutions to control problems. The versatility of Matlab's numerical algorithms allows consideration of more complex processes, actuator constraints, sensor noise and disturbances. In short, a much more realistic problem.

Keywords: PID, optimal control, numerical optimization, Matlab, Simulink, Simulink.

1 Full Professor, Department of Project and Systems Management, UNIMET. Electrical Engineer UNIMET, MSc Electronic Engineering and MSc Mathematics, Universidad Simón Bolívar, PhD Control Systems, Université Paul Sabatier, France, Postdoctorate Control Systems, LAAS du CNRS, France.

2 Electrical Engineer, UNIMET. Project Engineer, Inelectra.

3 Electrical Engineer, UNIMET.

ABSTRACT

A tuning method is developed based on the optimal design of PID controllers using the Matlab/Simulink platform. With the powerful tools provided in Matlab/Simulink, many optimal control problems can be converted into conventional optimization problems and then can be easily solved. These methods, while not providing elegant analytical solutions, can be extremely powerful in providing practical solutions to control problems. The versatility of Matlab's numerical algorithms allows considering more complex processes, actuator constraints, sensor noise, and disturbances. In short, a much more realistic problem.

Key words: PID, Optimal Control, Numerical Optimization, Matlab, Simulink.

RECEIVED: 09-03-2021 ACCEPTED: 11-05-2021 PUBLISHED: 15-12-2021

How to quote: Teppa Garran *et al*. (2021). The brain imprint of attention and its fight against the distracting effect of pain. *Anales*, 41, 15-32.
https://doi.org/10.58479/acbfn.2021.20

ISSN-e: 2244-8276

CONTENIDO

1. INTRODUCTION

The Proportional - Integral - Derivative (PID) controller is one of the first control strategies used in practice, its initial implementation was through pneumatic devices, followed by the use of vacuum tubes and analog electronic circuits, until the current digital implementation using microprocessors (Bennett, 1993). Today, despite significant advances in control theory, it is the most widely used strategy in industrial process control. According to Åström and Hagglund (1995) more than 90% of industrial control systems contain some form of PID control. A more recent study conducted by Desbourough and Miller (2002) considered about 11,000 controllers in refineries, chemical industries and paper mills, concluding that about 97% of the controllers had a PID structure. This ubiquity of the PID controller is not only due to its simple configuration, easy to understand conceptually and to tune manually, but also to the fact that the design algorithms provide a satisfactory performance of the controlled system in many real applications. The PID controller parameters must be tuned to meet the design requirements and to that end, several rules have been formulated, some examples appearing in the literature and several of them commonly used in practice are: Ziegler and Nichols' rule (1942), Chien, Hrones and Reswick's rule (1952), Cohen and Coon's rule (1953), the complementary rule (Mantz and Taconi, 1989), the somewhat overpeak and non-overpeak rules (Seborg, Edgar and Mellichamp, 1989), the modified Ziegler and Nichols' rule (Hang, Åström and Ho, 1991), the integral of the quadratic error times time integral rule (Zhuang and Atherton, 1993), the absolute error integral rule (Pessen, 1994), the Wang, Juang and Chan (1995) formula, and active disturbance rejection (Teppa-Garrán and García, 2013, 2017), among others. These rules, many of them easy to implement, provide methods for tuning the PID controller parameters. However, the diversity of existing rules and methods is an indication that each has some kind of limitation or disadvantage. This is mainly because they use little information of the dynamics of the controlled process, so that in many situations, the closed-loop response is not satisfactory. In fact, the tuning of PID controllers in industrial environments is complicated due to several factors, among them can be mentioned (Diu and Daley, 2001): nonlinearities produced by constraints on the actuators, aging and wear of the process, uncertainties caused by unmodeled dynamics, measurement noise and external disturbances and deterioration in performance when operating under load conditions due to the variable nature of the latter.

This paper develops a method for tuning the parameters of PID controllers based on an optimal design using the Matlab/Simulink platform. This allows, by using the powerful tools provided by Matlab/Simulink, to transform many optimal control problems into a conventional control problem. To then perform the optimal tuning of the parameters of a PID controller, in a simple way, through numerical methods. These methods, while not providing elegant analytical solutions, can be extremely powerful in providing practical solutions to control problems.

The versatility of Matlab's numerical algorithms makes it possible to consider more complex processes, actuator constraints, sensor noise and disturbances. In short, a much more realistic problem. The article is organized as follows: section 2 describes the mathematical model of the PID controller, section 3 considers the developed method, i.e., the optimal tuning of the PID parameters using the Matlab/Simulink computational platform. Section 4 considers two examples of application of the methodology and finally in section 5 the conclusions of the research are presented.

2. PID CONTROLLER

The typical structure of a PID controller is shown in Fig. 1, the error signal is used to generate the proportional, integral and derivative actions, which are then weighted and summed to form the control signal. $e(t)$ is used to generate the proportional, integral and derivative actions, which are then weighted and summed to form the control signal. $u(t)$. A mathematical model of the PID controller is

$$u(t) = K_p\left[e(t) + \frac{1}{T_i}\int_0^t e(\tau)dt + T_d\frac{de(t)}{dt}\right] \tag{1}$$

The error signal is defined as $e(t) = r(t) - y(t)$, with $r(t)$ the reference input signal. When the reference signal changes discontinuously, the derivative term is not used in the pure form of (1) in order to avoid abrupt variations in the control signal, a situation known as the *derivative kick* phenomenon (Atherton and Majhi, 1999), but is cascaded with a first order low-pass filter (parameter N). In this way, the Laplace transform of (1) results in

$$U(s) = K_p\left[1 + \frac{1}{T_i s} + \frac{sT_d}{1 + s\frac{T_d}{N}}\right]E(s) \tag{2}$$

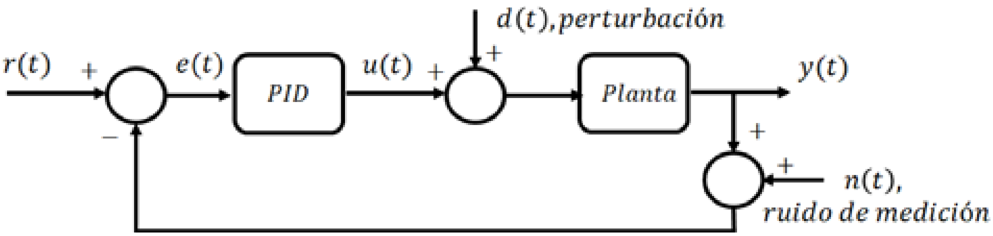


Figure 1. *Typical structure of a PID controller.*

3. OPTIMAL PID CONTROLLER DESIGN

A conventional unconstrained optimization problem has the form

$$\min_{x} \mathcal{J}(\boldsymbol{x}) \tag{3}$$

where $x = [x_1\ x_2\ \dots\ x_n\ ]^T$. The interpretation of (3) is: determine the vector x such that the objective function $J(x)$ is minimal. The Matlab function *fminsearch* allows to solve the optimization problem (3) using the simplex method of Nelder and Mead (1965). The programming code of the *fminsearch* function used in Matlab is obtained from (Lagarias et al, 1998). The syntax of the instruction is

$$x = \text{fminsearch(Fun, } x_0) \tag{4}$$

Where Fun corresponds to the objective function and the variable x_0 is the initial vector of parameters. The optimal control problem considered in this work, associated to the block diagram in Fig. 1, is as follows

$$\mathcal{J}(e,u) = \int_0^{t_f} \mathcal{L}(e(t),u(t))dt \ \ sujeto\ a\ \{y = \mathcal{P}(u+d), u = C(\boldsymbol{p})e \tag{5}$$

Where the integral $J(e,u) = \int_0^{t_f} \ \ L(e(t),u(t))dt$ corresponds to the objective function or performance index of the system and is such that the function is continuous and derivable in its arguments. $L: R\ x\ R \rightarrow R^+$ is continuous and derivable in its arguments. The operator P is the mathematical model describing the plant, which receives as inputs: the control signal and the disturbance signal. $u(t)$ and the disturbance signal $d(t)$. On the other hand, the operator $C(p)$ represents the PID controller, with $p \in R^4$ the vector of controller parameters, that is, $p = [K_p, T_i, T_d, N]^T$. In many applications it is satisfactory to choose $N = 10$ (Åström and Hagglund, 1995). The usual choices for the objective function $J(e,u)$ are listed in Table 1 (Kirk, 2004). The first six objective functions consider the tracking error of the feedback control loop. $e(t) = r(t) - y(t)$ of the feedback control loop and the last two use the control signal. $u(t)$. The ISE and IAE criteria minimize the quadratic error and the absolute value of the error. ITSE, ITAE, ISTAE and ISTSE incorporate weighting through the time variable in order to penalize errors that occur later in the time response. ISTAE and ISTSE provide a higher penalty than the ITSE and ITAE criteria.

TABLE 1.
Typical target functions

Criteria	Target function
ISE (*Integral of squared error*)	$J_{ISE}(e) = \int_0^\infty e^2(t)dt$
IAE (*Integral of absolute error*)	$J_{IAE}(e) = \int_0^\infty \|e(t)\|dt$
ITAE (*Integral of time-multiplied absolute value of error*)	$J_{ITAE}(e) = \int_0^\infty t\|e(t)\|dt$
ITSE (*Integral of time-multiplied squared error*)	$J_{ITSE}(e) = \int_0^\infty te^2(t)dt$
ISTAE (*Integral of squared time-multiplied absolute value of error*)	$J_{ISTAE}(e) = \int_0^\infty t^2\|e(t)\|dt$
ISTSE (*Integral of squared time-multiplied squared error*)	$J_{ITSE}(e) = \int_0^\infty t^2e^2(t)dt$
Minimum energy	$J(u) = \int_0^\infty u^2(t)dt$
Minimum fuel	$J(u) = \int_0^\infty \|u(t)\|dt$

To perform the optimal tuning of the PID controller parameters the first step is to represent the optimal control problem (5), associated to the controlled system of Fig. 1, by means of a simulink diagram. For example, if it is required to minimize the objective function ITAE, a simulink diagram of the controlled system, which we will call '*Model*', is illustrated in Fig. 2 (The following has been selected $N = 10$).

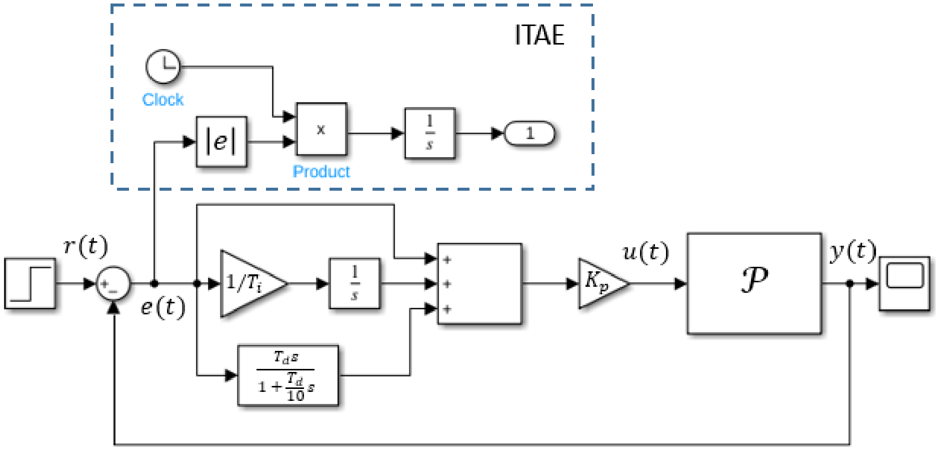


Figure 2. *Simulink diagram of the controlled system (Model).*

To continue with the optimization process it is required to program a function, this corresponds to a Matlab script that will allow to determine the optimal values (for a selected objective function) of the PID controller parameters. For this purpose, a *script* is created and saved in the Matlab files and will be implemented later. It is important to underline that this *script* must be stored with the same name as the function to ensure its correct execution. The name assigned was '*optimize*'. The function code is illustrated in Fig. 3. The function receives as argument a vector $x$whose components are the parameters of the PID controller (K_p,T_i,T_d) y t_f corresponds to the simulation time.

```
function J= optimiza(x)
assignin('base','Kp',x(1));
assignin('base','Ti',x(2));
assignin('base','Td',x(3));
[t,x,y]=sim('Modelo',tf);
J=y(end);
end
```

Figure 3. *Function 'optimize'.*

Finally, to solve the optimization problem, the following instructions are executed in the Matlab work window:

" $x_0 = [Kp_0; Ti_0; Td_0]$; <enter>

(The vector $[Kp_0; Ti_0; Td_0]$ corresponds to the initial values of the PID controller parameters.)

“ x =fminsearch (‘optimize’, x_0) <enter>

In the vector $x = [x(1)\ x(2)\ x(3)\ ]^T = \left[K_p\ T_i\ T_d\ \right]^T$ the optimal parameters of the PID controller are stored, which result after simulating the simulink ‘*Model’* of the controlled system of Fig. 2 for a time t_f using the ‘*optimize*’ function of Fig. 3.

An advantage of the method is that it allows a more realistic design incorporating, for example, constraints on actuators, measurement noise, disturbances, more complete models of the plant, etc. Pamong others. This situation is illustrated in the diagram in Fig. 4, which shows some of the discontinuities that can be placed between the PID controller and the plant, depending on the nature of the constraints on the actuator.

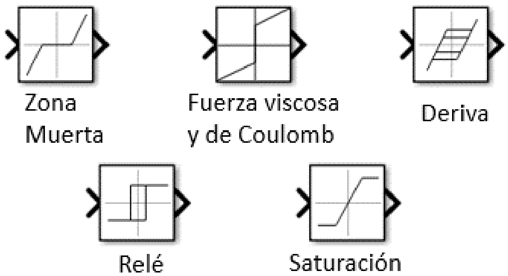


Figure 4. *Some of the actuator constraints that can be considered in the optimal PID controller design.*

Regarding the plant model P typical representations can be considered, such as the transfer function or in state variables in Fig. 5, but also more complex descriptions such as the nonlinear one in Fig. 6.

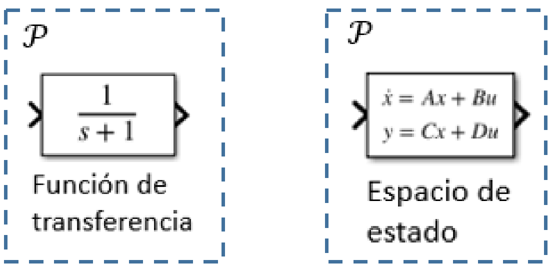


Figure 5. *Representation of* P *by means of a transfer function or in state variables.*

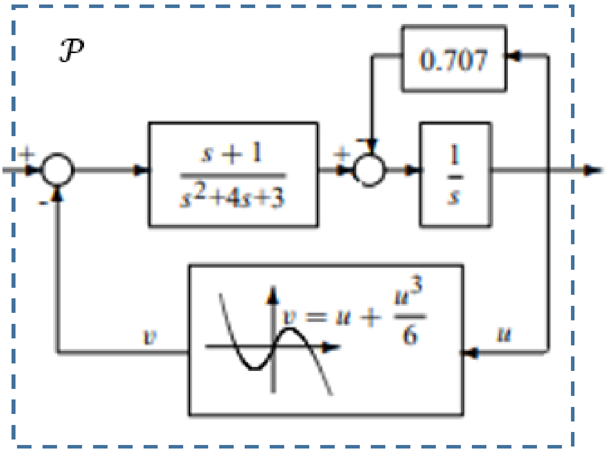


Figure 6. *Non-linear representation of* P.

4. EXAMPLES

Two examples are considered: the first one consists of the design of a PID controller for a linear system when the actuator has amplitude restrictions. Initially, the performance of a PID controller tuned through the popular and widely used Ziegler and Nichols (1942) rule under ideal conditions (no constraint on the actuator) is shown. Subsequently, it is shown how the controller performance degrades when the practical amplitude constraint is incorporated in the actuator and how the proposed method achieves an excellent solution. Immediately, in example 2, a more demanding design, the optimal tuning of a PI controller for a time-varying system, is discussed.

4.1 EXAMPLE 1

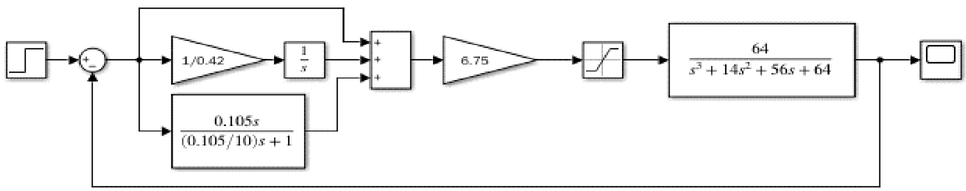
The plant is P is modeled by the transfer function

$$G_p(s) = \frac{64}{s^3 + 14s^2 + 56s + 64} \tag{6}$$

Applying the rules of Ziegler and Nichols (1942) results in the following PID controller parameter values

$$K_p = 6.75, T_i = 0.42, T_d = 0.105 \tag{7}$$

The controlled system is shown in Fig. 7. The closed-loop response for a unit step reference is shown in Fig. 8, considering two situations. In the first one, ideal conditions are considered, i.e. the saturation block of Fig. 7 is not included, while in the second one, it is included by an actuator amplitude restriction between $[-2.5, 2.5]$.



Controlled system of Example 1 considering that there is and that there is no amplitude saturation in the actuator.

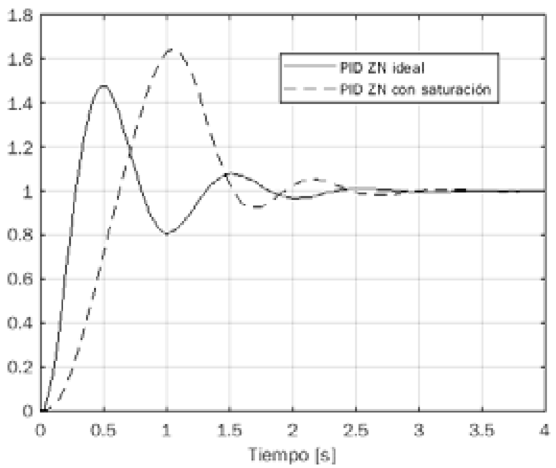


Figure 8. *Closed-loop response of Example 1 considering that there is and that there is no amplitude saturation in the actuator.*

As can be seen in Fig. 8, the response, which is already initially not very satisfactory due to oscillations and high overshoot, is further degraded by including the actuator constraint.

The following is the optimal tuning of the PID controller using the proposed methodology, for which the simulink diagram of Fig. 9 is initially prepared, where the ITAE criterion has been selected as the objective function and then the '*optimize*' function of Fig. 3 is implemented. When the *fminsearch* instruction is executed in the Matlab work window, using as initial values of the PID controller parameters those previously obtained by the Ziegler and Nichols rule, the result is

$$K_p = 7.50, T_i = 3.08, T_d = 0.16 \quad (8)$$

It should be noted that other initial values can be used, for example all zero, however, convergence to the optimum may take longer. The response of the system with amplitude constraints on the actuator previously obtained by using the PID controller tuned by Ziegler and Nichols and the response resulting from applying the proposed numerical optimization method are shown simultaneously in Fig. 10. Finally, Fig. 11 illustrates the evolution of the control signal within the set limits of the actuator.

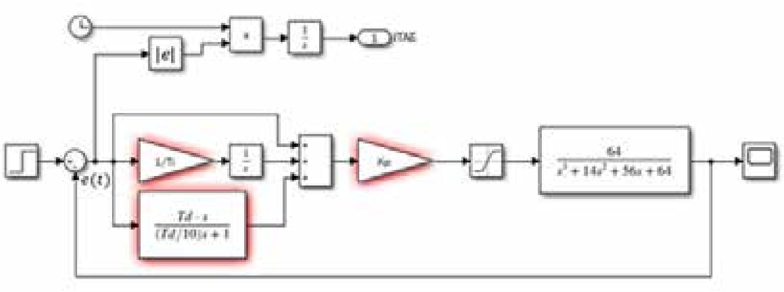


Figure 9. *Elaboration of the 'Model' used in the 'optimize' function of Fig. 3.*

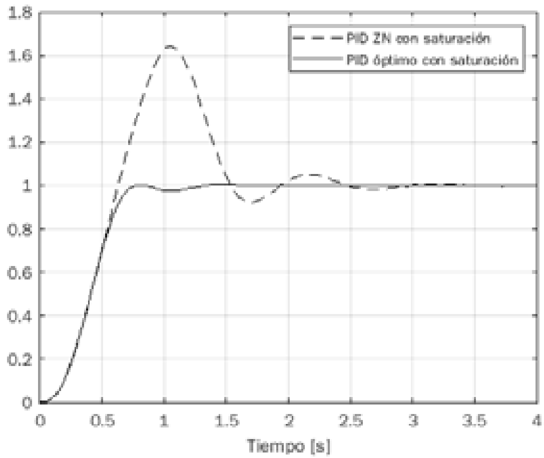


Figure 10. *Response of the controlled system with actuator constraints using both the Ziegler and Nichols rule and the proposed optimization method.*

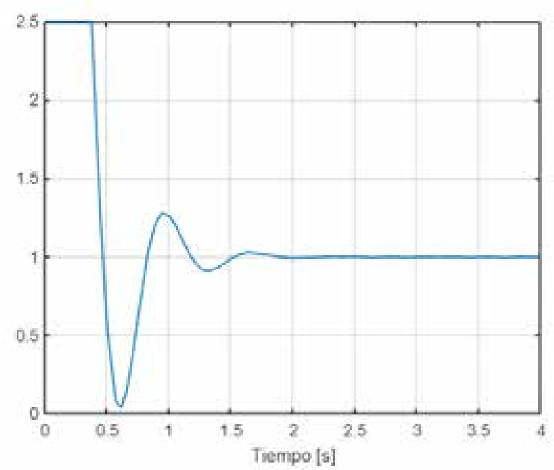


Figure 11. *Control signal of example 1.*

4.2 EXAMPLE 2

The plant P is modeled by the differential equation

$$\ddot{y}(t) + e^{-0.2t}\dot{y}(t) + e^{-5t}sin(2t+6)y(t) = u(t) \quad (9)$$

Selecting as state variables $x_1(t) = y(t), x_2(t) = \dot{y}(t)$ the representation in state variables of the plant is obtained as follows P following

$$\{\dot{x}_1(t) = x_2(t)\ \dot{x}_2(t) = -e^{-5t}sin(2t+6)x_1(t) - e^{-0.2t}x_2(t) + u(t) \quad (10)$$

The simulink diagram of the '*Model*' to be used for the optimal tuning of a PI controller using the ITAE criterion, constraints on the actuator between [-5, 5] and initial values of the controller is illustrated in Fig. 12. $K_p = 200, T_i = 20$ The closed-loop response of the controlled system to a unit step reference using the optimum values determined by the method ($K_p = 108.26, T_i = 423.57)$ is shown in Fig. 13. The control signal, always within the actuator limits, is illustrated in Fig. 14.

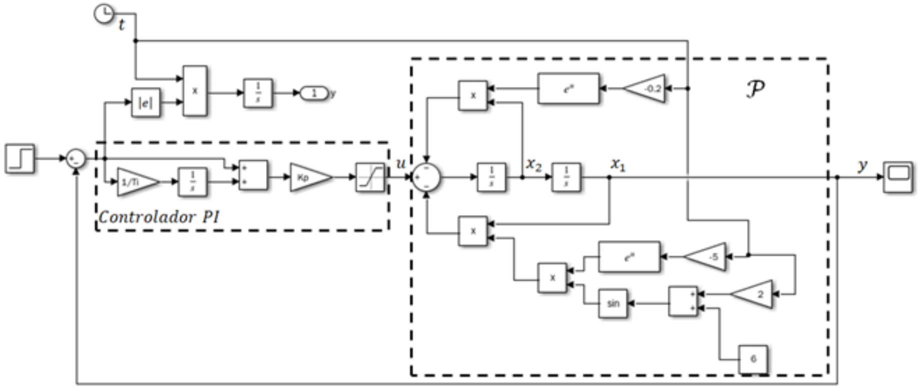


Figure 12. *Simulink diagram of the 'Model' of example 2.*

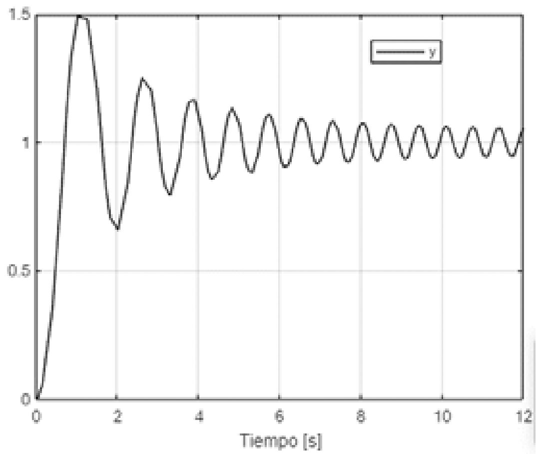


Figure 13. *Controlled output of example 2.*

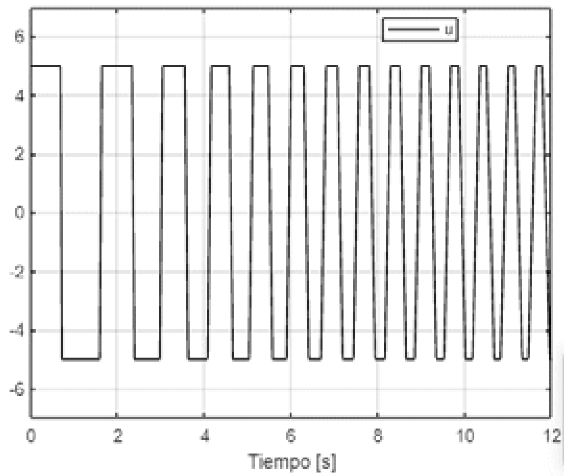


Figure 14. *Control signal of example 2.*

5. CONCLUSIONS

A method of optimal tuning of the parameters of a PID controller using the Matlab/Simulink platform is proposed. The versatility of Matlab numerical algorithms allows considering more realistic control problems.

In conventional control theory, the *hardware implementation of* the PID controller is not often treated, for that reason very high amplitude signals are accepted. However, in real-time control, the signals can not be very large in order to avoid damage to equipment. In example 1, although the graph is not included, the control signal without restrictions, reaches an initial value of 80, which is very high. The same could cause problems in the *hardware* by a bad design and saturate the actuator causing a nonlinear operation. The proposed method allows to include the saturation in the actuator from the initial stage by adding a simple saturation block in the simulink diagram. In example 2, a highly complex plant is considered, a time-variant system, where practical and popular PID controller tuning rules are not satisfactory. These two examples are just a sample of the versatility and potential of the proposed methodology in the design of PID controllers for more realistic control problems.

REFERENCES

ÅSTRÖM, K. and HAGGLUND, T. (1995). *PID controllers: Theory, design and tuning*. NC: Instrument Society of America, Research Triangle Park.

ATHERTON, D.P. and MAJHI, S. (1999). "Limitations of PID controllers". *Proc. of the 1999 American Control Conference*, San Diego, USA.

BENNETT, S. (1993). "Development of the PID controllers". *IEEE Control System Magazine*, Vol. 13(2), pp. 58 - 65.

CHIEN, K., HRONES, J. and RESWICK, J. (1952). "On the automatic control of generalised passive systems". *Transactions of the ASME*, pp. 175 -185.

COHEN, G. and COON, G. (1953). "Theoretical considerations of retarded control". *Transactions of the ASME*, pp. 827 - 834.

DESBOUROUGH, L. and MILLER, R. (2002). "Increasing customer value of industrial control performance monitoring: Honeywell's experience," In *6th International Conference on Chemical Process Control*, AIChE Symposium Series, Vol. 98(326).

HANG, C., ÅSTRÖM, K. and HO, W. (1991). "Refinements of the Ziegler - Nichols tuning formula". *IEE Proceedings-D*, Vol. 138(2), pp. 111 - 118.

KIRK, D. (2004). *Optimal Control Theory*, New York: Dover Publications Inc.

LAGARIAS, J., REEDS, J., WRIGHT, M and WRIGHT, P. (1998). "Convergence properties of the Nelder - Mead simplex method in low dimensions". *SIAM Journal of Optimization*, Vol. 9(1), pp. 112 - 147.

LIU, G. and DALEY, S. (2001). "Optimal-tuning PID control for industrial systems". *Control Engineering Practice*, Vol. 9, pp. 1185 - 1194.

MANTZ, R. and TACONI, E. (1989). "Complementary rules to Ziegler-Nichols: rules for a regulating and tracking controller". *International Journal of Control*, Vol. 49(5), pp. 1465 - 1471.

NELDER, J. and MEAD, R. (1965). "A simplex method for function minimization", *Computer Journal*, Vol. 7, pp. 308 - 313.

PESSEN, D. (1994). "A new look at PID controller tuning". *Journal of Dynamic Systems, Measures and Control*, Vol. 116, pp. 53 - 557.

SEBORG, D., EDGAR, T. and MELLICHAMP, D. (1989). *Process Dynamics and Control*. New York: Wiley.

TEPPA-GARRAN, P. and GARCIA, G. (2013). "Optimal tuning of PI/PID/PID(n-1) controllers in active disturbance rejection control," *Journal of Control Engineering and Applied Informatics*, Vol. 15(4), pp. 26-36.

TEPPA-GARRAN, P. and GARCIA, G. (2017). "Design of an Optimal PID controller for a coupled tanks system employing ADRC," *IEEE Latin America Transactions*, Vol. 15(2), pp. 189-192.

WANG, F., JUANG, W. and CHANG, C. (1995). “Optimal tuning of PID controllers for single and cascade control loops”. *Chemical Engineering Communications*, Vol. 132, pp. 15 - 34.

ZHUANG, M. and ATHERTON, D. (1993). “Automatic tuning of optimum PID controllers”. *IEE Proceedings-D*, Vol. 140(3), pp. 216 - 224.

ZIEGLER, J. and NICHOLS, N. (1942). “Optimum settings for automatic controllers”. *Transactions of the ASME*, Vol. 64, pp. 759 -768.