10th IFAC Symposium Advances in Control Education
The International Federation of Automatic Control
August 28-30, 2013. Sheffield, UK

An Optimization Software Tool for Performance/Robustness Analysis and Tuning of PID Controllers

D.E. Fernández *, **V.M. Alfaro** *, **O. Arrieta** *, **R. Vilanova** **

* *Departamento de Automática, Escuela de Ingeniería Eléctrica, Universidad de Costa Rica, 11501-2060 San José, Costa Rica. corresponding authors: {Victor.Alfaro, Orlando.Arrieta}@ucr.ac.cr*
** *Departament de Telecomunicació i d'Enginyeria de Sistemes Escola d'Enginyeria, Universitat Autònoma de Barcelona 08193 Bellaterra, Barcelona, Spain. e-mail: Ramon.Vilanova@uab.cat*

Abstract: This paper presents an optimization and simulation software tool with a graphical user interface to analyze the trade-off between the performance and robustness of PID control systems. The process can be represented by a first- or second-order plus dead-time model and the controller is a PID-type controller with one or two degrees of freedom. From the information of the model, the PID is designed for optimal integrated absolute error (IAE) performance and also it is possible to tune the controller with three different levels of degraded IAE performance, in order to modify the robustness of the system. The tool allows to test different model/controller combinations and it is concluded that degrading the control system performance to increase its robustness has a higher impact over the controller gain than over other of its parameters. The software tool allows to study the relation between control system performance and robustness by undergraduate students of a control system course.

Keywords: PID controllers, two-degree-of-freedom controllers, performance/robustness trade-off.

1. INTRODUCTION

As it has been widely reported, the *proportional integral derivative* (PID) are the controllers most extensive used in the process industry, most of them actually being *proportional integral* (PI) controllers (Åström and Hägglund, 2001, 2006). Their success is mainly due to its simple structure that makes them easier to understand by the control engineers, than others most advanced control approaches.

In industrial process control applications, the set-point normally remains constant and good load-disturbance rejection is required, usually know as *regulatory control*. In addition due to process operation conditions, eventually the set-point may need to be changed and then a good response to this change is required, know as *servo control* operation. However, because these two demands can not be simultaneously satisfied with a one-degree-of-freedom (1DoF) controller, the use of a two-degree-of-freedom (2DoF) controller allows to tune the controller considering the control system robustness and the regulatory control performance, and use the extra parameter to improve the servo control behavior.

Moreover, robustness is an important attribute for control systems, because the design procedures are usually based on the use of low-order linear models identified at the closed loop operation point. Due to the non-linearity found in most of the industrial processes, it is necessary to consider the expected changes in the process characteristics assuming certain relative stability margins, or robustness requirements, for the control system. Therefore, the design of the closed-loop control system must take into account the system performance to load-disturbance and set-point changes and its robustness to variation of the controlled process characteristics, preserving the well known *trade-off* between all these variables.

With regard to the design and tuning of PID controllers, there are many methods that can be found in the literature over the last seventy years. In fact, since Ziegler and Nichols (1942) presented their PID tuning rules, a huge number of procedures have been developed, using different approaches to deal with a variety of control problems. In O'Dwyer (2003) a collection of tuning rules for PID controllers is presented, where can be seeing its abundance.

The great evolution that control systems have had in the last years, has brought the need to make a more precise control, taking into account the conflicting aspects that can be presented. Within these possibilities, the control system performance - as well as - the robustness must be considered as important attributes that every control loop has to consider.

The above statement makes that computational software became a fundamental tool to analyze the control system behavior through simulation. On this idea is where software with an interactive interface helps the teaching-learning process, bringing to the students the necessary

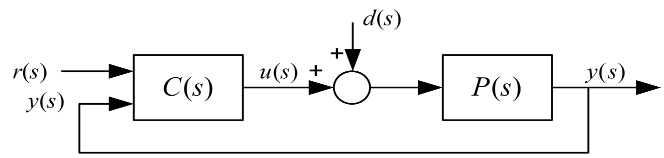


Figure 1. closed-loop control system

tools to study, understand, and properly design control systems. See the works by Garpinger et al. (2012); Guzmán et al. (2012), and Dormido et al. (2012).

Taking into account the spread use of PID controllers in industrial applications, it is very important that undergraduate control students, get the necessary skills to successfully face practical control problems.

In this paper, an optimization and simulation software tool is presented. This was developed using MATLAB® with a graphical user interface, that allows the students to optimize and simulate PID closed-loop control system, in order to evaluate its performance and robustness but more important, to analyze the existing trade-off between them.

From the controlled process model information, than can be of first- or second-order plus dead-time (FOPDT or SOPDT), the tool performs an optimization procedure to get the PID-type controller (1DoF or 2DoF) parameters for optimal performance and for three different user selected levels of degraded performance. This in order to increase the robustness of the system. In addition, the interface shows the system output to step changes in the set-point and disturbance inputs, as well as the Nyquist plot diagram. With this information the student is able to analyze the control system performance and robustness.

Considering the characteristics and the aim of the tool, it is called *ACDER-PID*, from the Spanish acronym for *Performance Robustness Trade-off Analysis in PID Control*. The tool is used by students of an entry level control systems course, contributing to speed-up their control systems knowledge.

The paper is organized as follows. Section 2 presents the control system configuration, the used metrics for performance and robustness and the controller optimization procedure, that are themselves the general framework for the tool. Section 3 describes the software interface and in Section 4 it is shown the use and the capabilities of the tool with some examples. The students feedback from the tool use experience is presented in Section 5 and the paper ends with some conclusions in Section 6.

2. PROBLEM FORMULATION AND FRAMEWORK

For the closed-loop control system shown in Fig. 1, $P(s)$ is the controlled process model and $C(s)$ the controller. In this system, $r(s)$ is the set-point, $u(s)$ the controller output, $d(s)$ the load-disturbance, and $y(s)$ the controlled variable.

2.1 Processes and Controllers

The controlled process will be represented by a low-order (first- or second-order) plus dead-time model given by the general transfer function

$$P(s) = \frac{K\mathrm{e}^{-Ls}}{(T_1 s + 1)(T_2 s + 1)}, \tag{1}$$

with a gain K, time constants T_1 and T_2, and a dead-time L. Process model parameters are $\theta_p = \{K, T_1, T_2, L\}$.

The controller is a one- or two-degree-of-freedom (1DoF, 2DoF) proportional integral (PI) or proportional integral derivative (PID) controller with an output signal given by:

$$u(s) = K_p \left\{e_p(s) + e_i(s) + e_d(s)\right\}, \tag{2}$$

with

$$e_p(s) = \beta r(s) - y(s), \tag{3}$$

$$e_i(s) = \frac{1}{T_i s}[r(s) - y(s)], \tag{4}$$

$$e_d(s) = -\frac{T_d s}{\alpha T_d s + 1} y(s), \tag{5}$$

where K_p, T_i, T_d, β, and α are the controller gain, integral time, derivative time, proportional set-point weight, and derivative filter constant ($\alpha = 0.1$), respectively. For 1DoF controllers the set-point weight $\beta = 1$. As shown in (5) the controller derivative mode is apply only to the feedback signal. Controller parameters are $\theta_c = \{K_p, T_i, T_d, \beta\}$.

Control system stability and regulatory control performance is affected only by K_p, T_i, and T_d and the servo control performance depends on these parameters but also on β.

2.2 Performance and Robustness

The main purpose of the tool is to allow an analysis of the control system performance/robustness trade-off and uses the methodology presented in Alfaro et al. (2010).

Performance: Performance will be optimized using the integrated absolute error (IAE) given by

$$J_e \doteq \int_0^\infty |e(t)|\,\mathrm{d}t = \int_0^\infty |r(t) - y(t)|\,\mathrm{d}t, \tag{6}$$

and computed for a load-disturbance step change, J_{ed}, and for a set-point step change, J_{er}.

For 1DoF PI and PID controllers two different cases (operating modes) are considered: regulatory control (load-disturbance rejection) optimal tuning and servo control (set-point tracking) optimal tuning. In a second step, achieved performance in the other operating mode is evaluated with the controller previously optimized.

For 2DoF PI and PID controllers in a first step, the performance is optimized for a load-disturbance change and then, β is used to obtain the best possible set-point response in the IAE-sense.

Robustness: The control system robustness is evaluated using the maximum sensitivity defined as

$$M_S \doteq \max_\omega |S(\mathrm{j}\omega)| = \max_\omega \frac{1}{|1 + C_y(\mathrm{j}\omega)P(\mathrm{j}\omega)|}. \tag{7}$$

It is widely accepted that the minimum robustness level to have a *robust control system* corresponds to $M_S = 2.0$. Then, control systems with $M_S > 2.0$ are “non-robust”. Robustness ranges from $M_S = 2.0$ (minimum) to $M_S = 1.4$ (high) being $M_S = 1.6$ considered as an *intermediate robustness level*.

2.3 Controller Optimization Procedure

The control system design is normally based on the use of a low-order linear model with parameters identified at the system normal operating point, to represent the nonlinear, and probably high-order, controlled process dynamics. Then, it is necessary to take into account the expected changes in process dynamics assuming certain relative stability or robustness level.

It is well documented that performance optimized control systems have very poor robustness. In the other hand, if the system is designed to have high robustness and if the performance is not evaluated, the designer does not have any idea of the cost of having such highly robust control system.

Performance optimized controllers: As a starting point the servo control (1DoF PI/PID) optimal controller parameters θ_c^{or} are obtained such that

$$J_{er}^o(\theta_p) \doteq J_{er}(\theta_c^{or}, \theta_p) = \min_{\theta_c} J_{er}(\theta_c, \theta_p), \tag{8}$$

or the regulatory control (1DoF PI/PID, 2DoF PI/PID) optimal controller parameters θ_c^{od} are obtained such that

$$J_{ed}^o(\theta_p) \doteq J_{ed}(\theta_c^{od}, \theta_p) = \min_{\theta_c} J_{ed}(\theta_c, \theta_p). \tag{9}$$

Degraded performance: In order to improve the control system robustness *performance degradation factors* F_p are defined as

$$F_{pd}(\theta_p) \doteq \frac{J_{ed}^o(\theta_p)}{J_{ed}(\theta_c, \theta_p)};\ F_{pd}(\theta_p) \le 1, \tag{10}$$

$$F_{pr}(\theta_p) \doteq \frac{J_{er}^o(\theta_p)}{J_{er}(\theta_c, \theta_p)};\ F_{pr}(\theta_p) \le 1. \tag{11}$$

$F_{pd/r} = 1$ corresponds to an optimal performance tuning.

For the degraded performance optimization the cost functional is defined as:

$$J_{F_p} \doteq J(F_p, \theta_c, \theta_p) = \left| \frac{J_e^o(\theta_c^o, \theta_p)}{J(\theta_c, \theta_p)} - F_p^t \right|, \tag{12}$$

where F_p^t is the selected performance degradation factor.

Controller optimization made use of MATLAB® optimization toolbox functions (The MathWorks Inc., 2011b) to obtain controller parameters for optimal performance (optimizing J_e) and for degraded performance (optimizing J_{F_p}).

3. TOOL USER INTERFACE

The software tool has a graphic user interface (GUI) programmed in MATLAB® (The MathWorks Inc., 2011a).

The main sections and use of the interface shown in Fig. 2 are (Fernández, 2012):

(1) Controller
To select the controller type and main operation mode from a list including: 1DoF PI or PID, servo or regulatory control operation, and 2DoF PI or PID.

(2) Controlled Process Model
To select the controlled process model: FOPDT or SOPDT.

(3) Model Parameters
To enter model parameters: (K, T, L) of the FOPDT model or (K, T_1, T_2, L) of the SOPDT model.

(4) Degraded Performance Factor
To enter up to three F_p values (Fp1, Fp2, Fp3) for the degraded performance tuning. Performance and robustness of this tunings are compared with the optimal tuning (corresponding to $F_p = 1.0$).

(5) Start
Press the "Start" button to run the optimization and simulation program.

(6) Controller parameter, performance and robustness
In this area the controller parameters (K_p, T_i, T_d, β), and the control system performance (IAE_r, IAE_d) and robustness (M_S) for the optimal tuning ($F_p = 1$) and the selected degraded factors (Fp1, Fp2, and/or Fp3) are displayed.

(7) Control system responses
In this area the tool shows the control systems responses to a set-point step change followed by a disturbance step change. The responses can be displayed in a single MATLAB window using the [max] button.

(8) Step responses configuration
The total simulation time, the size and application instant of the set-point and disturbance changes, and other characteristics of the time responses can be configured. After that, the output display is [Refresh] using the controllers already obtained.

(9) Robustness plot
The Nyquist plot for the simulated control system with minimum and high robustness circles is displayed in this area. The robustness plot can be displayed in a single MATLAB window using the [max] button.

4. PERFORMANCE/ROBUSTNESS ANALYSIS

Consider the controlled process model given by the FOPDT transfer function

$$P_1(s) = \frac{1.25\mathrm{e}^{-4s}}{5s+1}. \tag{13}$$

For the first analysis, it is selected a 1DoF PI controller tuned for disturbance rejection (regulatory control) and use the degraded performance factor F_p to increase the resulting control system robustness. The results obtained adjusting the F_p values are shown in Fig. 3.

The regulatory performance optimized PI controller ($F_p = 1.0$) has poor robustness ($M_S = 2.51$). It is necessary to degrade the system performance with $F_p = 0.85$ to arrive to the minimum robustness level of $M_S = 2.0$.

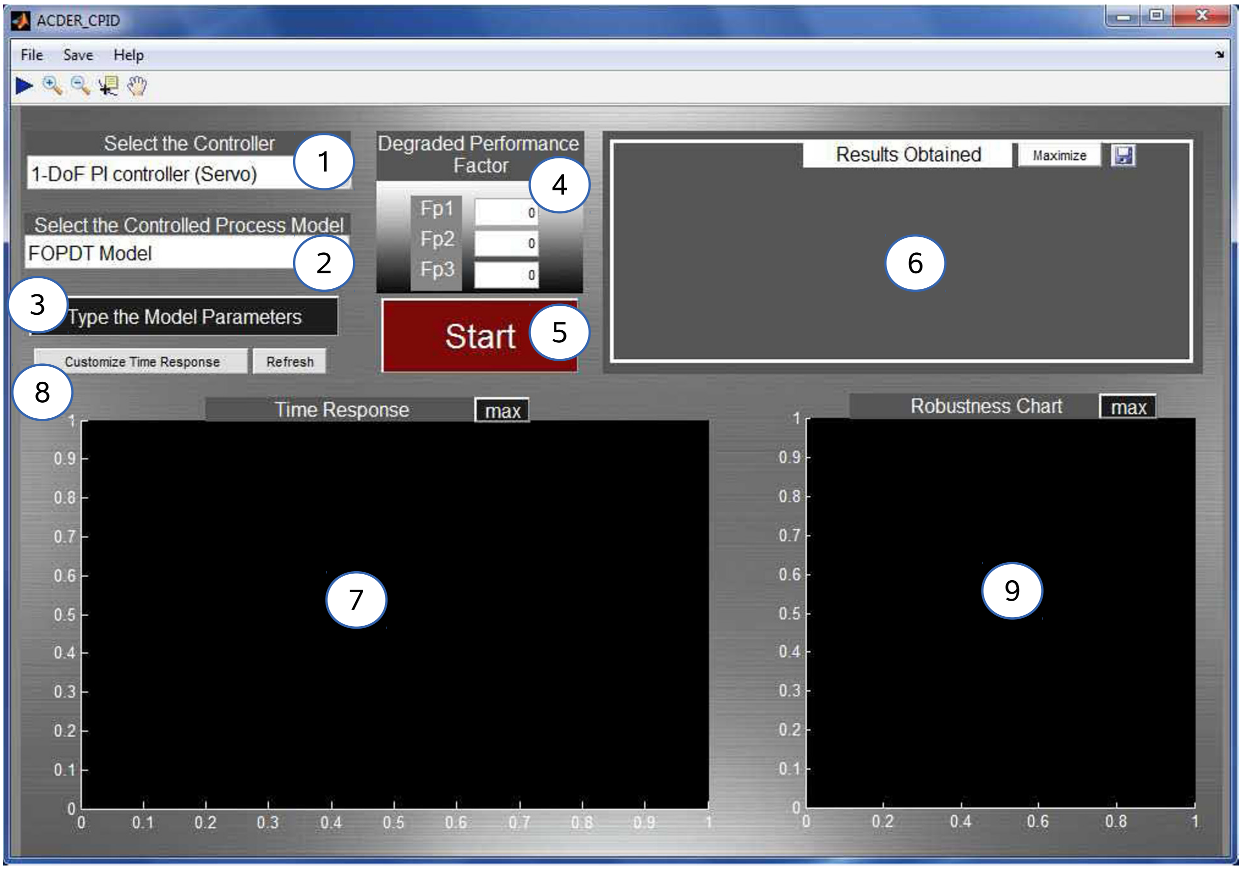


Figure 2. Tool Main Screen

If its performance is degraded further the robustness level increases: $F_p = 0.74 \rightarrow M_S = 1.80$ and $F_p = 0.62 \rightarrow M_S = 1.61$. It is interesting to note that the control system with $M_S = 2.0$ has better servo control performance that the regulatory performance optimized controller but, if robustness becomes very high the servo control performance also decreases.

The effects of reducing the control system performance to increase its robustness are noted in the control system responses and robustness plots shown in Fig. 3.

To control model (13), we select the same 1DoF controller but the design is for set-point tracking (servo control). The degraded performance factors, controllers parameters, and control systems performance and robustness are shown in Fig. 4.

It is possible to note that the servo control optimized PI controller is robust with $M_S = 1.86$ and, as expected, has better servo control performance (lower J_{er}) but lower regulatory control performance (higher J_{ed}) than the regulatory control performance optimized PI controller.

To achieve a system with $M_S = 1.8$ a very small reduction in performance is required ($F_p = 0.996$).

From a comparison over the same robustness level, for example $M_S = 1.8$ and $M_S = 1.6$, it is noted that, even that different degraded factors are required to arrive to the same target robustness, performances obtained (J_{er} and J_{ed}) are very similar.

Now, if model (13) is used again but now controlled with a 2DoF PID controller. The 2DoF controllers are tuned in two steps, first the controller parameters K_p, T_i, and T_d are obtained for optimal (or degraded) regulatory control performance and, in a second step, β is selected to obtain the best possible servo control performance (with the other controller parameters fixed).

The degraded performance factors, controllers parameters, and control systems performance and robustness with the 2DoF PID are shown in Fig. 5.

The performance optimized PID controller has a very low robustness ($M_S = 2.96$) but high performance. To bring the controller to the $M_S = 2.0$ level an $F_p = 0.75$ is required.

From the performance data in Figs. 3, 4, and 5 for $M_S = 1.6$ it is noticed that the 2DoF PID controller outperforms the PI controller in both operation modes (servo and regulation control).

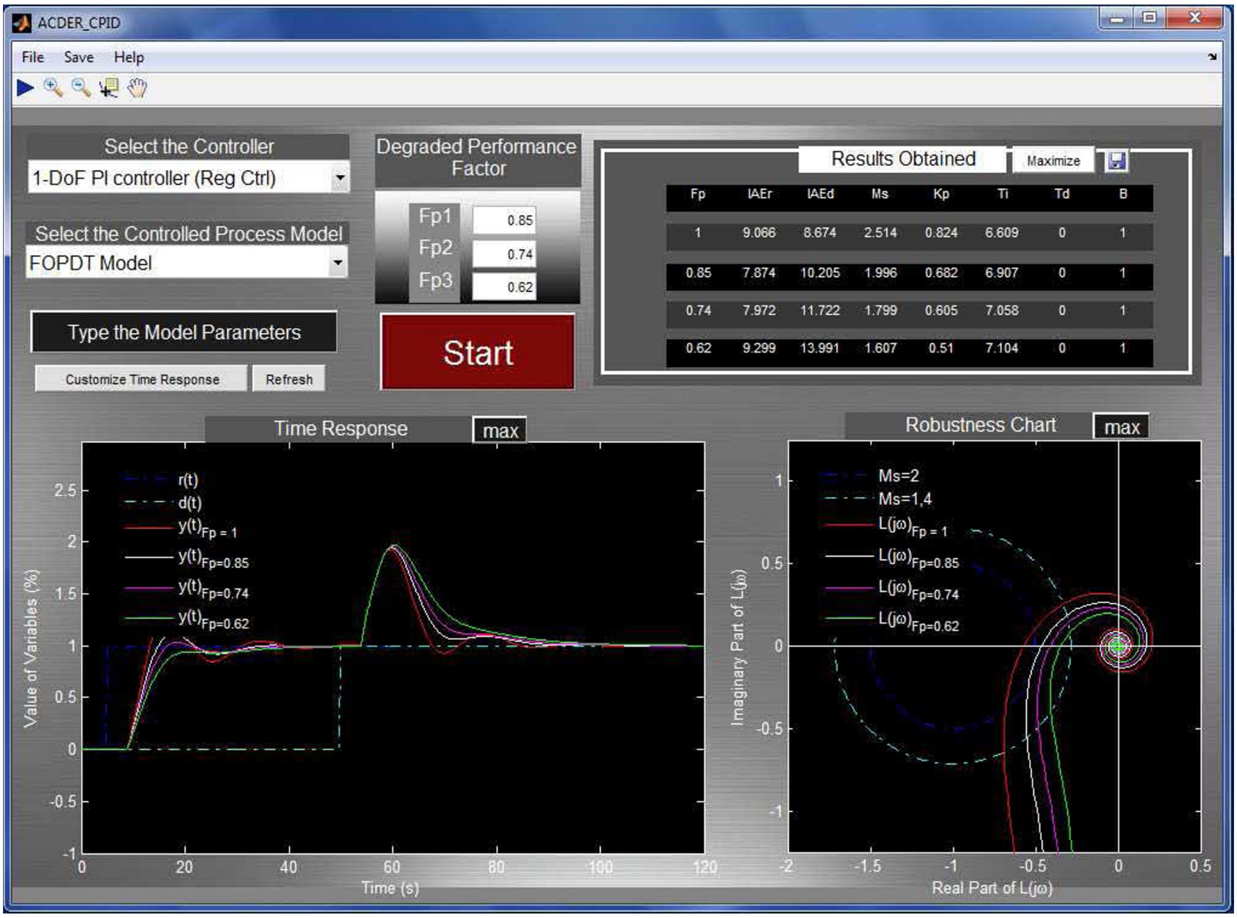
Consider now the SOPDT model

$$P_2(s) = \frac{0.5e^{-5s}}{(10s+1)(2s+1)}. \qquad (14)$$

and the use a 2DoF PID controller.

The performance and robustness of the performance optimized controller and two degraded performance controllers are shown in Fig. 6.

Results confirm that the IAE performance optimized regulatory control system is non-robust. It is necessary to



Fp	IAEr	IAEd	Ms	Kp	Ti	Td	B
1	9.066	8.674	2.514	0.824	6.609	0	1
0.85	7.874	10.205	1.996	0.682	6.907	0	1
0.74	7.972	11.722	1.799	0.605	7.058	0	1
0.62	9.299	13.991	1.607	0.51	7.104	0	1

Figure 3. Model: P_1, Controller: 1DoF PI (regulatory control)

Results Obtained

Fp	IAEr	IAEd	Ms	Kp	Ti	Td	B
1	7.809	8.921	1.864	0.741	6.572	0	1
0.986	7.92	9.636	1.795	0.718	6.877	0	1
0.825	9.466	11.868	1.599	0.61	7.207	0	1
0.62	12.595	15.79	1.404	0.464	7.305	0	1

Figure 4. Model: P_1, Controller: 1DoF PI (servo control)

Results Obtained

Fp	IAEr	IAEd	Ms	Kp	Ti	Td	B
1	7.642	5.174	2.955	1.106	4.825	1.627	0.505
0.75	7.904	6.898	1.999	0.824	4.97	1.666	0.729
0.56	8.286	9.238	1.604	0.611	5.152	1.713	1

Figure 5. Model: P_1, Controller: 2DoF PID

reduce the performance to obtain a robust control system. The tool also shows the "cost" of such performance reduction.

From the controllers parameters it is also possible to note that, to increase the control system robustness a reduction on the controller gain is required, while the integral time and derivative time remains more or less constant.

5. FEEDBACK FROM THE STUDENTS

The proposed software tool has been used by students of an entry level control system course at the School of Electrical Engineering of the University of Costa Rica.

After a lecture where the control system performance to robustness trade-off is presented, a project is assigned to design a PID controller with different controlled processes models. The students must report how much the control system robustness is increased degrading its performance and concluding about the compromise between them.

Students feedback, from the interface use, highlights that it provides them, in a quantitative and visual form, the information required to design a robust PID control system making very ease the selection of the "best" (in performance to robustness ratio sense) controller for the application.

6. CONCLUSIONS

An interactive software tool that allows the PID control design has been presented. For this, it is analyzed the performance/robustness trade-off and, performing an optimization procedure, the controller parameters are obtained. The tool is used to analyze several

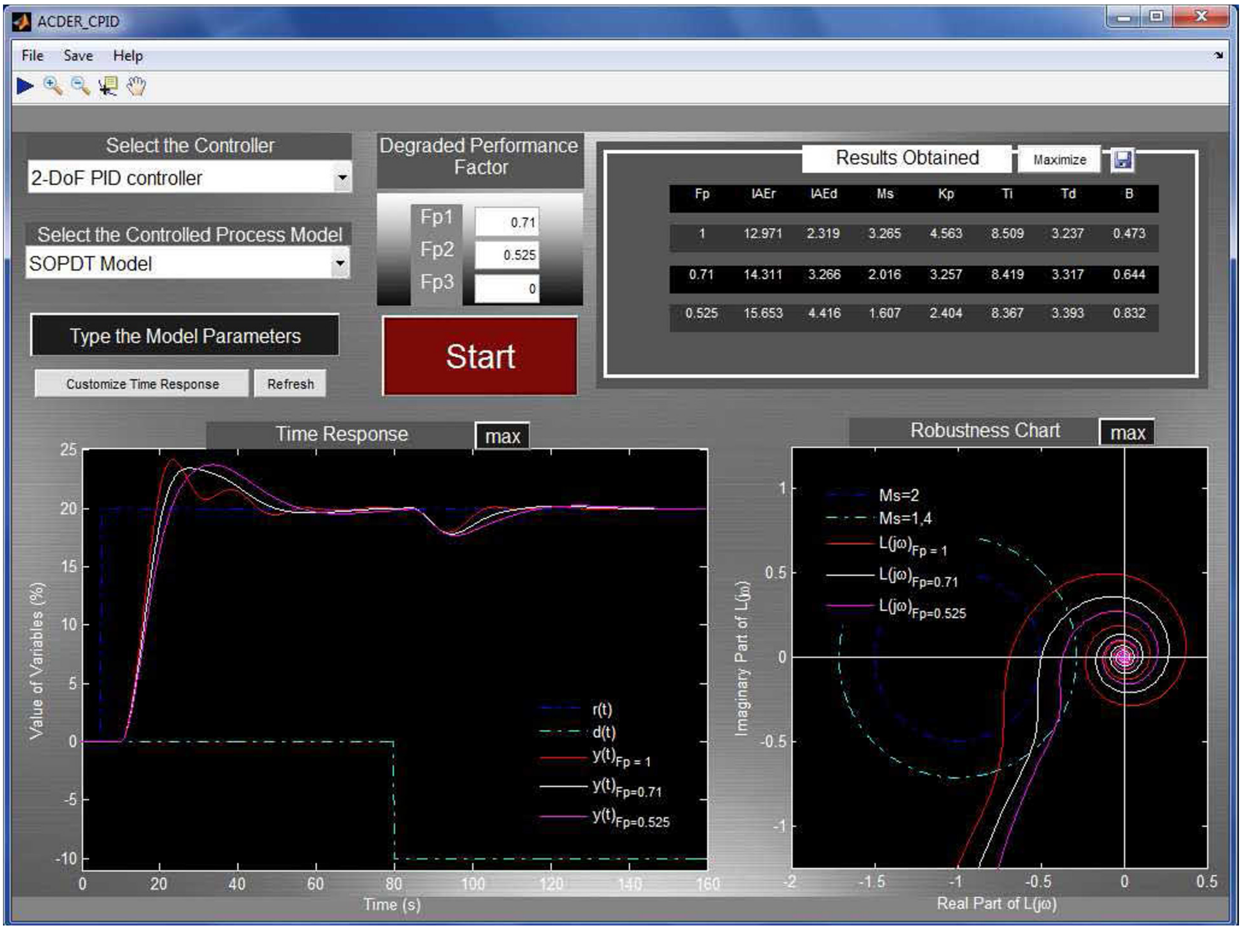


Figure 6. Model: P_2, Controller: 2DoF PID

model/controller combinations and it was concluded that degrading the control system performance to increase its robustness has a higher impact over the controller gain than over other parameters.

The software has been implemented using the graphical user interface (GUI) tool of MATLAB® and allows study the relation between performance and robustness in PID control by undergraduate students of a control system course. Considering the spread use of PID controllers in industrial applications the tool represents a contribution to the control system teaching-learning process.

ACKNOWLEDGEMENTS

The financial support from the University of Costa Rica, under the grant 322-B2-727, is greatly appreciated. Also, this work has received financial support from the Spanish CICYT program under grant DPI2010-15230.

REFERENCES

Alfaro, V.M., Méndez, V., Vilanova, R., and Lafuente, J. (2010). Performance/Robustness Tradeoff Analysis of PI/PID Servo and Regulatory Control Systems. In *IEEE International Conference on Industrial Technology (ICIT 2010)*. 14-17 March 2010, Viña del Mar, Chile.

Åström, K. and Hägglund, T. (2001). The future of PID control. *Control Engineering Practice*, 9, 1163–1175.

Åström, K. and Hägglund, T. (2006). *Advanced PID Control*. ISA - The Instrumentation, Systems, and Automation Society.

Dormido, S., Beschi, M., Sánchez, J., and Visioli, A. (2012). An interactive software tool for the study of event-based pi controller. In *IFAC Conference on Advances in PID Control*. March 28-30, Brescia, Italy.

Fernández, D.E. (2012). Software tool for performance/robustness analysis of PID control systems. Technical report, School of Electrical Engineering, University of Costa Rica, San José, Costa Rica. (in Spanish).

Garpinger, O., Hägglund, T., and Cederqvist, L. (2012). Software for PID design: Benefits and pitfalls. In *IFAC Conference on Advances in PID Control*. March 28-30, Brescia, Italy.

Guzmán, J., Rivera, D., Berenguel, M., and Dormido, S. (2012). *i-pIDtune*: An interactive tool integrated system identification and pid control. In *IFAC Conference on Advances in PID Control*. March 28-30, Brescia, Italy.

O'Dwyer, A. (2003). *Handbook of PI and PID Controller Tuning Rules*. Imperial College Press, London, UK.

The MathWorks Inc. (2011a). *MATLAB® - Creating Graphical User Interfaces (R2011a)*. Natick, MA, USA.

The MathWorks Inc. (2011b). *Optimization Toolbox User's Manual (R2011a)*. Natick, MA, USA.

Ziegler, J. and Nichols, N. (1942). Optimum Settings for Automatic Controllers. *ASME Transactions*, 64, 759–768.