

Tuning rules for robust FOPID controllers based on multi-objective optimization with FOPDT models

Helem Sabina Sánchez ^{a*}, Fabrizio Padula ^b, Antonio Visioli ^c, Ramon Vilanova ^a

^aDepartament de Telecomunicació i Enginyeria de Sistemes, ETSE

Universitat Autònoma de Barcelona

Carrer de es Sitges, 08193 Bellaterra, Barcelona, Spain

tel.: +34-93-5812197

fax: +34-93-5814031

e-mail: {helemsabina.sanchez,ramon.vilanova}@uab.cat

^bDepartment of Mathematics & Statistics

Curtin University

Kent St, Bentley WA 6102, Australia

tel.: +61- 89-2660000

fax: +61- 89-2660000

e-mail: {fabrizio.padula}@curtin.edu.au

^cDipartimento di Ingegneria Meccanica e Industriale

University of Brescia

Via Branze 38, I-25123 Brescia, Italy

tel.: +39-030-3715460

fax: +39-030-380014

e-mail: {antonio.visioli}@unibs.it



*Corresponding author.

Tuning rules for robust FOPID controllers based on multi-objective optimization with FOPDT models

Abstract

This paper addresses the problem of tuning fractional-order proportional-integral-derivative (PID) controllers for a balanced servo/regulation operation. In particular, we consider the control problem of minimizing the integrated absolute error for both the set-point step response and the load disturbance step response, by constraining at the same time the value of the maximum sensitivity in order to provide a satisfactory robustness. The control problem is therefore stated as a multi-objective optimization problem where a first-order-plus-dead-time process model has been considered. A set of Pareto optimal solutions is obtained for different normalized dead times and the optimal balance between the competing objectives is determined by choosing the Nash solution as a bargain game based trade-off solution among the Pareto-optimal ones. The results have been eventually fitted in order to obtain simple tuning rules. Several simulation results show the effectiveness of the proposed approach.

Keywords: Fractional-order PID controllers, Multi-objective optimization, Nash solution, Balanced tuning.

1. Introduction

A considerable number of tuning rules for Proportional-Integrative-Derivative (PID) controllers has been developed in the last decades. Indeed, they are with no doubt the most extensively employed controller in industrial applications [1, 2].

In recent years, fractional control has received a great interest from the control community due to the fact that it provides more flexibility in the design phase than the classical integer order one. This leads to controllers that are capable to accomplish more demanding control requirements [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]. Owing to this, it seems natural to generalize the derivative and the integral orders of the PID controller to any real number leading to the fractional-order PID (FOPID) controller [14, 15]. The use of a FOPID controller implies that a better performance can be achieved, but on the other hand it means that design can be more difficult. However, in order to foster the application of FOPID controllers in industry, the same ease of use as classical PID controllers must be ensured. For this purpose, the stability issue of this kind of controllers has been investigated in [16, 17]. Then, a large number of strategies to tune a FOPID controller has been proposed in the literature to facilitate their implementation [18]. In particular, optimization techniques have been proposed, mainly with the aim of achieving the so-called iso-damping property [19, 20, 21], that is, achieving a flat phase at the gain crossover frequency so that the same overshoot is obtained in the set-point step response despite process gain variations [22]. It has however to be recognized that, as for standard PID controllers, the presence of tuning rules can represent a key factor for the success of FOPID controllers. Taking this into account, rules that consider the optimization of the load disturbance response with a robustness constraint have been proposed [23, 24]. Similarly, the minimization of the integrated absolute error with a constraints on the maximum sensitivity has been pursued in [25], where both the set-point following and the load disturbance rejection tasks have been considered separately.

It turns out that a FOPID controller tuning rule that addresses both the set-point following task (servo mode) and the load disturbance task (regulatory mode) at the same time is still missing. Indeed, for the design of a control system, it is important to take into account the trade-off between these specifications. Moreover, the other important trade-off between performance and robustness

has to be considered. Nevertheless, it is well known that these are in general competing objectives that have been already investigated for standard, integer order, PID controllers (see [26], where however the robustness issue is disregarded).

In this context, as there are different conflicting requirements to handle, it is natural to set up a Multi-Objective Problem (MOP) [27]. In fact, in general, a good disturbance rejection response is not compatible with a good set-point step response and a high performance is often not compatible with a controller which is robust to process model mismatch. The controller design can be therefore be viewed as the search for the best compromise between all the specifications and thereby the idea of using a Multi-Objective Optimization Design (MOOD) procedure can be an effective approach to solve this kind of problems [28]. Eventually, tuning techniques can be obtained based on the so called Pareto front approximation, where all the solutions are Pareto-optimal and offer different trade-offs between the objectives, see for example [29, 30, 31, 32].

In this paper the MOOD procedure is implemented in order to find a controller with an optimal balance between the posed objectives. For this purpose a new approach for the Multi-Objective Optimization (MOO) procedure has been implemented in order to improve the convergence capabilities. We consider a servo/regulatory optimally balanced tuning, where the MOO procedure is performed by optimizing the set-point following and the load disturbance rejection performance, measured in terms of the integrated absolute error (IAE) and considering First-Order-Plus-Dead-Time (FOPDT) processes with different normalized dead times. In order to achieve a reasonable performance/robustness trade-off, the maximum sensitivity M_s has been used as a measure of the system robustness. Regarding the robustness constraint, a double approach has been pursued: in the first case the value of M_s has been constrained in a reasonable range while in the second case the value of M_s has been constrained to a specific value that the user can selected depending on the application. Then, in order to select the best compromise between the different objectives, the Nash solution (NS) has been determined [33] as a Multi-Criteria Decision Making (MCDM) technique for each normalized dead time. Finally, tuning rules have been determined by using a least squares fitting technique with the obtained optimal results with respect to the normalized dead time. It is worth to highlight that the obtained tuning rules follow the same structure for all the situations considered.

The main contribution of the present paper are the balanced tuning rules. Existing optimal tunings for fractional-order PID controllers should of course offer better performance than the one proposed in this paper as far as just one operation mode is considered. The proposed controller tuning provides a better trade-off when a global performance is considered, that is, when the control systems addresses both the operating modes. This is operating in any one of both modes. The key point achieved by the use of the multi-objective optimization approach is that of maintaining as much as possible the optimality with respect to both tasks. Hence, it is from a global viewpoint that the proposed tuning rules for fractional-order PID controllers provides an improvement with respect to pre-existing results. In this context, the paper also proposes a Global Performance Index (GPI) that includes the different considered performance indicators. Simulations results show the effectiveness and the robustness of the proposed tuning rules and the advantage of a unique tuning strategy capable to address both the servo and the regulatory tasks.

The paper is organized as follows. In Section 2 the control system is described while, in Section 3, the control problem is formulated. In Section 4 the MOOD procedure for tuning the FOPID controller is described. In Section 5 the optimization results are presented along with the optimal tuning rules. Section 6 is devoted to the performance assessment. Simulation results are presented and discussed in Section 7. Finally, conclusions are drawn in Section 8.

2. Control system

The purpose of a control system is to obtain a desired response for a given system. We consider the typical feedback control system represented in Figure 1, where P is the process, K is the controller, r is the set-point signal, u is the control signal, d is the load disturbance signal, y is the process output and $e := r - y$ is the control error. The process dynamics, described by the transfer function $P(s)$, is considered to be self-regulating and represented by a FOPDT model of the form:

$$P(s) = \frac{K}{1 + T_s} e^{-Ls}, \quad (1)$$

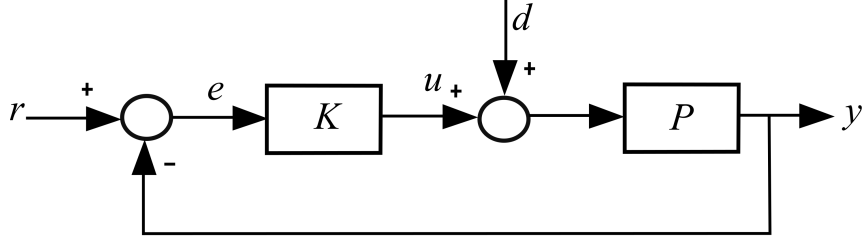


Figure 1: Feedback control scheme.

where K is the process gain, T is the time constant and L is the dead time. The process dynamics can be fully characterized in terms of the normalized dead-time defined in (2), which represents a measure of the difficulty in controlling the process:

$$\tau = \frac{L}{T}, \quad \tau \in [0.1, 4]. \quad (2)$$

Note that FOPDT models are frequently used in process control because they are simple and they describe with a sufficient accuracy the dynamics of many industrial processes [2]. Further, such models can be obtained by means of very simple step tests or relay experiments and therefore they are also often employed in automatic tuning techniques.

In this paper, the process will be controlled with a fractional-order PID controller, which is defined as a generalization of the standard ISA form for the PID controller, whose transfer function is given by [34]:

$$K(s) = K_p \left(1 + \frac{1}{T_i s^\lambda} + \frac{T_d s^\mu}{\frac{T_d}{N^\mu} s^\mu + 1} \right), \quad (3)$$

where K_p is the proportional gain, T_i is the integral time constant, T_d is the derivative time constant, N is the derivative filter parameter, λ and μ are the non-integer order of the integral and of the derivative actions, respectively (*i.e.*, the tuning parameters).

Remark 1. *It is worth mentioning that the derivative filter time constant $\frac{T_d}{N^\mu}$ is selected in such a way that the phase transition of the filter occurs in $\log(N)$ decades after the derivative term $T_d s^\mu$ crosses the 0 dB axis, regardless of the derivative order μ . Indeed, selecting the filter time constant as $\frac{T_d}{N^\mu}$ prevents the fractional poles migration [35] when the derivative order changes. Here, $N = 10$ has been selected as it is usual in industrial controllers [36, 2].*

Finally, it must be pointed out that, in order to implement the fractional-order controller, the Oustaloup continuous integer-order approximation [6] has been used; it consists in using the following approximation based on a recursive distribution of zeros and poles:

$$s^\nu \cong k \prod_{n=1}^{\bar{N}} \frac{1 + \frac{s}{\omega_{z,n}}}{1 + \frac{s}{\omega_{p,n}}}, \quad \nu > 0, \quad (4)$$

where $\omega_{z,n}$ and $\omega_{p,n}$ are, respectively, the frequencies at which the zeros and the poles occurs, $\nu \in \mathbb{R}$ is the fractional order, \bar{N} is the number of poles and zeros used for the approximation and k is the gain. The approximation is only valid in a frequency interval $[\omega_l, \omega_h]$, where ω_l and ω_h are, respectively, the lower and the higher limit. Finally, the gain is adjusted so that both sides of (4) have the same gain in the logarithmic mid point of the interval.

In this paper the values of ω_l and ω_h have been selected as $0.001\omega_c$ and $1000\omega_c$, respectively, where ω_c is the gain crossover frequency of the loop transfer function. Furthermore, \bar{N} has been chosen equal to 8. It is worth stressing that the use of these criteria to select the approximation parameters leads to a mismatch between the responses that would be obtained with the ideal fractional controller and the approximated controller that is negligible. Indeed, within the approximation frequency range, the selected number of poles and zeros makes the two controllers virtually undistinguishable. Below the lower approximation limit, because of the large approximation range centered at ω_c , the presence of a (possibly fractional) integrator flats the closed-loop frequency response, both in the ideal and in the approximated cases, along the 0 dB axis. Finally, above the upper limit, the high frequency roll-off makes again the ideal-approximated closed-loop mismatch negligible.

3. Problem formulation

As stated in the introduction, we aim at finding an optimal tuning for the proposed FOPID controller taking in to account both the servo and the regulatory modes.

In order to measure the performance of a given set of tuning parameters, we use here the Integrated Absolute Error (IAE) defined as

$$IAE := \int_0^\infty |e(t)| dt = \int_0^\infty |r(t) - y(t)| dt. \quad (5)$$

Indeed, it is well-known that in general it yields both a low overshoot and a low settling time at the same time [37].

Nevertheless, since we must take into account both the servo and the regulatory mode, we must consider two different IAEs, the one obtained in the set-point step response (servo mode) and the other obtained in the load-disturbance step response (regulatory mode). The problem arises since these two objectives are conflicting. Indeed, it is well known that a good servo tuning may result in a sluggish disturbance rejection response and, viceversa, a good regulatory tuning may result too aggressive for the servo task ending up in an oscillatory set-point response.

Hence, the FOPID controller has to be tuned in order to simultaneously minimize two cost functions: the IAE of the set-point step response, IAE_{sp} , and the IAE of the load disturbance step response, IAE_{ld} .

Moreover, aiming just at obtaining the minimal IAE may lead to a poor control performance because the robustness issue is not taken into account. For this reason, in this work we also consider the maximum sensitivity, defined as

$$M_s := \max_{w \in [0, +\infty)} \left| \frac{1}{1 + K(s)P(s)} \right|_{s=jw} . \quad (6)$$

It represents the inverse of the minimum distance of the Nyquist plot from the critical point (-1,0). This index is an indication of the system robustness (relative stability) and typical values range from 1.4 (robust tuning) to 2.0 (aggressive tuning) [38]. The maximum sensitivity is used as a constraint during the IAE minimization procedure and may be fixed to a specific value or constrained to a given range. Both approaches will be pursued in the paper.

Bearing in mind the previous reasoning, it appears that the control problem has a clear multi-objective nature. Accordingly, in the next section, after introducing the fundamentals of MOO, the control problem will be recast as MOP.

4. Multi-objective optimization

This section provides an overview about multi-objective optimization, showing how the FOPID controller design can be tackled with this technique. As it has been mentioned in [39], a MOP can

always be expressed as follows:

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^n} \mathbf{J}(\boldsymbol{\theta}) = [J_1(\boldsymbol{\theta}), \dots, J_m(\boldsymbol{\theta})] \in \mathbb{R}^m \quad (7)$$

subject to:

$$\mathbf{g}(\boldsymbol{\theta}) \leq 0 \quad (8)$$

$$\mathbf{h}(\boldsymbol{\theta}) = 0 \quad (9)$$

$$\theta_{li} \leq \theta_i \leq \theta_{ui}, \quad i = [1, \dots, n] \quad (10)$$

where $\boldsymbol{\theta} = [\theta_1, \dots, \theta_n]$ is defined as the decision variables vector, $\mathbf{J}(\boldsymbol{\theta})$ is the objective vector, $\mathbf{g}(\boldsymbol{\theta})$ and $\mathbf{h}(\boldsymbol{\theta})$ are the inequality and equality constraint vectors, respectively, and θ_{li} and θ_{ui} are the lower and upper bounds in the decision space of the θ_i variable. Therefore, a set of optimal solutions is defined as the Pareto set and each solution within this set defines an objective vector. The projection into the objective space is known as Pareto front (PF). All the solutions in the PF are said to be non-dominated and Pareto-optimal solutions. This means in the PF, there is not a solution that is better than another one for all the conflictive objectives. To improve in one objective will imply to introduce a loss regarding the other ones. It is important to mention that the true PF is unknown, for this reason MOO techniques search for a discrete description of the Pareto set capable of generating a good approximation of the PF, see Figure 2. Finally, when the PF approximation has been obtained, the multi-criteria decision making (MCDM) stage is carried out in order to select the most preferable solution for the designer.

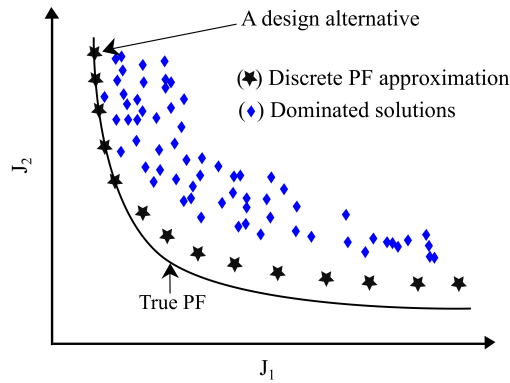


Figure 2: PF concept and design alternative (for a particular design concept).

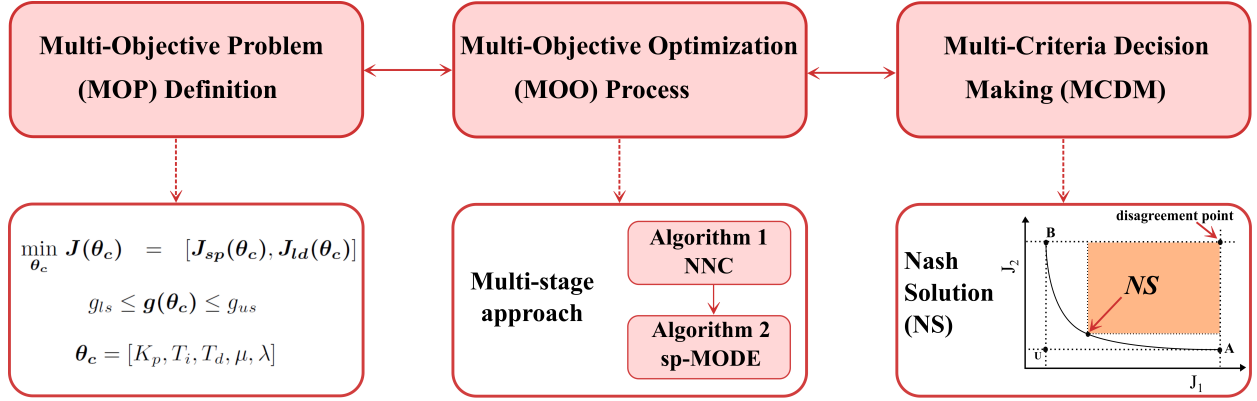


Figure 3: Multi-objective Optimization Design Procedure for FOPID controller.

In order to incorporate the MOO into any engineering design procedure, a general framework is required [40, 41]: i) the MOP definition (objectives, decision variables and constraints), ii) the MOO process (search) and iii) the MCDM stage (analysis and selection). Overall, this procedure is known as Multi-objective Optimization Design (MOOD) procedure and is represented in Figure 3. Hence, to obtain the best trade-off from the Pareto front approximation for the controller tuning, the aforementioned steps are needed [28]. In the next subsection, they will be specifically developed for the stated problem.

4.1. MOP definition for the FOPID controller

In the previous two sections the control system has been described and the control problem formulated. Therefore, the MOP statement for FOPID controller tuning could be formulated as:

$$\min_{\theta_c} \mathbf{J}(\theta_c) = [\mathbf{J}_{sp}(\theta_c), \mathbf{J}_{ld}(\theta_c)] \quad (11)$$

such that

$$g_{ls} \leq \mathbf{g}(\theta_c) \leq g_{us}, \quad (12)$$

where

$$\theta_c = [K_p, T_i, T_d, \mu, \lambda]$$

are the parameters of the FOPID controller (decision space).

According to the control problem stated in Section 3, the goals for the controller tuning are the

minimization of the IAEs for the set-point and the load disturbance step response, namely:

$$\mathbf{J}_{sp}(\boldsymbol{\theta}_c) = IAE_{sp} := \int_0^{\infty} |e(t)| dt, \quad d = 0 \quad (13)$$

when a set-point step response is considered, and

$$\mathbf{J}_{ld}(\boldsymbol{\theta}_c) = IAE_{ld} := \int_0^{\infty} |e(t)| dt, \quad r = 0 \quad (14)$$

for a load disturbance step response.

The minimization is constrained by using the maximum sensitivity M_s , that is:

$$\mathbf{g}(\boldsymbol{\theta}_c) = M_s = \max_{w \in [0, +\infty)} \left| \frac{1}{1 + K(s; \boldsymbol{\theta}_c)P(s)} \right|_{s=jw} \quad (15)$$

$$g_{ls} \leq \mathbf{g}(\boldsymbol{\theta}_c) \leq g_{us}, \quad (16)$$

where g_{ls} , g_{us} , are the lower ($M_s = 1.4$) and upper ($M_s = 2.0$) constraint limits.

As stated in Section 3, M_s can be constrained to be into a range or to be fixed to a specific value.

Therefore, the alternative to the inequality constraint (12), is to use an equality constraint

$$\mathbf{h}(\boldsymbol{\theta}_c) = h, \quad (17)$$

where $\mathbf{h}(\boldsymbol{\theta}_c) = M_s$ and $h \in \{1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0\}$.

4.2. MOO

In this paper, a multi-stage approach is proposed (see Figure 4), where two algorithms are merged¹ in a sequential manner for controller tuning, when a MOP as described by (11)-(12) is being solved:

- The Normalized Normal Constraint (NCC) algorithm: it is a deterministic algorithm proposed in [42]. It is used in a bi-objective statement, where the optimization problem is less

¹Both algorithms are available in Matlab Central at <http://www.mathworks.com/matlabcentral/fileexchange/38976> and <http://www.mathworks.com/matlabcentral/fileexchange/39215>.

sensitive to the anchor solutions and more likely to converge to the PF. In this context, it is in charge of searching the local space (convergence area). This algorithm incorporates a critical linear mapping of the design objectives. This mapping has the desirable property that the resulting performance of the method is entirely independent of the design objectives scales and in the ability to generate a well distributed set of Pareto points even in numerically demanding situations.

- The Multi-objective Differential Evolution Algorithm with Spherical Pruning (sp-MODE): it is an evolutionary optimization approach based on differential evolution [43]. This algorithm is in charge of searching the global space. It uses the locally Pareto-optimal approximation calculated by the deterministic (NCC) approach as the initial population. The basic idea of the spherical pruning is to analyze the proposed solutions in the current PF approximation by using normalized spherical coordinates from a reference solution.

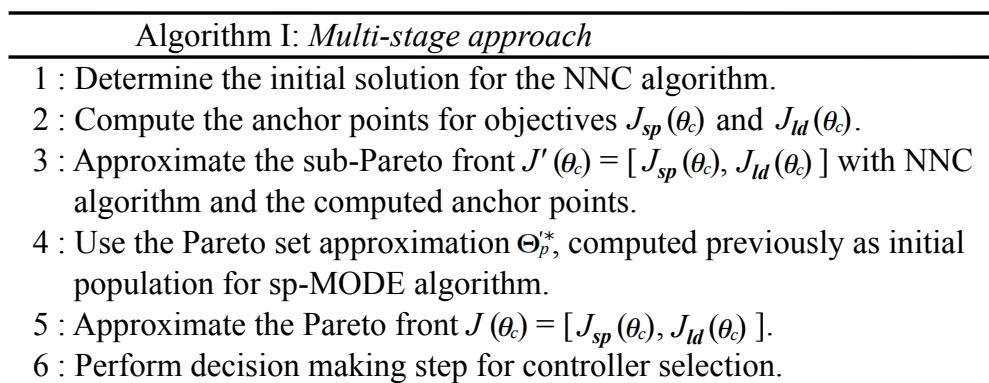


Figure 4: Algorithm I: Multi-stage approach

It should be noted that using this approach the sought set of solutions will have some desirable characteristics as exploitation capabilities (local and dominated solution are avoided from the Pareto front). Furthermore, it allows both algorithms to complement each other despite of their drawbacks and to improve the results of optimization in terms of convergence and accuracy. This approach has been shown in a previous works [44] to be effective in addressing control engineering problems.

This approach will improve the exploitation capabilities of the algorithms and it will reduce the numerical burden associated with the generation of the Pareto front approximations for reliability problems. It is worth stressing that, if a solution gives the desirable trade-off, then it is possible to optimize such solution, looking for a locally Pareto optimal solution that dominates the preferable solution. In this way, the preferability (from the point of view of the trade-off among conflicting objectives) and convergence are fulfilled.

4.3. MCDM stage

In this final step of the MOOD procedure, the designer needs to analyze the trade-off among the competitive objectives and select the most reasonable solution according to his/her preferences. In fact, all points within the PF are equally acceptable solutions. However, there is the need to choose one of that points as the final solution to the MOP for the implementation phase. It is worth recalling that an analytical expression of the PF is usually unknown and the decision-making can only rely on the discrete approximation. Therefore, for decision making, we propose the Nash Solution (NS) [33]. This criteria provides an automatic selection and a direct approach for the choice of one point from the PF approximation. The NS lies on the PF surface and is computed as the intersection between the surface and the diagonal line that passes through two opposite vertexes of the smallest cube that inscribes the surface. In Figure 3, in the MCDM stage there is an image that shows where the NS is located into the PF. As it can be seen, there is a rectangle defined by the disagreement point, B, NS and A. The disagreement point can not be a common solution for the MOP because none of the objectives would agree on this point due to the fact that it represents the worst situation. Furthermore, this point can be improved with respect to both objectives; the area of the rectangle (see Figure 3) represents the set of solution dominated by the NS. Therefore, the NS provides the PF solution that dominates the larger number of solutions (*being absolutely better*), with respect to both objectives at the same time. This criterion has already been used successfully in [45, 46, 47].

5. Optimal tuning

In order to find a set of optimal tuning rules for the FOPID controller, the MOOD procedure stated in the previous section has been applied to a set of FOPDT process models with different values of the normalized dead time τ ranging from 0.1 to 4.0. Note that the considered range of normalized dead times spans from lag dominant processes to truly delay dominant ones, being the upper limit $\tau = 4.0$ whereas most of the existing tuning rules cover just till $\tau = 2.0$.

The gain of each process has been set equal to one and the time constant has been normalized to one without loss of generality since the gain is a pure scale factor and the process dynamics is completely parameterized by using the normalized dead time τ .

Unit step signals have been employed for the set-point following and load disturbance rejection tasks in order to compute the corresponding IAEs (13)-(14).

The optimization problem (11) has been solved by using two different types of constraints:

1. the maximum sensitivity M_s has to be in a range (hereafter addressed as M_s -range case):

$$1.4 \leq M_s \leq 2.0; \quad (18)$$

2. the maximum sensitivity M_s has to be equal to a given value (hereafter addressed as M_s -value case):

$$M_s = \{1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2.0\}. \quad (19)$$

The following framework has been used in order to solve the MOO problem:

- the servo and the regulation operation modes have been considered in the same optimization problem;
- as initial guess for the NNC algorithm, the optimal parameters proposed in [25] have been used and the parameter search space selected as $K_p \in [0.1, 11]$, $T_i \in [0.01, 3]$, $T_d \in [0.01, 3]$ and $\mu \in [0.9, 2]$;
- the sp-MODE algorithm has been used with:

- the scaling factor (F), which is a real and constant factor that controls the amplification of the differential variations that allows the mutant vector. In this context, F=0.5 is usually a good initial choice (see [48]);
- for each target vector and its mutant vector, a trial child vector is calculated. In order to increase the diversity of the disturbed parameter vector the Crossover rate (Cr) is introduced. A good choice for it is Cr=0.9, according to the guidelines proposed in [48, 43, 49].

The outcome of the MOO procedure is a set of PFs, for different normalized dead times and, in the M_s -value case, for different maximum sensitivities. It is worth stressing that the PFs obtained for the M_s -value case are two dimensional curves where the x and y axes are, respectively, IAE_{ld} and IAE_{sp} . Conversely, for the M_s -range case, the obtained PFs are three dimensional surfaces where the x and y axes remain the same, while the z axis is M_s .

Once a suitable set of PFs has been obtained, the NS for each PF is computed. An optimal tuning for the FOPID controller, *i.e.*, a set of optimal parameter K_p, T_i, T_d, λ and μ , corresponds to each NS.

It is worth noting that, when considering the IAE minimization as the objective function, the obtained optimal integrator is always of integer order (that is $\lambda = 1$), as pointed out in [25, 50, 34]. In fact, the optimizer always tries to decrease the integrator order λ . However, for $\lambda < 1$ the IAE is unbounded. Indeed, by using the final value theorem it is immediate to check that the integral error for the set point unit step response is

$$\lim_{s \rightarrow 0} \frac{s}{s^2} \frac{1}{1 + P(s)K(s)} = \frac{1}{s} \frac{s^\lambda \left(1 + \frac{T_d}{N^\mu} s^\mu\right) (Ts + 1)}{\left(1 + \frac{T_d}{N^\mu} s^\mu\right) (Ts + 1) + (T_i s^\lambda + 1)(T_d s^\mu + 1)e^{-Ls}}, \quad (20)$$

while the integral error for the unit load disturbance step response is

$$\lim_{s \rightarrow 0} \frac{s}{s^2} \frac{P(s)}{1 + P(s)K(s)} = \frac{1}{s} \frac{s^\lambda \left(1 + \frac{T_d}{N^\mu} s^\mu\right)}{\left(1 + \frac{T_d}{N^\mu} s^\mu\right) (Ts + 1) + (T_i s^\lambda + 1)(T_d s^\mu + 1)e^{-Ls}}. \quad (21)$$

Evidently, both of them are unbounded when $\lambda < 1$, but a bounded integral error is a necessary condition for a bounded IAE. Hence, the solution $\lambda = 1$ is chosen, also coherently with the necessary condition for the existence of a finite IAE presented in [51]. This allows the selection of

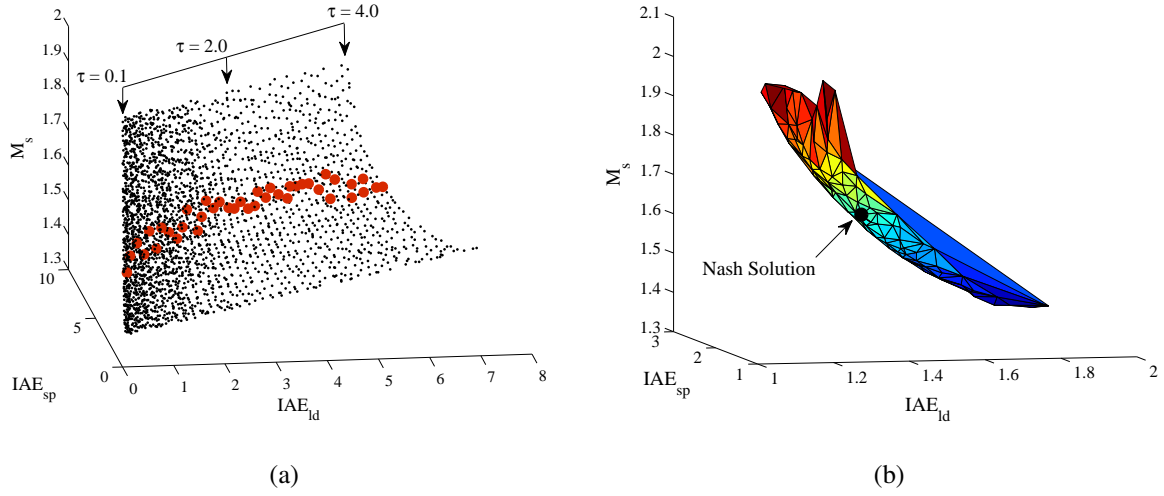


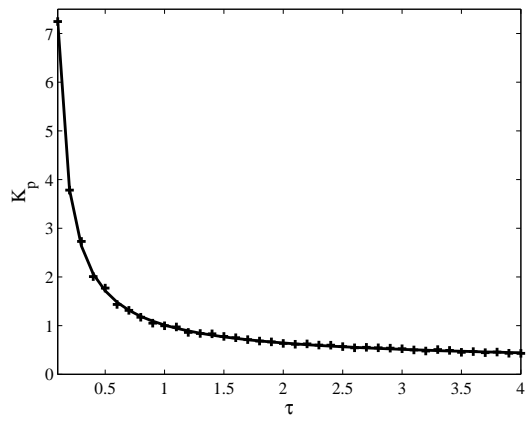
Figure 5: (a) PFs for the M_s -range case for different normalized dead times and the corresponding NSs and (b) the PF for $\tau = 0.7$ and the corresponding NS.

the optimal integrator order analytically. Moreover, another advantage of these results is that the dimensionality of the optimization problem is reduced since λ is fixed in advance.

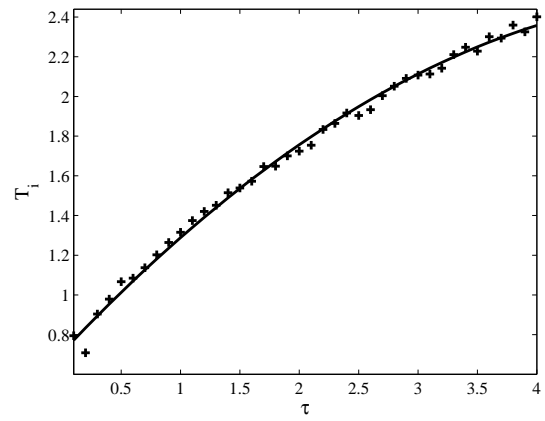
Eventually, all the optimal parameters are computed as functions of the normalized dead time and, for the M_s -value case, also as functions of the desired maximum sensitivity. The fitting function, suitably scaled with respect to the process dc-gain and with respect to process time constant will constitute the set of tuning rules.

5.1. M_s -range case

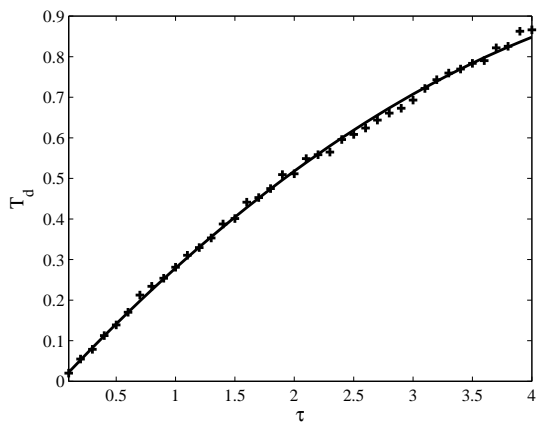
As a first case we consider the M_s -range case introduced above. We consider the MOP where the M_s is constrained in a range between 1.4 and 2.0. The results obtained for the different normalized dead times are shown in Figure 5, where a set of PFs in the $(IAE_{sp}, IAE_{ld}, M_s)$ space is obtained and the NSs are displayed. It is worth stressing that PFs are actually surfaces even if this can be hardly appreciated from the figure. Moreover, in order to clarify this issue, the PF for the case $\tau = 0.7$ is shown as an illustrative example.



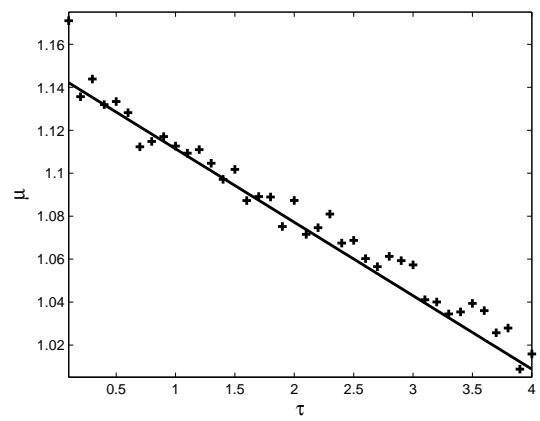
(a)



(b)



(c)



(d)

Figure 6: Tuning parameters for the FOPID controller for the M_s -range case. *Plus sign*: optimal values of the parameter. *Solid line*: fitting function.

Table 1: Tuning rules coefficients of the FOPID controllers for the M_s -range case.

	a	b	c	d
K_p	0.5951	-1.0417	0.4161	-0.2331
T_i	-0.0558	0.6356	0.7086	
T_d	-0.0246	0.3127	-0.0091	
μ	-0.0342	1.1456		

5.1.1. Tuning rules

After the MCDM stage is carried out, a set of NSs is obtained. Each one of them is the optimal solution for a different normalized dead time τ . A set of optimal tuning parameters $[K_p, T_i, T_d, \lambda, \mu]$, therefore a specific optimal controller tuning, corresponds to each NS.

The whole set of tuning parameters has been calculated by using the least squares fitting technique, as a function of the normalized dead time τ , leading to the results shown in Figure 6. Eventually, the following structure for the controller parameters has been devised:

$$K_p = \frac{1}{K} (a\tau^b + c\tau^d), \quad (22)$$

$$T_i = T^\lambda (a\tau^2 + b\tau + c), \quad (23)$$

$$T_d = T^\mu (a\tau^2 + b\tau + c), \quad (24)$$

$$\mu = a\tau + c, \quad (25)$$

$$\lambda = 1, \quad (26)$$

where the values of the coefficients are show in Table 1. It is worth noting that the fitting functions have been scaled with respect to the process dc-gain K and with respect to the process time constant T . In this way the obtained tuning rules have the nice property of being both time scale and gain invariant. In order to achieve this result, the scaling factor of T_i and T_d depends, respectively, on λ and μ . This prevents, again, the fractional pole/zero migration when the orders change.

Table 2: Tuning rules coefficients for the FOPID controllers for the M_s -value case with $M_s = 1.4$.

	a	b	c	d
K_p	0.4937	-1.0286	0.2082	-0.1928
T_i	-0.0190	0.3847	0.7913	
T_d	-0.0137	0.3188	-0.0117	
μ	-0.0691	1.1682		

5.2. M_s -value case

As a second case we consider the M_s -value approach, in which the MOP considered has the value of M_s constrained to each one of the specific values of the set (19). Following the same approach used in the previous section, the MOOD procedure has been repeated for different normalized dead times and for different value of M_s . For each PF obtained, the NS is eventually computed.

5.2.1. Tuning rules

Once again, in order to obtain a set of tuning rules for each considered M_s value, each tuning parameter has been calculated by using the least squares fitting technique, as a function of the normalized dead time τ . Hence, the number of fitting is equal to the number of considered levels of M_s , *i.e.*, seven fitting have been performed for each parameter. The structure of the obtained tuning rules for the controller parameters is reported in (22)-(26), where the values of the coefficients are in Tables 2-8, depending on the desired level of robustness. It is worth noting that the formulas are the same as the M_s -range case due to the fact that the parameters have the same trend. The plots of the fitting curves are not shown for the sake of brevity.

6. Performance assessment

The performance assessment of the proposed rules is addressed in this section in order to provide, in a single and direct way, the expected outcome for the selected controller. The main purpose of the performance assessment is to allow the user to know in advance the values of the

Table 3: Tuning rules coefficients for the FOPID controllers for the M_s -value case with $M_s = 1.5$.

	a	b	c	d
K_p	0.2321	-0.0826	0.5987	-1.0242
T_i	-0.0448	0.5446	0.6857	
T_d	-0.0165	0.3128	-0.0102	
μ	-0.0467	1.1556		

Table 4: Tuning rules coefficients for the FOPID controllers for the M_s -value case with $M_s = 1.6$.

	a	b	c	d
K_p	0.3812	-0.2357	0.5658	-1.0813
T_i	-0.1105	0.8771	0.5574	
T_d	-0.0177	0.3043	-0.0085	
μ	-0.0449	1.1555		

Table 5: Tuning rules coefficients for the FOPID controllers for the M_s -value case with $M_s = 1.7$.

	a	b	c	d
K_p	0.7256	-1.0221	0.3064	-0.0624
T_i	-0.0900	0.8472	0.5113	
T_d	-0.0273	0.3198	-0.0075	
μ	-0.0343	1.1403		

Table 6: Tuning rules coefficients for the FOPID controllers for the M_s -value case with $M_s = 1.8$.

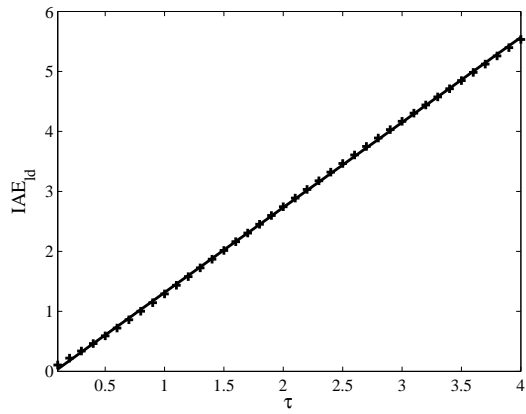
	a	b	c	d
K_p	0.6788	-1.0718	0.4356	-0.1779
T_i	-0.0625	0.7417	0.5621	
T_d	-0.0249	0.3187	-0.0087	
μ	-0.0343	1.1378		

Table 7: Tuning rules coefficients for the FOPID controllers for the M_s -value case with $M_s = 1.9$.

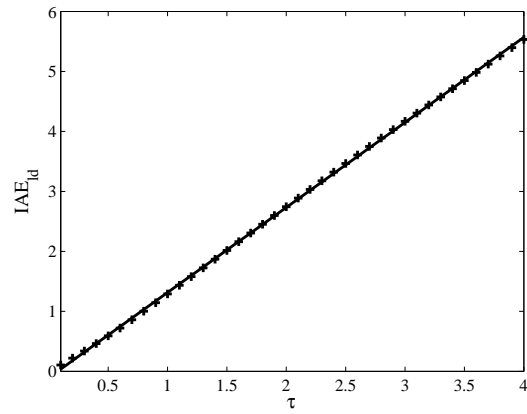
	a	b	c	d
K_p	0.6818	-0.3413	0.4936	-1.1998
T_i	-0.0767	0.8260	0.4788	
T_d	-0.0272	0.3320	-0.0111	
μ	-0.0464	1.1544		

Table 8: Tuning rules coefficients for the FOPID controllers for the M_s -value case with $M_s = 2.0$.

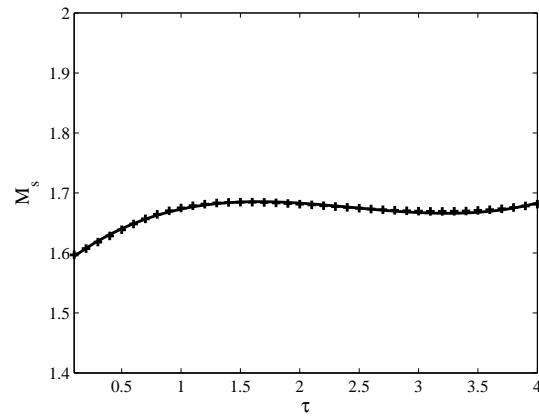
	a	b	c	d
K_p	0.4646	-0.1509	0.7615	-1.0902
T_i	-0.0571	0.7219	0.5585	
T_d	-0.0301	0.3464	-0.0133	
μ	-0.0440	1.1430		



(a)



(b)



(c)

Figure 7: Performance and robustness assessment index function for the range case. *Solid line*: fitting function (27), (28) and (29). *Plus sign*: optimal value of IAE_{ld} , IAE_{sp} and M_s for the M_s -range case.

IAE cost function that will be obtained (both for set-point following and the load disturbance rejection) when applying the proposed tuning rules. Thus, the user can evaluate the effectiveness of the proposed rules against other ones and decide if they are suitable for a given application without the need of a simulation. Further, the user can calculate the IAE obtained with an existing controller and he/she can decide if it is worth substituting it, as the performance (and the robustness) that can be achieved with the proposed tuning rules can be determined a priori by knowing the FOPDT process model.

When using M_s -range rules, the obtained maximum sensitivity is not constrained to a specific value, but in a range from 1.4 to 2.0. Hence, in addition, for the M_s -range case, the robustness assessment for the M_s is also presented so that the user can calculate in advance the maximum sensitivity that will be obtained.

In order to obtain the performance assessment rules, the IAE has been calculated for all the considered normalized dead times and maximum sensitivities. The obtained IAE values in the M_s -range and in the M_s -value case has been calculated by using a fitting technique with suitable functions. Also, the obtained M_s in the case of M_s -range has been fitting as a function of the normalized dead time. The obtained results can be expressed as

$$IAE_{ld} = KT (a_1\tau + a_2), \quad (27)$$

$$IAE_{sp} = KT (b_1\tau + b_2), \quad (28)$$

$$M_s = c_1\tau^3 + c_2\tau^2 + c_3\tau + c_4, \quad (29)$$

where the values of the coefficients to be employed are shown in Tables 9 and 10.

The optimal values of IAE_{ld} , IAE_{sp} for the M_s -range case, along with the M_s values and the corresponding fitting functions are reported, for the different values of τ , in Figure 7. Finally, in Figure 8 the same results are illustrated for the M_s -value case where, for the sake of comparison, also the M_s -range results are plotted. By looking at the results in Figure 8, it can be noticed that, after a certain limit that is outlined by the results for the M_s - range case, there is no point in increasing the M_s , because the performance improvement is not significant compared to the loss of robustness. This behavior is even more evident for low time delays and, for the set-point tracking

Table 9: Performance assessment function parameters for IAE in load disturbance rejection and set-point following task.

	a_1	a_2	b_1	b_2
M_s -range	1.4184	-0.1046	1.3731	0.0687
$M_s = 1.4$	1.8603	-0.0537	1.8413	-0.0153
$M_s = 1.5$	1.6148	-0.0508	1.5808	0.0742
$M_s = 1.6$	1.5670	-0.0902	1.5169	0.0613
$M_s = 1.7$	1.4290	-0.0740	1.3793	0.0888
$M_s = 1.8$	1.3217	-0.0521	1.2900	0.1286
$M_s = 1.9$	1.2872	-0.0673	1.2675	0.1541
$M_s = 2.0$	1.2644	-0.0978	1.2734	0.1433

Table 10: Robustness assessment function parameters for M_s -range in load disturbance rejection and set-point following task.

	c_1	c_2	c_3	c_4
M_s -range	0.0096	-0.0705	0.1543	1.5797

task, a performance loss is eventually obtained by increasing too much the M_s . Finally, another effect of this behavior can be appreciated in Figure 7, where the optimal M_s indeed exhibits a decrement for low dead times.

7. Simulation results

In order to verify the effectiveness of the proposed tuning rules, different illustrative simulation results are presented in this section. Moreover, for the sake of comparison, other methods proposed in the literature will be also considered. In particular:

- the optimal tuning rules for FOPID controllers proposed in [25]. These tuning rules are denoted as SP or LD (which means that the set-point following or the load disturbance re-

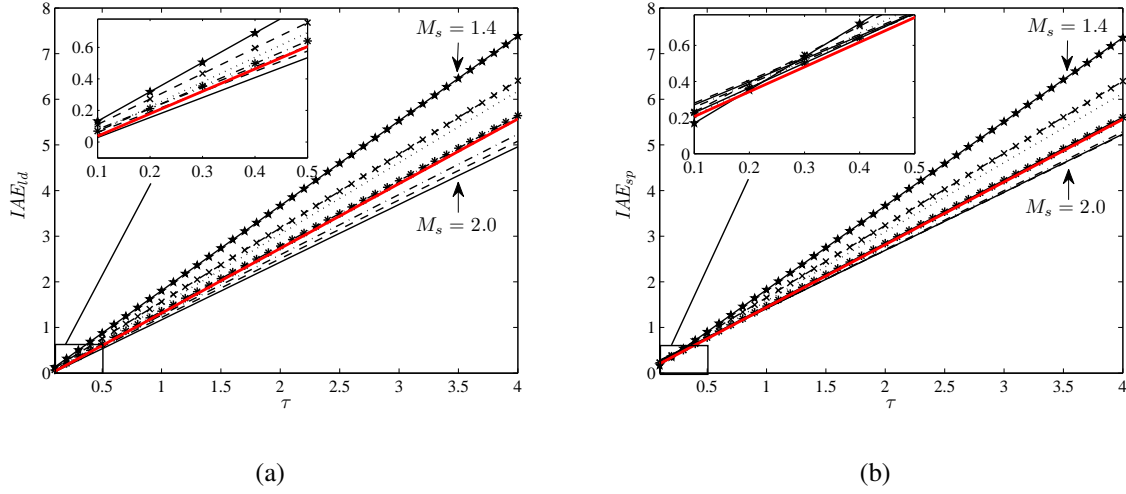


Figure 8: Performance assessment index function for the M_s -value case. *Red line*: Performance assessment index for M_s -range case.

jection task are optimized, respectively) followed by the target maximum sensitivity (which can be 1.4 or 2.0) and the letter F (which means that the controller is of fractional order). These rules have been obtained by minimizing the IAE with a constraint on the maximum sensitivity, but the set-point following and the load disturbance rejection tasks have been considered *separately*. Hence, these rules represent the anchor points of the PFs. Thus, it is worth considering them in order to underlying the benefits of a balanced tuning that considers both tasks at once;

- the intermediate tuning rules for integer order PID controller design proposed in [26], denoted by $\alpha = 0.50$, which means that the load disturbance and the set-point responses are considered to be equally important. These rules, in general, lead to high values of M_s . Indeed, the pure IAE optimization is not sufficient to achieve good level of robustness. Hence, they have been considered in the comparison in order to show the need of constraining the value of M_s also in a balanced tuning framework.

Optimal tuning from [25] are considered because they provide the extremum optimal cases but just considering one operational task for the control system. Therefore the balanced tuning ben-

efits should be faced against the eventual poor performance when the loop operates on the task it was not designed for and, also with the optimal one to evaluate the level of optimality achieved by the balanced tuning. On the other hand, the tuning proposed in [26] has been chosen because it follows similar spirit as the ones proposed here but without the MOO framework and without considering robustness. Therefore some advantages of the current proposal are to be advised. In order to thoroughly test the proposed method, three kinds of processes will be considered: a FOPDT process (with different normalized dead times), a high-order process and a non-minimum-phase process. In all cases, the processes are represented by FOPTD models with different instances for the value of the normalized dead-time and also representing different kind of plant-model mismatch (higher order and non-minimum phase dynamics). For each process and for each tuning, with the exception of the rules proposed in [26], a global performance index (GPI) has been computed by considering the obtained IAE in both the servo and the regulatory tasks, the M_s and the Total Variation (TV) of the control variable for each task. Indeed, although it is clear that reducing the M_s (when considering time delay processes) means also reducing the bandwidth, hence the control effort, a direct evaluation of the control effort in terms of TV is interesting.

The proposed global performance index is computed by normalizing each performance index (IAE(sp), IAE(ld), TV(sp), TV(ld) and M_s) against its highest value, obtained by ranging over the considered set of tuning rules. Then, the normalized indexes are summed up and the result is eventually divided by the number of considered performance indexes (5 in this case). In this way, the global performance index is always between 0 and 1, where 1 is the worst possible tuning and 0 is the utopia point.

Roughly speaking, the GPI is a compact index that synthesizes the radar plot information. Hence, exactly as for the radar plot, it does not have an absolute meaning; rather, it can be used to evaluate a given approach against a selected benchmark tuning rules. In this context, the GPI and the radar plot are only provided for the proposed rules and the ones of [25], aiming at evaluating the NS against the PF anchor points. On the contrary, the rules proposed in [26] have such a different behavior that it can be appreciated directly from the tables with results and from plot of the responses. Moreover, they would flat down all the differences between the proposed balance tuning and the anchor points (recall that the GPI is relative index) and for this reason they have not been

included.

7.1. FOPDT processes

The following FOPDT processes are considered, where $K = 1$ and $T = 1$:

$$P_1(s) = \frac{1}{s+1}e^{-0.67s}, \quad \tau = 0.67, \quad (30)$$

$$P_2(s) = \frac{1}{s+1}e^{-2.5s}, \quad \tau = 2.5. \quad (31)$$

First, the tuning rules for the M_s -range and M_s -value have been applied to the process (30) and the resulting values of the controller parameters, of the integrated absolute errors, the total variations and the maximum sensitivity for the different cases are shown in Table 11. The process responses and the control variables for the different cases and for both the set-point following and load disturbance rejection tasks are plotted in Figure 9. In particular, the M_s -range case is considered with the M_s -value cases for $M_s = 1.4$ and $M_s = 2.0$. In these latter two cases the step responses obtained with the tuning rules proposed in this paper (solid line) are compared with those obtained by applying correctly the tuning rules specifically devised in [25] for a single task (dashed line) and with those that are obtained by inverting the use of them, that is, the tuning rule devised for the load disturbance is applied to the set-point following task and vice versa (dotted line). In this way, the balancing of the tuning between the two tasks can be clearly seen. Moreover, the performance obtained with the considered tuning rules is compared in a synthetic way by means of the radar diagram in Figure 10 and the corresponding global performance indices are shown in Table 12. Finally, a deeper analysis has been carried out in order to show the effect of the selected M_s on the process response; the results are shown in Figure 11, where the physical meaning of the M_s choice appears.

Tuning rule	K_p	T_i	T_d	μ	λ	IAE_{sp}	IAE_{ld}	TV_{sp}	TV_{ld}	Ms
M_s -range	1.36	1.11	0.19	1.12	1	1.02	0.82	30.26	1.40	1.66
$M_s=1.4$	0.97	1.04	0.20	1.12	1	1.26	1.11	18.94	1.17	1.40
$M_s=2.0$	1.67	1.02	0.21	1.11	1	1.04	0.65	43.31	2.03	2.00
SP 1.4 F	0.83	0.98	0.22	1.2	1	1.26	1.20	41.20	1.14	1.42
SP 2.0 F	1.26	1.03	0.27	1.2	1	0.92	0.83	78.80	2.06	2.15
LD 1.4 F	0.61	0.54	0.33	1.2	1	1.38	1.12	29.71	1.38	1.44
LD 2.0 F	0.91	0.52	0.38	1.1	1	1.15	0.70	24.50	1.65	1.95
$\alpha = 0.50$	1.78	0.80	0.35	1	1	1.24	0.58	41.07	3.43	3.26

Table 11: Results related to $P_1(s)$ ($\tau = 0.67$).

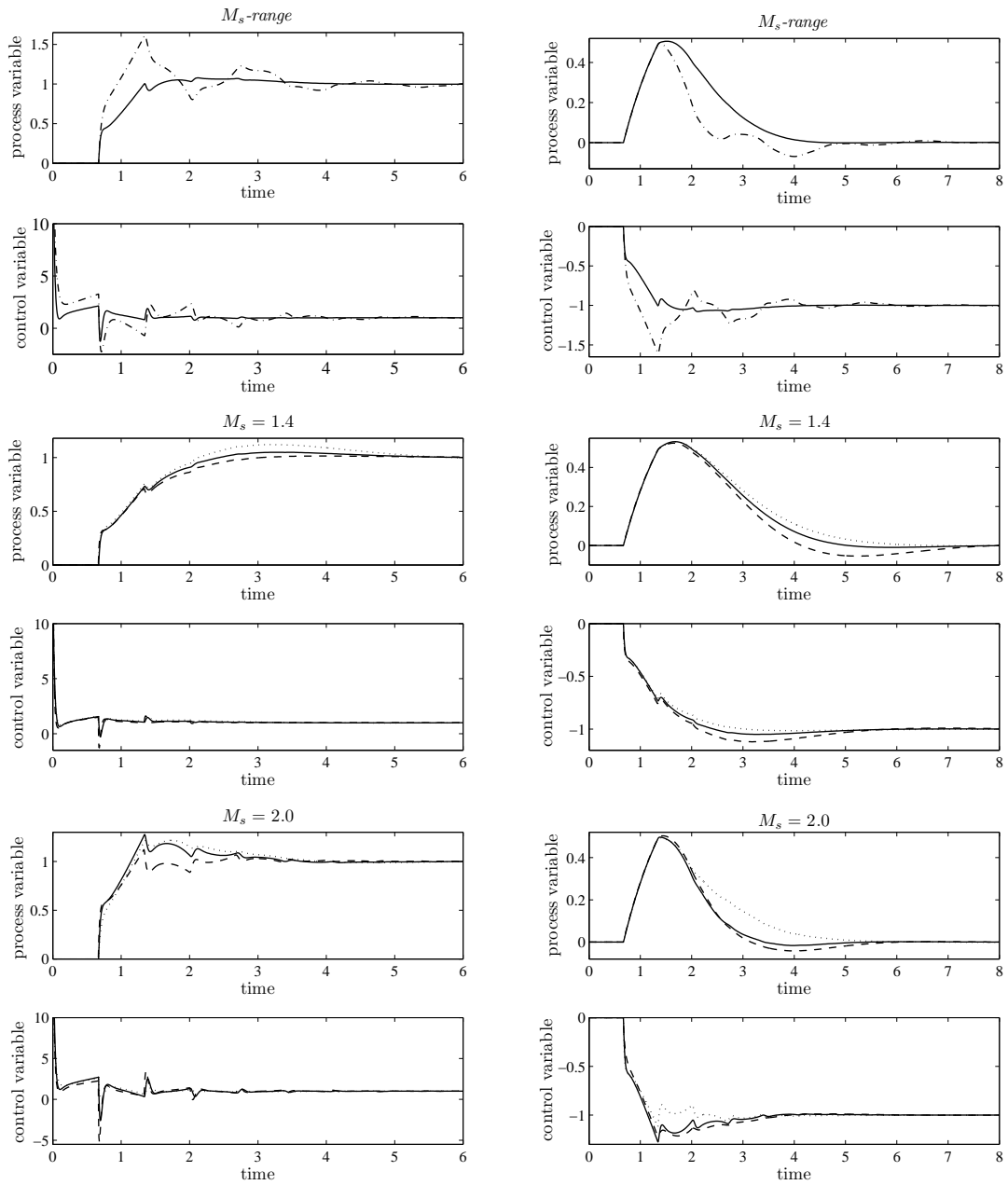


Figure 9: Set-point and load disturbance step responses for $P_1(s)$. Solid line: proposed tuning rules for FOPID controllers. Dash-dot line: tuning rules for PID controllers proposed in [26]. Dashed line: tuning rules for FOPID controllers proposed in [25]. Dotted line: tuning rules for FOPID controllers proposed in [25] used for the other control task they have been devised.

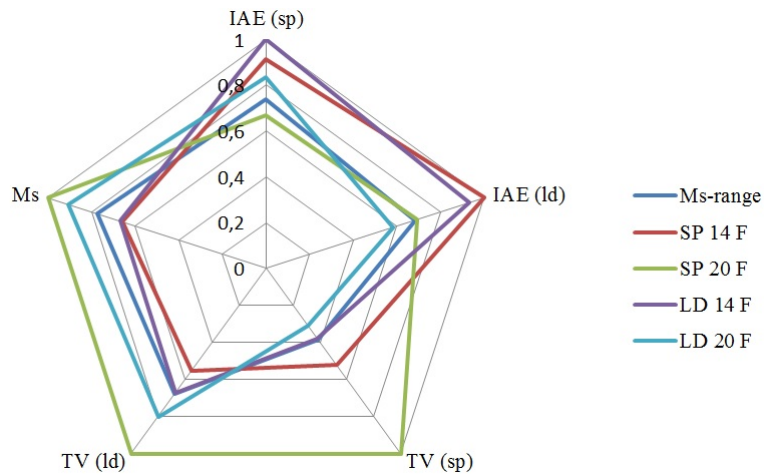


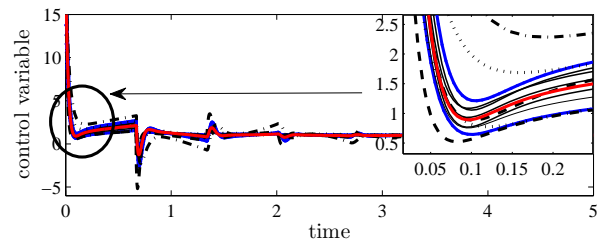
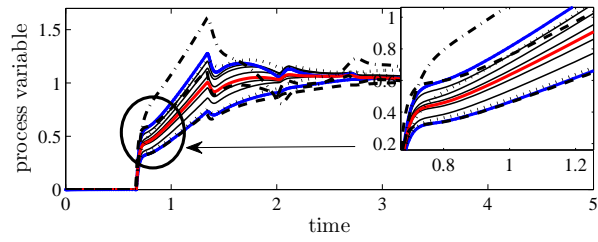
Figure 10: Radar diagram for $P_1(s)$

Tuning rule	GPI
M_s -range	0.65
SP 1.4 F	0.73
SP 2.0 F	0.87
LD 1.4 F	0.73
LD 2.0 F	0.69

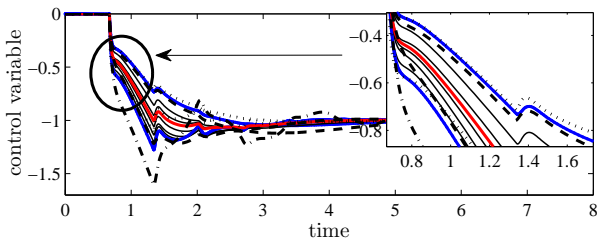
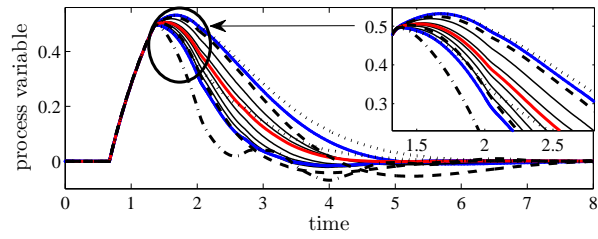
Table 12: Global performance index for each tuning of $P_1(s)$

Results related to the other FOPDT process (31) are shown in Table 13. The process responses and the control variables for both the set-point following and load disturbance rejection tasks are plotted in Figure 12. Further, the performance obtained with the considered tuning rules is compared by means of the radar diagram in Figure 13 and the corresponding global performance indices are shown in Table 14.

A deeper analysis has been then carried out again in order to show the effect of the selected M_s on the process response; the results are shown in Figure 14, where the physical meaning of the M_s choice appears.



(a)



(b)

Figure 11: Step responses for $P_1(s)$. Comparison between the tuning rules for M_s -range and M_s -value case and the tuning rules proposed in [25]. (a) Set-point following task. (b) Load disturbance rejection task. Solid red line: M_s -range tuning. Solid blue lines: boundaries of the M_s -value tuning. Dashed line: optimal tuning rules proposed in [25]. Dotted line: response obtained using the tuning proposed in [25] for opposite operation mode. Dash-dot line: intermediate tuning rules proposed in [26].

Tuning rule	K_p	T_i	T_d	μ	λ	IAE_{sp}	IAE_{ld}	TV_{sp}	TV_{ld}	Ms
M_s -range	0.57	1.95	0.62	1.06	1	3.46	3.44	11.17	1.42	1.68
$M_s=1.4$	0.37	1.63	0.70	1	1	4.51	4.50	5.21	1.07	1.41
$M_s=2.0$	0.69	2.01	0.66	1.03	1	3.32	2.99	14.03	1.94	2.00
SP 1.4 F	0.21	0.96	1.07	1.2	1	4.81	4.80	9.71	1.09	1.38
SP 2.0 F	0.35	1.03	1.31	1.2	1	3.40	3.27	19.33	1.94	2.08
LD 1.4 F	0.26	1.08	0.93	1.2	1	4.44	4.42	12.26	1.13	1.44
LD 2.0 F	0.53	1.40	0.90	1.2	1	3.40	2.96	32.59	2.57	2.34
$\alpha = 0.50$	not applicable									

Table 13: Results related to $P_2(s)$ ($\tau = 2.5$).

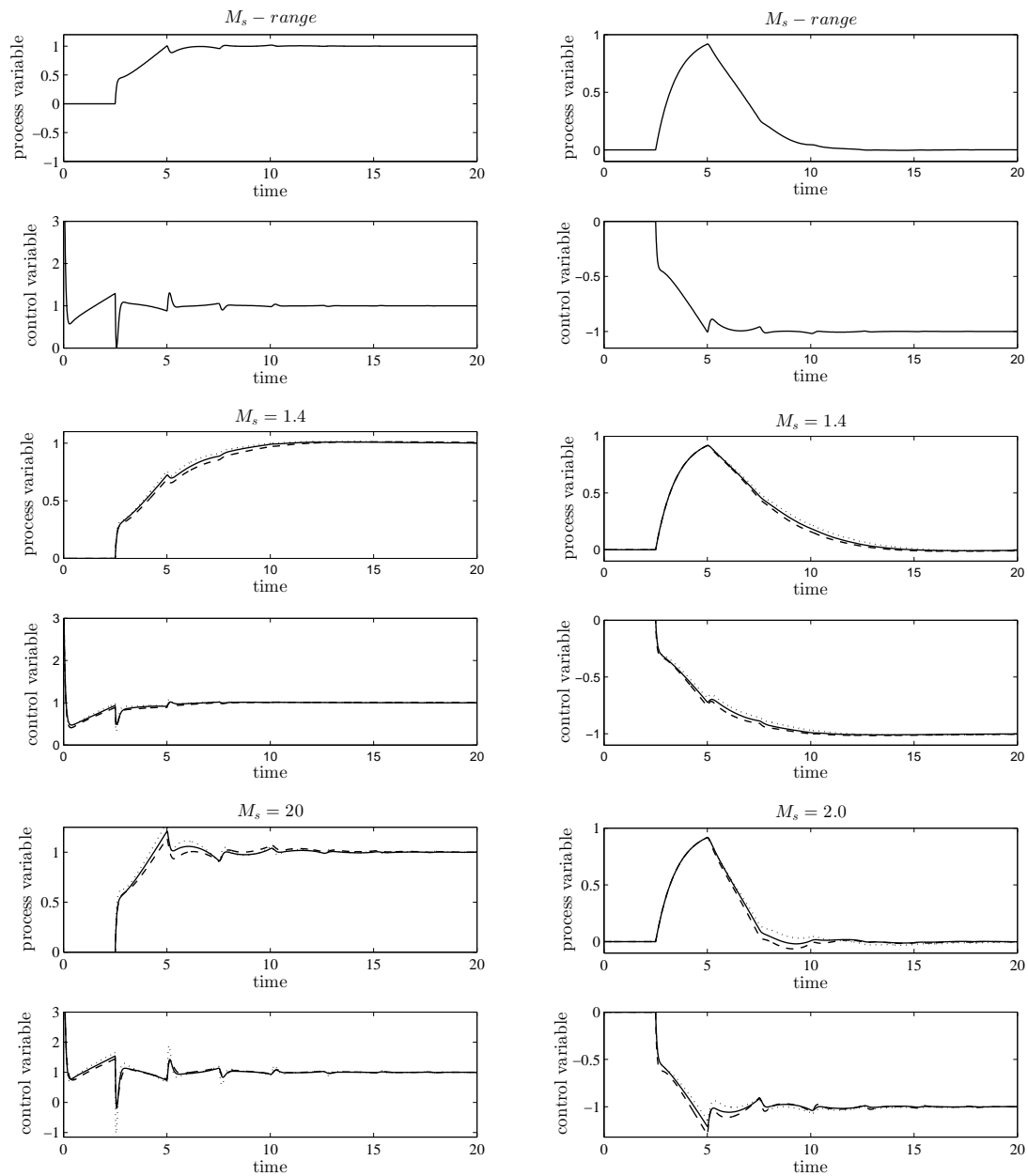


Figure 12: Set-point and load disturbance step responses for $P_2(s)$. Solid line: proposed tuning rules for FOPID controllers. Dash-dot line: tuning rules for PID controllers proposed in [26]. Dashed line: tuning rules for FOPID controllers proposed in [25]. Dotted line: tuning rules for FOPID controllers proposed in [25] used for the other control task they have been devised.

Tuning rule	GPI
M_s -range	0.61
SP 1.4 F	0.66
SP 2.0 F	0.72
LD 1.4 F	0.66
LD 2.0 F	0.86

Table 14: Performance index for each tuning of $P_2(s)$

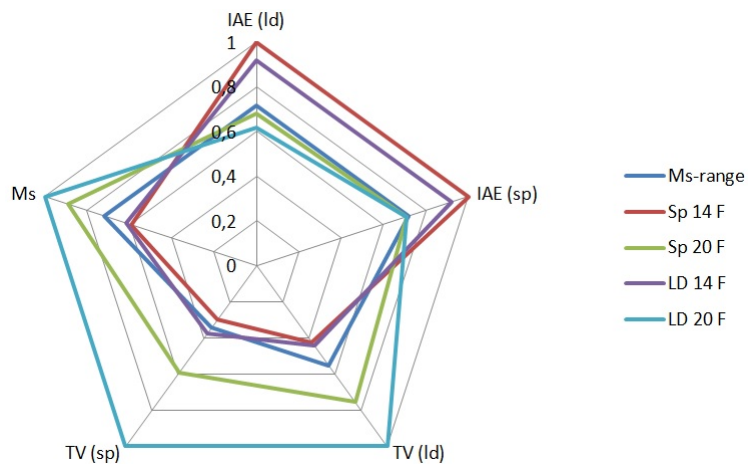
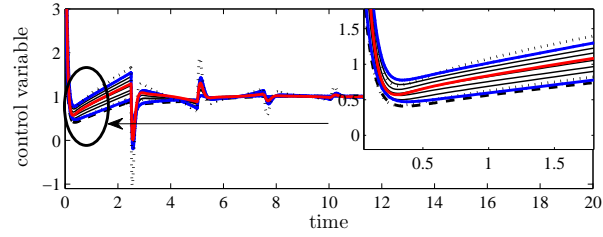
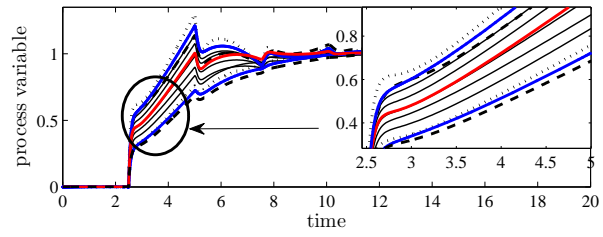
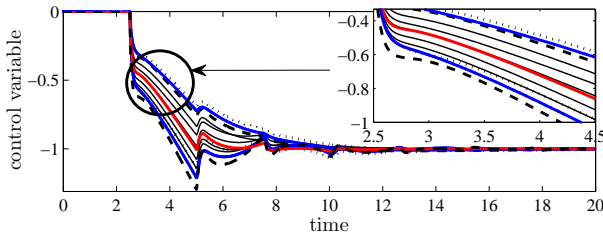
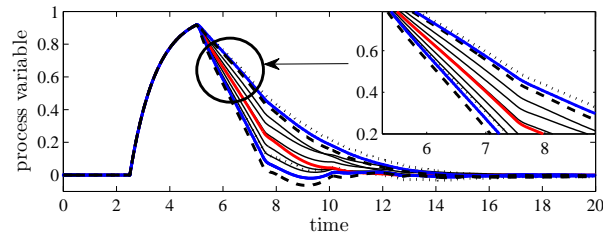


Figure 13: Radar diagram for $P_2(s)$



(a)



(b)

Figure 14: Step responses for $P_2(s)$. Comparison between the tuning rules for M_s -range and M_s -value case and the tuning rules proposed in [25]. (a) Set-point following task. (b) Load disturbance rejection task. Solid red line: M_s -range tuning. Solid blue lines: boundaries of the M_s -value tuning. Dashed line: optimal tuning rules proposed in [25]. Dotted line: response obtained using the tuning proposed in [25] for opposite operation mode.

From the obtained results it can be appreciated that the proposed approach is effective and that

the obtained results are very close and sometimes better in both the objectives compared to the optimal ones achieved in [25]. Nevertheless, from the comparison with the tuning rules specifically devised in [25] for a single task (dashed line) and with those that are obtained by inverting the use of them, the balancing of the tuning between the two tasks can be clearly seen.

7.2. High-order process

As another example, in order to verify the robustness of the proposed rules, the following high-order process has been considered:

$$P_3(s) = \frac{1}{(s + 1)^8}. \quad (32)$$

In order to apply the tuning rules originally devised for FOPID controller, the process has been modelled as a FOPDT process with $K = 1$, $T = 3.06$, $L = 4.95$ and $\tau = 1.62$ (note that the process is therefore dead time dominant). Then, the optimal tuning rules have been applied. The results obtained are shown in Table 15 and in Figure 15. As for the previous process, the radar diagram and the global performance index for the different considered tuning rules have been determined. They are shown in Figure 16 and in Table 16, respectively. From the obtained results it can be appreciated that the proposed approach gives satisfactory performance and the sensitivity values are acceptable

7.3. Non-minimum-phase process

Finally, a non-minimum-phase process is considered:

$$P_4(s) = \frac{1 - s}{(s + 1)^3}. \quad (33)$$

Again, the process has been modeled as a FOPDT process with $K = 1$, $T = 1.62$, $L = 2.39$ and $\tau = 1.48$. Then, the optimal tuning rules have been applied. Results are shown in Table 17, the responses are plotted in Figure 17, the radar diagram is in Figure 18 and the global performance indices are in Table 18.

Tuning rules	K_p	T_i	T_d	μ	λ	IAE_{sp}	IAE_{ld}	TV_{sp}	TV_{ld}	M_s
M_s -range	0.73	4.87	1.46	1.09	1	7.42	6.98	9.81	1.16	1.68
$M_s=1.4$	0.49	4.17	1.53	1.06	1	8.91	8.79	6.18	1.04	1.42
$M_s=2.0$	0.88	4.83	1.55	1.07	1	7.29	6.36	11.31	1.38	1.98
SP 1.4 F	0.34	2.96	2.43	1.2	1	9.25	9.20	17.10	1.02	1.40
SP 2.0 F	0.54	3.15	2.97	1.2	1	6.88	6.48	26.78	1.14	2.00
LD 1.4 F	0.33	2.63	2.53	1.2	1	8.95	8.81	16.56	1.06	1.44
LD 2.0 F	0.59	3.07	2.62	1.2	1	7.28	6.40	29.50	1.35	2.00
$\alpha = 0.50$	0.91	4.16	2.20	1	1	6.91	5.90	9.80	1.49	2.92

Table 15: Results related to the high-order process $P_3(s)$.

Tuning rule	GPI
M_s -range	0.72
SP 1.4 F	0.81
SP 2.0 F	0.84
LD 1.4 F	0.80
LD 2.0 F	0.89

Table 16: Performance index for each tuning for process $P_3(s)$.

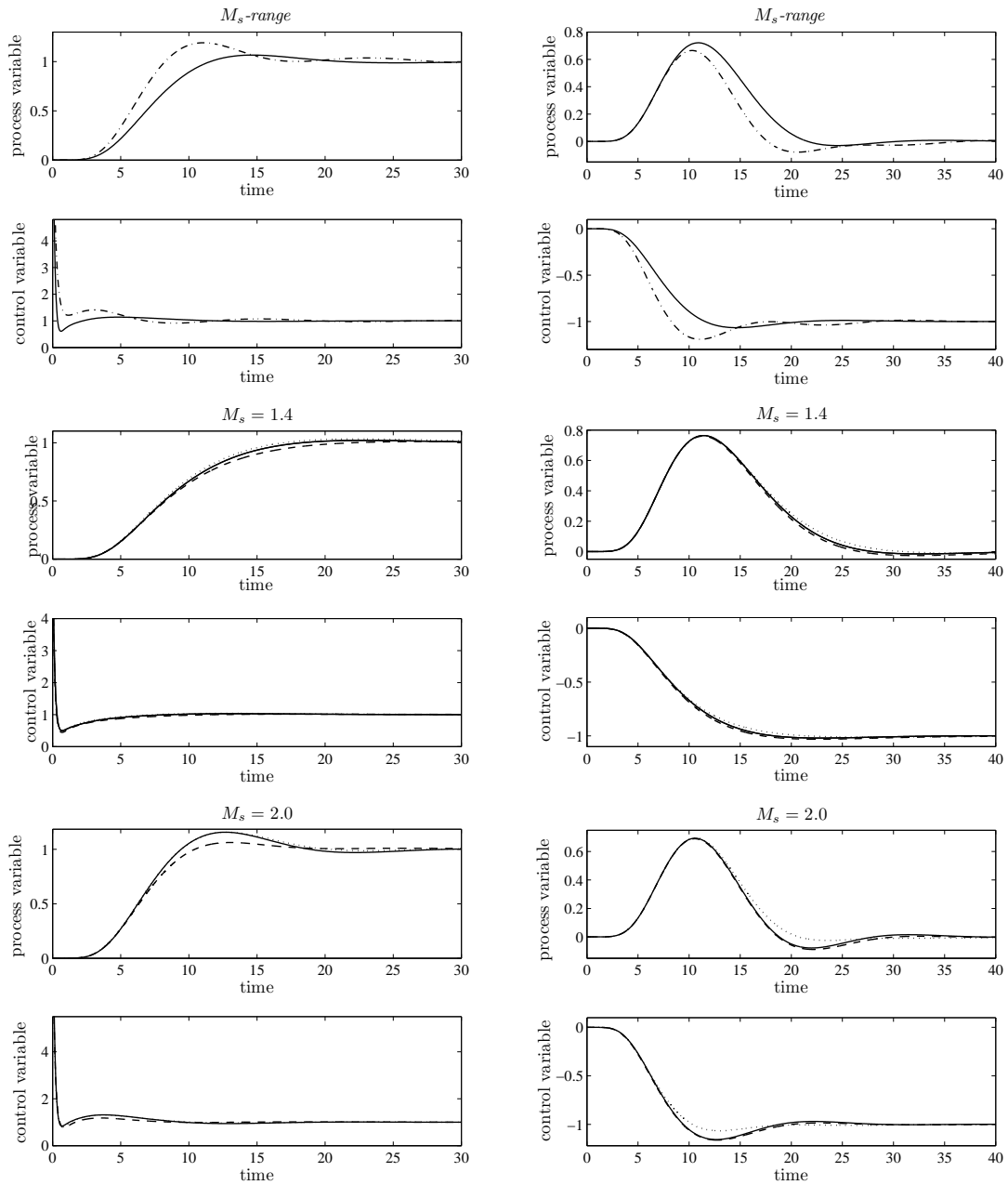


Figure 15: Set-point and load disturbance step responses for $P_3(s)$. Solid line: proposed tuning rules for FOPID controllers. Dash-dot line: tuning rules for PID controllers proposed in [26]. Dashed line: tuning rules for FOPID controllers proposed in [25]. Dotted line: tuning rules for FOPID controllers proposed in [25] used for the other control task they have been devised.

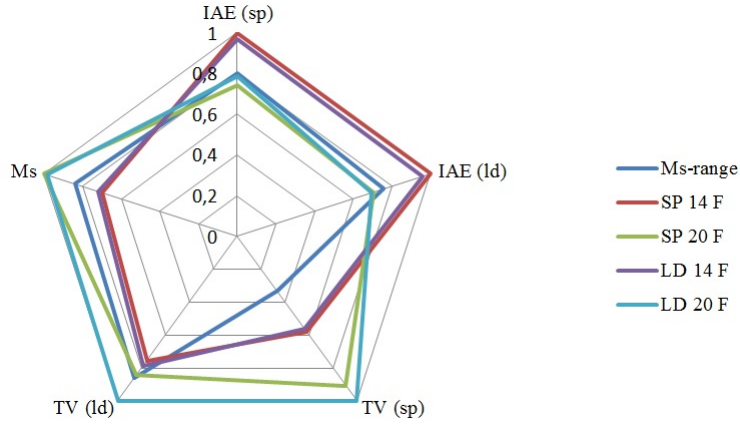


Figure 16: Radar diagram for the high-order process $P_3(s)$

Tuning rules	K_p	T_i	T_d	μ	λ	IAE_{sp}	IAE_{ld}	TV_{sp}	TV_{ld}	M_s
M_s -range	0.78	2.47	0.68	1.10	1	3.34	3.45	9.91	1.39	1.68
$M_s=1.4$	0.52	2.13	0.72	1.07	1	4.29	4.46	6.42	1.26	1.41
$M_s=2.0$	0.93	2.43	0.73	1.08	1	3.04	3.00	11.13	1.55	1.97
SP 1.4 F	0.37	1.57	1.01	1.2	1	4.46	4.66	16.11	1.24	1.39
SP 2.0 F	0.59	1.67	1.24	1.2	1	3.15	3.32	25.01	1.47	2.05
LD 1.4 F	0.35	1.33	1.10	1.2	1	4.32	4.45	15.16	1.30	1.43
LD 2.0 F	0.61	1.51	1.10	1.1	1	3.09	2.91	11.86	1.61	2.16
$\alpha = 0.50$	0.97	2.09	1.09	1	1	2.78	2.87	10.60	1.85	2.90

Table 17: Results related to the non-minimum-phase process $P_4(s)$.

Tuning rule	GPI
M_s -range	0.71
SP 1.4 F	0.81
SP 2.0 F	0.86
LD 1.4 F	0.80
LD 2.0 F	0.76

Table 18: Performance index for each tuning for process $P_4(s)$

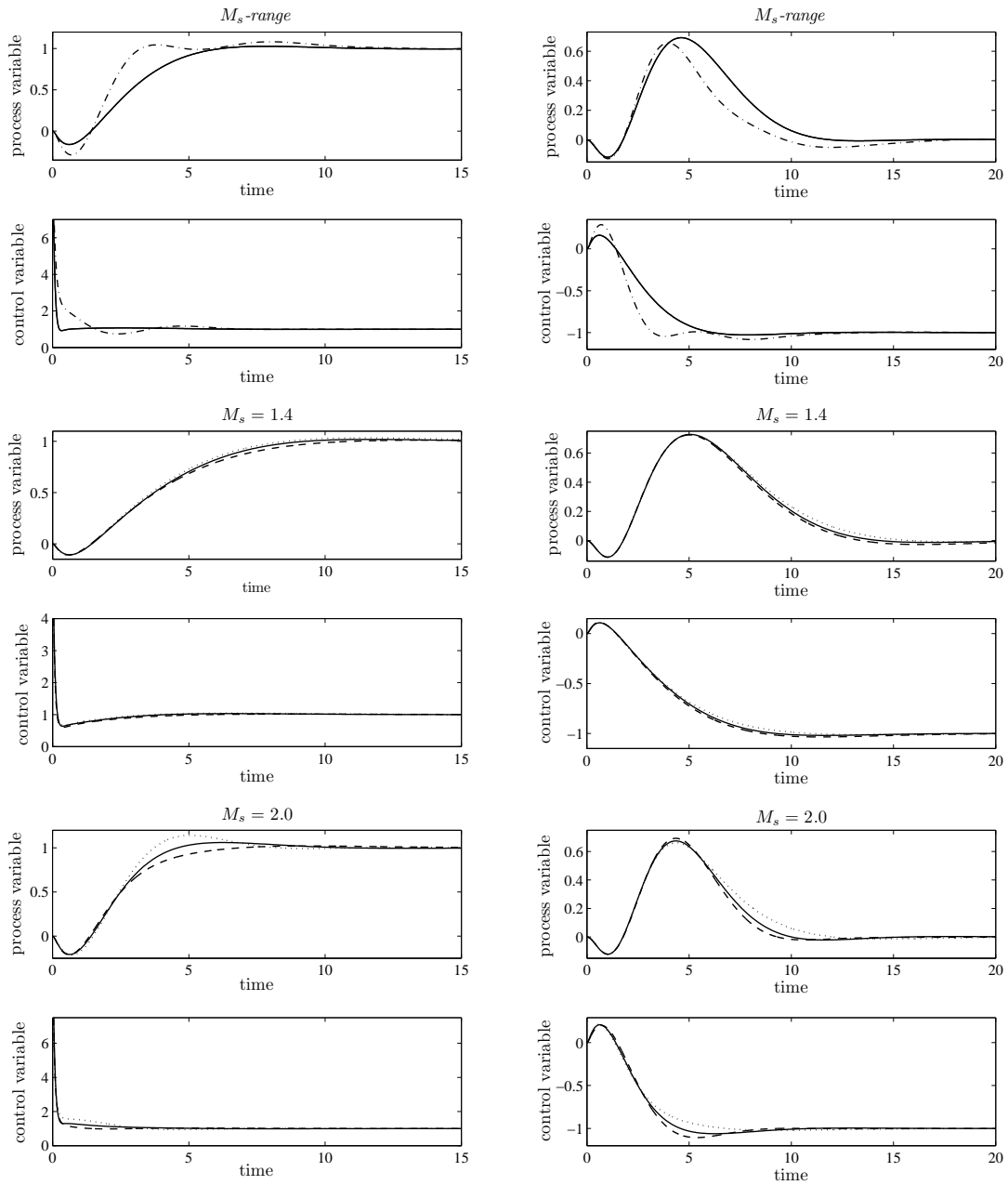


Figure 17: Set-point and load disturbance step responses for $P_4(s)$. Solid line: proposed tuning rules for FOPID controllers. Dash-dot line: tuning rules for PID controllers proposed in [26]. Dashed line: tuning rules for FOPID controllers proposed in [25]. Dotted line: tuning rules for FOPID controllers proposed in [25] used for the other control task they have been devised.

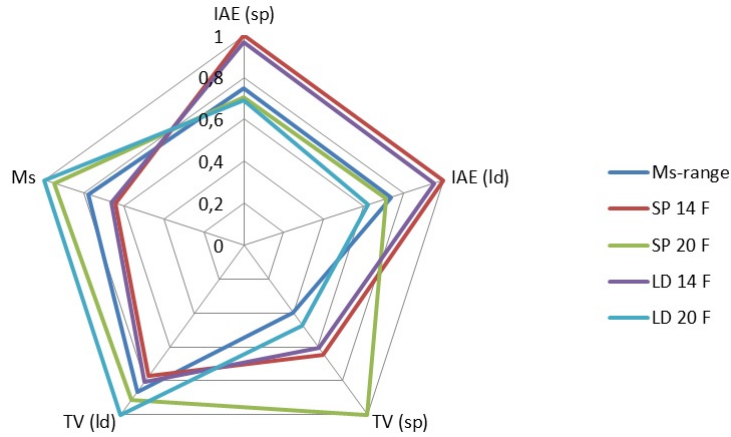


Figure 18: Radar diagram for the non-minimum-phase process $P_4(s)$

7.4. Discussion

According to the simulation results, it can be seen that the intermediate tuning rules for PID controllers proposed in [26] achieve the best IAE performance for all the simulated processes for the load disturbance rejection and also the best IAE performance for set-point step response for $P_4(s)$. However, this comes at the expense of a higher value of the maximum sensitivity and therefore of a higher control action (indeed, as already mentioned, the maximum sensitivity has not been taken into account in the development of the intermediate tuning rules proposed in [26]), as it can be observed by looking at the step responses. On the contrary, the proposed tuning rules for the M_s -range case ensure a medium-high robustness level ($M_s \approx 1.7$) and also a smoother response for both operation modes (less oscillations in the output response and a smoother control variable) in all the examples.

For the cases of $M_s = 1.4$ and $M_s = 2.0$, it can be noticed that using the tuning rules for the M_s -value case, in all the processes the set-point step response shows an overshoot, but not significantly higher than the one obtained with the rules for FOPID controllers proposed in [25]. However, in the load disturbance step response, the rules for the M_s -value cases provide less oscillations in comparison with the rules proposed in [25]. Furthermore, it can be observed that, for $P_3(s)$ and $P_4(s)$, the optimal values of each operation mode with the rules in [25] are similar to those obtained with the M_s -value rules.

In general, it appears that the proposed rules are effective, especially considering their simplicity:

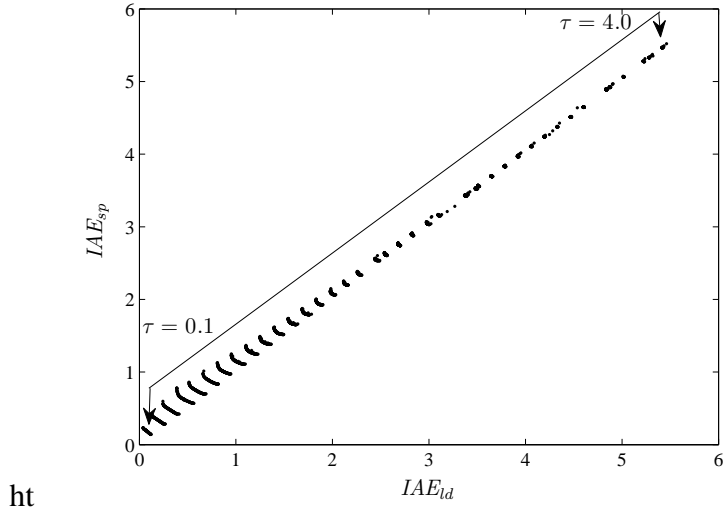


Figure 19: The PF for the M_s -valued case, $M_s = 1.7$.

indeed, the user is not forced to decide which task (servo or regulatory) is more important. Indeed, both operation modes are balanced in an optimal way by choosing the Nash solution.

It is also interesting to note that the value of M_s obtained by using the M_s -range tuning is quite close to 1.7, especially for processes with $\tau > 1$ (see Figure 7). Indeed, the NS in the M_s -range case is optimally balanced between three objectives, namely the set-point following, the load disturbance rejection and the robustness. By considering the results in Figure 8, as well as the simulation results, it turns out that there is no point in further increasing the value of M_s after 1.7 since the performance improvements are minimal in spite of the loss of robustness. As a final consideration, it is interesting to note that, for the normalized dead time greater than 2 ($\tau = 2$) the PF curve (surface, in the M_s -range case) approximation tends to collapse into a single point (curve, in the M_s -range case) as it can be observed in Figure 19, where only the case $M_s=1.7$ is considered for the sake of readability (but the other cases lead to similar results). Hence, the trade-off between load disturbance rejection and set-point following task is no longer appreciable for high delays and the optimal tuning for pure load disturbance rejection tends to coincide with the one for pure set-point following.

8. Conclusions

A set of optimally balanced tuning rules for FOPID controllers has been presented in this paper. Based on a MOOD procedure, the problem of tuning the FOPID controller with several requirements is addressed. Such procedure implemented a multi-stage approach into a MOO process in order to improve the convergence properties for the PF approximation. The primary goal of this work has been to minimize the IAE for either the load disturbance rejection task and the set-point following task with a constraint on the maximum sensitivity.

The NS criterion is applied to select a preferable solution among the PF approximation and finally the tuning rules for the FOPID controller have been obtained by fitting the optimal results.

The obtained rules have the valuable features to be able to take into account at once both the servo and the regulatory modes in an optimal way. Moreover, the user can select the desired level of robustness or keep it between given bounds depending on his/her preferences.

The performance assessment for both cases has been presented. This allows the designer to know in advance the performance index he/she will obtain and evaluate if the performance obtained with an existing controller can be improved.

Acknowledgment

This work was partially supported by the Spanish Ministry of Economy and Competitiveness program under grant DPI2013-47825-C3-1-R.

Appendix A

In this work, the optimal parameters proposed in [25] were used in the framework to solve the MOO problem. This tuning rules have the following structure:

$$\tau = \frac{L}{L + T}, \quad (34)$$

$$K_p = \frac{1}{K} (a\tau^b + c), \quad (35)$$

$$T_i = T^\lambda \left(a \left(\frac{L}{T} \right)^b + c \right), \quad (36)$$

$$T_d = T^\mu \left(a \left(\frac{L}{T} \right)^b + c \right), \quad (37)$$

where the values of a , b and c are shown in Tables 19-21 . The tuning rules for λ and μ are shown in Table 22.

Table 19: K_p tuning rule parameters for a FOPID controller.

Control task	$M_s = 1.4$			$M_s = 2.0$		
	a	b	c	a	b	c
set-point	0.6846	-0.9166	-0.7096	0.9294	-0.933	-0.9205
load disturbance	0.2776	-1.095	-0.1426	0.164	-1.449	0.2108

Table 20: T_i tuning rule parameters for a FOPID controller.

Control task	$M_s = 1.4$			$M_s = 2.0$		
	a	b	c	a	b	c
set-point	0.04701	-0.2611	0.9276	-0.001427	-1.003	1.031
load disturbance	0.6241	0.5573	0.0442	0.6426	0.8069	0.05627

Table 21: T_d tuning rule parameters for a FOPID controller.

Control task	$M_s = 1.4$			$M_s = 2.0$		
	a	b	c	a	b	c
set-point	0.3563	1.2	0.0003108	0.4203	1.229	0.01822
load disturbance	0.4793	0.7469	-0.02393	0.597	0.5568	-0.09536

Table 22: λ and μ tuning rule parameters for a FOPID controller.

Control task	$M_s = 1.4$		$M_s = 2.0$	
	λ	μ	λ	μ
set-point	1	1.1 if $\tau < 0.1$ 1.2 if $0.1 \leq \tau$	1	1.0 if $\tau < 0.1$ 1.1 if $0.1 \leq \tau < 0.4$ 1.2 if $0.4 \leq \tau$
load disturbance	1	1.0 if $\tau < 0.1$ 1.1 if $0.1 \leq \tau < 0.4$ 1.2 if $0.4 \leq \tau$	1	1.0 if $\tau < 0.2$ 1.1 if $0.2 \leq \tau < 0.6$ 1.2 if $0.6 \leq \tau$

References

- [1] K. J. Åström, T. Hägglund, *Advanced PID Control*, ISA, Research Triangle Park, NJ, 2006.
- [2] A. Visioli, *Practical PID Control.*, Springer, London (UK), 2006.
- [3] Y. Q. Chen, Ubiquitous fractional order controls?, in: *Proceedings of the 4th IFAC Workshop of Fractional Differentiation and its Applications*, Porto (P), 2006, pp. 481–492.
- [4] Y. Q. Chen, I. Petras, D. Xue, Fractional order control - a tutorial, in: *Proceedings of the American Control Conference*, St. Louis (MO), 2009.
- [5] C. A. Monje, Y. Chen, B. M. Vinagre, D. Xue, V. Feliu, *Fractional-order Systems and Controls: Fundamentals and Applications*, Springer-Verlag, London (UK), 2010.
- [6] A. Oustaloup, *La Commande CRONE: Commande Robuste d'Ordre Non Entier*, Hermes, Paris (F), 1991.
- [7] A. Oustaloup, P. Lanusse, P. Melchior, X. Moreau, J. Sabatier, The CRONE approach: theoretical developments and major applications, in: *Proceedings of the 2nd IFAC Workshop on Fractional Differentiation and its Applications*, Porto (P), 2006, pp. 324–354.
- [8] A. Oustaloup, J. Sabatier, P. Lanusse, R. Malti, P. Melchior, X. Moreau, M. Moze, An overview of the crone approach in system analysis, modeling and identification, observation and control, in: *Proceedings of the 17th IFAC World Congress*, Seoul (Korea), 2008, pp. 14254–14265.
- [9] C. A. Monje, B. M. Vinagre, V. Feliu, Y. Q. Chen, Tuning and auto-tuning of fractional order controllers for industry applications, *Control Engineering Practice* 16 (7) (2008) 798–812.
- [10] D. Valerio, J. Sá da Costa, Introduction to the single-input, single-output fractional control, *IET Control Theory and Applications* 5 (8) (2011) 1033–1057.
- [11] M. D. Ortigueira, *Fractional Calculus for Scientists and Engineers*, Springer, London(UK), 2011.
- [12] R. E. Gutierrez, J. M. Rosario, J. A. T. Machado, Fractional order calculus: Basic concepts and engineering applications, *Mathematical Problems in Engineering* 2010.
- [13] J. Sabatier, O. P. Agrawal, J. A. T. Machado, *Advances in Fractional Calculus: Theoretical Developments and Applications in Physics and Engineering*, Springer, London, (UK), 2007.
- [14] I. Podlubny, Fractional-order systems and $PI^{\lambda}D^{\mu}$ controllers, *IEEE Transactions on Automatic Control* 44 (1) (1999) 208–214.
- [15] B. M. Vinagre, C. A. Monje, A. J. Calderon, J. I. Suarez, Fractional PID controllers for industry application. a brief introduction., *Journal of Vibration and Control* 13 (2007) 1419–1429.
- [16] R. Caponetto, G. Dongola, L. Fortuna, A. Gallo, New results on the synthesis of FO-PID controllers, *Communications in Nonlinear Science and Numerical Simulation* 15 (4) (2010) 997–1007.
- [17] F. J. Castillo, V. Feliu, R. Rivas, L. Sanchez, Comparative analysis of stability and robustness between integer and fractional-order PI controllers for first order plus dead time plants, in: *Proceedings of the 18th IFAC World Congress*, Milan (I), 2011, pp. 15019–15024.

- [18] D. Valerio, J. Sá da Costa, A review of tuning methods for fractional PIDs, in: Preprints IFAC Workshop on Fractional Differentiation and its Applications, Badajoz (E), 2010.
- [19] R. S. Barbosa, J. A. Tenreiro Machado, I. M. Ferreira, Tuning of PID controllers based on Bode's ideal transfer function, *Nonlinear Dynamics* 38 (2004) 305–321.
- [20] C. A. Monje, B. M. Vinagre, A. J. Calderon, V. Feliu, Y. Q. Chen, On fractional PI^λ controllers: some tuning rules for robustness to plant uncertainties, *Nonlinear Dynamics* 38 (2004) 369–381.
- [21] D. Valerio, J. Sá da Costa, Tuning of fractional PID controllers with Ziegler-Nichols-type rules, *Signal Processing* 86 (2006) 2771–2784.
- [22] M. Beschi, F. Padula, A. Visioli, The generalised isodamping approach for robust fractional PID controllers design, *International Journal of Control* in press (2016) 369–381, DOI:10.1080/00207179.2015.1099076.
- [23] J. J. Gude, E. Kahoraho, New tuning rules for PI and fractional PI controllers, in: Proceedings European Control Conference, Budapest (HU), 2009.
- [24] Y. Q. Chen, T. Bhaskaran, D. Xue, Practical tuning rule development for fractional order proportional and integral controllers, *ASME Journal of Computational and Nonlinear Dynamics* 3 (2008) 0214031–0214037.
- [25] F. Padula, A. Visioli, Tuning rules for optimal PID and fractional-order PID controllers, *Journal of Process Control* 21 (1) (2011) 69–81.
- [26] O. Arrieta, A. Visioli, R. Vilanova, PID autotuning for weighted servo/regulation control operation, *Journal of Process Control* 20 (4) (2010) 472–480.
- [27] M. A. Martínez, J. Sanchis, X. Blasco, Multiobjective controller design handling human preferences, *Engineering Applications of Artificial Intelligence* 19 (8) (2006) 927–938.
- [28] G. Reynoso-Meza, J. Sanchis, X. Blasco, M. Martínez, Controller tuning using evolutionary multi-objective optimisation: current trends and applications, *Control Engineering Practice*.
- [29] Q. Gao, J. Chen, L. Wang, S. Xu, Y. Hou, Multiobjective optimization design of a fractional order PID controller for a gun control system, 2013, doi:10.1155/2013/907256, *The Scientific World Journal*, Hindawi Publishing Corporation.
- [30] I. Pan, S. Das, Chaotic multi-objective optimization based design of fractional order $PI^\lambda D^\mu$ controller in AVR system, *International Journal of Electrical Power & Energy Systems* 43 (1) (2012) 393–407.
- [31] L. Meng, D. Xue, Design of an optimal fractional-order PID controller using multi-objective GA optimization, in: Proceedings Chinese Control and Decision Conference, IEEE, 2009, pp. 3849–3853.
- [32] A. Hajiloo, N. Nariman-Zadeh, A. Moeini, Pareto optimal robust design of fractional-order PID controllers for systems with probabilistic uncertainties, *Mechatronics* 22 (6) (2012) 788–801.
- [33] R. Aumann, S. Hart, *Handbook of Game Theory with Economic Applications*, Elsevier, 1994.
- [34] F. Padula, A. Visioli, *Advances in Robust Fractional Control*, Springer, London (UK), 2014.
- [35] R. Malti, X. Moreau, F. Khemane, A. Oustaloup, Stability and resonance conditions of elementary fractional

- transfer functions, *Automatica* 47 (11) (2011) 2462–2467.
- [36] K. H. Ang, G. Chong, Y. Li, PID control system analysis, design, and technology, *IEEE Transactions on Control Systems Technologies* 13 (4) (2005) 559–576.
- [37] F. G. Shinskey, *Feedback controllers for the process industries*, McGraw-Hill Professional, 1994.
- [38] K. J. Åström, T. Hägglund, *PID controllers: theory, design and tuning*, ISA, Research Triangle Park, NJ, 1995.
- [39] K. M. Miettinen, *Nonlinear multiobjective optimization*, Kluwer Academic Publishers, Boston, 1998.
- [40] C. C. Coello, G. B. Lamont, D. A. V. Veldhuizen, Multi-criteria decision making, in: *Evolutionary algorithms for solving multi-objective problems*, genetic and evolutionary computation series, Springer, New York (NY), 2007, pp. 515–545.
- [41] G. Reynoso-Meza, J. Sanchis, X. Blasco, J. M. Herrero, Multiobjective evolutionary algorithms for multivariable PI controller tuning, *Expert Systems with Applications* 39 (2012) 7895–7907.
- [42] A. Messac, A. Ismail-Yahaya, C. Mattson, The normalized normal constraint method for generating the Pareto frontier, *Structural and Multidisciplinary Optimization* (25) (2003) 86 – 98.
- [43] S. Das, P. N. Suganthan, Differential evolution: A survey of the state-of-the-art, *IEEE Transactions on Evolutionary Computation* 15 (1) (2011) 4–31.
- [44] G. Reynoso-Meza, H. Sánchez, X. Blasco, R. Vilanova, Reliability based multiobjective optimization design procedure for PI controller tuning, in: *Proceedings of the 19th IFAC World Congress*, Cape Town (South Africa), 2014, pp. 10263–10268.
- [45] H. Sánchez, R. Vilanova, Nash-based criteria for selection of pareto optimal controller, in: *Proceedings of the 17th International Conference on System Theory, Control and Computing*, Sinaia (RO), 2013.
- [46] H. Sánchez, R. Vilanova, Optimality comparison of 2DoF PID implementations, in: *Proceedings of the 18th International Conference on System Theory, Control and Computing*, Sinaia (RO), 2014.
- [47] H. Sánchez, A. Visioli, R. Vilanova, Nash tuning for optimal balance of the servo/regulation operation in robust PID control, in: *Proceedings 23th Mediterranean Conference on Control and Automation*, IEEE, 2015, pp. 715–721.
- [48] R. Storn, K. Price, Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces, *Journal of global optimization* 11 (4) (1997) 341–359.
- [49] G. Reynoso-Meza, J. Sanchis, X. Blasco, S. García-Nieto, Physical programming for preference driven evolutionary multi-objective optimization, *Applied Soft Computing* 24 (2014) 341–362.
- [50] F. Padula, A. Visioli, Optimal tuning rules for proportional-integral-derivative and fractional-order proportional-integral-derivative controllers for integral and unstable processes, *IET Control Theory and Applications* 6 (6) (2012) 776–786.
- [51] M. Tavazoei, Notes on integral performance indices in fractional-order control systems, *Journal of Process Control* 20 (2010) 285–291.