

ON AUTO-TUNING OF FRACTIONAL ORDER
 $PI^\lambda D^\mu$ CONTROLLERS

Concepción A. Monje ^{*,1} Blas M. Vinagre ^{*}
Vicente Feliu ^{**} YangQuan Chen ^{***}

^{*} *Escuela de Ingenierías Industriales, Universidad de
Extremadura, Badajoz, Spain,
email: {cmonje,bvinagre}@unex.es*

^{**} *Escuela Técnica Superior de Ingenieros Industriales,
Universidad de Castilla-La Mancha, Ciudad Real, Spain,
email: Vicente.Feliu@uclm.es*

^{***} *CSOIS, Utah State University, Logan, Utah, USA,
email: yqchen@ece.usu.edu*

Abstract: In this paper, a method for the auto-tuning of fractional order $PI^\lambda D^\mu$ controllers using relay feedback tests is proposed. A design method for this kind of controllers is discussed, based on the magnitude and phase measurement of the plant to be controlled from relay feedback tests at a frequency of interest. Simple relationships among the parameters of the fractional controller are established and specifications such as the static error constant (k_{ss}), phase margin (φ_m) and gain crossover frequency (ω_c) can be fulfilled, with a robustness argument by inspecting the flatness of phase Bode plot of the controller. An illustrative example of application is presented to show the reliability and effectiveness of the method.

Keywords: Controller automatic tuning, relay feedback test, fractional order compensator, fractional order $PI^\lambda D^\mu$ controller, robust control.

1. INTRODUCTION

The basic concepts on auto-tuning control design were born over 1950 with the fundamental theoretical tools and formulations established in 1960. However, it was in the next decade, the 1970s, when a key method for auto-tuning was approached by K. J. Åström and Hägglund (see (Åström and Wittenmark, 1973) and (Åström and Hägglund, 1984)). This method consists of the use of a relay feedback test. During relay feedback tests, processes dynamics typically encountered in process control (e.g., delays) will usually generate sustaining limit cycle oscillations and sub-

sequently, the auto-tuner can identify one point on the Nyquist curve of the process from this experiment.

On the other hand, fractional order calculus and its potentials in many areas of science and engineering have started to get appreciated. In particular, many research efforts related to the applications of fractional controllers have touched various aspects of control analysis and synthesis. However, in practical industrial setting, a similar auto-tuning procedure of fractional order controllers is rarely found and in strong demand. Therefore, the theme of our work is to develop auto-tuning techniques for non-integer order controllers, using relay feedback tests. Our ultimate goal is to de-

¹ This work has been partially supported by the Spanish Research Grant 2PR02A024 (Junta de Extremadura)

velop a method to auto-tune a fractional $PI^\lambda D^\mu$ controller, formulated as:

$$C(s) = k_c x^\mu \left(\frac{\lambda_1 s + 1}{s} \right)^\lambda \left(\frac{\lambda_2 s + 1}{x \lambda_2 s + 1} \right)^\mu. \quad (1)$$

As can be observed, this controller has two different parts given by equations (2) and (3):

$$PI^\lambda(s) = \left(\frac{\lambda_1 s + 1}{s} \right)^\lambda, \quad (2)$$

$$PD^\mu(s) = k_c x^\mu \left(\frac{\lambda_2 s + 1}{x \lambda_2 s + 1} \right)^\mu. \quad (3)$$

Equation (2) corresponds to a fractional order PI^λ controller and equation (3) to a fractional order lead compensator that can be identified as a PD^μ controller plus a noise filter.

This paper is organized as follows. In section 2, the basic idea of relay tests is described. In section 3, the auto-tuning method proposed here is detailed, showing an illustrative example of application in section 4. Some conclusions are drawn in section 5.

2. RELAY TEST FOR AUTO-TUNING

The relay auto-tuning process has been widely used in industrial applications, as commented before (see (Hang *et al.*, 2002)). In this work we will use a variation of that scheme, shown in Fig. 1, in with a delay θ_a is introduced after the relay function. With this scheme, as explained in (Chen and Moore, 2005), the next relations are given:

$$\arg(G(j\omega_c)) = -\pi + \omega_c \theta_a, \quad (4)$$

$$|G(j\omega_c)| = \frac{\pi a}{4d} = \frac{1}{N(a)}. \quad (5)$$

where $G(j\omega_c)$ is the plant or process for which the controller has to be auto-tuned, d is the relay output, a is the amplitude of the output signal (signal "y" in Fig. 1), $N(a)$ is the equivalent relay gain, θ_a is the delay and ω_c the frequency of the output signal y corresponding to this delay θ_a .

This way, for each value of θ_a a different point on the Nyquist curve of the plant is obtained. Therefore, a point on the Nyquist curve of the plant at a particular desired frequency ω_c can be identified, for example, at the gain crossover frequency required for the controlled system. The problem would be how to select the right value of θ_a which corresponds to a specific frequency ω_c . An iterative method can be used to solve this problem as presented in (Chen and Moore, 2005). The

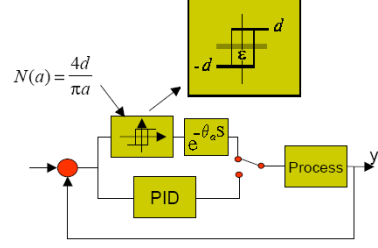


Fig. 1. Relay auto-tuning scheme with delay

artificial time delay parameter can be updated using the simple interpolation/extrapolation scheme $\theta_n = \frac{\omega_c - \omega_{n-1}}{\omega_{n-1} - \omega_{n-2}}(\theta_{n-1} - \theta_{n-2}) + \theta_{n-1}$, where n represents the current iteration number. With the new θ_n , after the relay test, the corresponding frequency ω_n can be recorded and compared with the frequency ω_c so that the iteration can continue or stop. Two initial values of the delay (θ_{-1} and θ_0) and their corresponding frequencies (ω_{-1} and ω_0) are needed to start the iteration. The pair (θ_{-1}, θ_0) can also be easily and automatically estimated by using a scheme similar to the one for θ_n .

In this work, this relay scheme is used for the auto-tuning of a fractional order $PI^\lambda D^\mu$ controller, as detailed next.

3. THE AUTO-TUNING METHOD PROPOSED

3.1 Introduction

As introduced before, there are a wide variety of auto-tuning methods for integer controllers. Some of them aim in someway the robustness of the controlled system (see (Tan *et al.*, 2002)), for example, forcing the phase of the open-loop system to be flat around the crossover frequency so that the system is robust to gain variations (see (Chen and Moore, 2005) and (Chen *et al.*, 2004)). However, the complexity of the equations relating the parameters of the controller increases when some kinds of robustness constraints are required for the controlled system. The implementation of this type of auto-tuning methods for industrial purposes will be really complicate since, in general, industrial devices such as a PLC cannot solve sets of complex nonlinear equations. To this respect, the auto-tuning method proposed here allows a flexible and direct selection of the parameters of the fractional order controller, using the relay test.

On the other hand, a wide range of processes commonly found in industry are characterized by gain or load changes. To obtain a system robust to these gain variations this method follows a flat phase robustness criterion. This way, a gain change will not affect the phase of the open

loop system, presenting the step responses of the system a constant overshoot to load changes. In the case of parametric variations different from gain variations, other design strategies must be considered. Let us introduce the method.

As commented previously, the fractional order $PI^\lambda D^\mu$ controller of equation (1) has two different parts given by the fractional order PI^λ controller of equation (2) and the fractional order PD^μ controller of equation (3). In this method, the fractional order PI^λ controller will be used to cancel the slope of the phase of the plant at the gain crossover frequency ω_c . This way we ensure a flat phase around the frequency of interest. Once the slope is cancelled, the PD^μ controller will be designed to fulfill the design specifications of gain crossover frequency, ω_c , and phase margin, φ_m , following a robustness criterion based on the flatness of the phase curve of this compensator, as will be explained later. This way the resulting phase of the open loop system will be the flattest possible, ensuring the maximum robustness to plant gain variations.

Let us firstly concentrate on the design of the fractional order PI^λ controller.

3.2 Design of the Fractional Order PI^λ Controller

The fractional order PI^λ controller of equation (2) will be used to cancel the slope of the phase of the plant in order to obtain a flat phase around the frequency point ω_c . The relay test will be used to determine this slope. The value of the phase of the plant at the frequencies ω_{\min} , ω_c and ω_{\max} will be estimated by the relay experiment, being $(\omega_{\min}, \omega_{\max})$ the frequency range centered at ω_c in which the slope of the phase is desired to be flat. With these three values of phase, the slope of the phase curve of the plant can be easily calculated.

The phase of the fractional order PI^λ controller is given by:

$$\psi = \arg(PI^\lambda(s)) = \lambda(\arctan(\lambda_1\omega) - \pi/2). \quad (6)$$

In order to cancel the slope of the phase curve of the plant, v , the derivative of the phase of $PI^\lambda(s)$ at the frequency point ω_c must be equal to $-v$, resulting the equation:

$$\psi' = \left[\frac{d\psi}{d\omega} \right]_{\omega=\omega_c} = \lambda \frac{\lambda_1}{1 + (\lambda_1\omega_c)^2} = -v. \quad (7)$$

The parameters λ and λ_1 must be selected so that this expression is fulfilled. Studying the function (7) and doing its derivative with respect to parameter λ_1 (see equation (8)), it is obtained that it

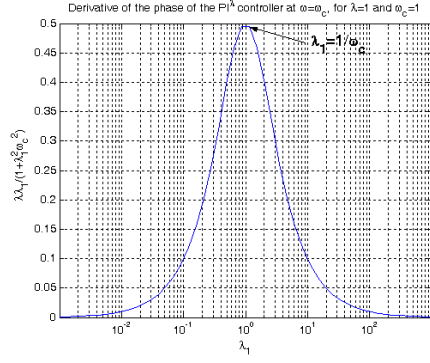


Fig. 2. Derivative of the phase of the PI^λ controller at $\omega = \omega_c$, for $\lambda = 1$ and $\omega_c = 1$

has a maximum at $\lambda_1 = 1/\omega_c$ (see equation (9)), as can be observed in Fig. 2.

$$\frac{d\psi'}{d\lambda_1} = \lambda \left(\frac{(\lambda_1\omega_c)^2 - 1}{(1 + (\lambda_1\omega_c)^2)^2} \right), \quad (8)$$

$$\frac{d\psi'}{d\lambda_1} = 0 \Rightarrow (\lambda_1\omega_c)^2 - 1 = 0 \Rightarrow \lambda_1 = \frac{1}{\omega_c}. \quad (9)$$

That is, choosing $\omega_{cero} = 1/\lambda_1 = \omega_c$ the slope of the plant at the frequency ω_c will be cancelled with the maximum slope of the fractional order controller. Once the value of λ_1 is fixed, the value of λ is easily determined by $\lambda = \frac{-v(1 + (\lambda_1\omega_c)^2)}{\lambda_1}$. It is observed that the value of λ obtained will be minimum when $\lambda_1 = 1/\omega_c$. Variations of the frequency ω_{cero} up or down the frequency ω_c will produce higher values of the parameter λ . Therefore, selecting $\omega_{cero} = \omega_c$ the phase lag of the resulting $PI^\lambda(s)$ controller will be the minimum one (minimum λ). This fact is very interesting from the robustness point of view. The less the phase lag of the controller $PI^\lambda(s)$, the less the phase lead of the controller $PD^\mu(s)$ at the frequency ω_c , favouring the flatness of its phase curve. Then, considering this robustness criterion the value of λ_1 will be fixed to $1/\omega_c$.

3.3 Design of the Fractional Order PD^μ Controller

Defining the system $G_{flat}(s) = G(s)PI^\lambda(s)$, now the controller $PD^\mu(s)$ will be designed so that the open loop system $F(S) = G_{flat}(s)PD^\mu(s)$ fulfils the specifications of gain crossover frequency, ω_c , and phase margin, φ_m , following a robustness criterion based on the flatness of the phase curve of this compensator, as explained in (Monje *et al.*, 2004) and (Monje *et al.*, 2005).

For a specified phase margin, φ_m , and gain crossover frequency, ω_c , the following relationship for the open loop system can be given in the complex plane:

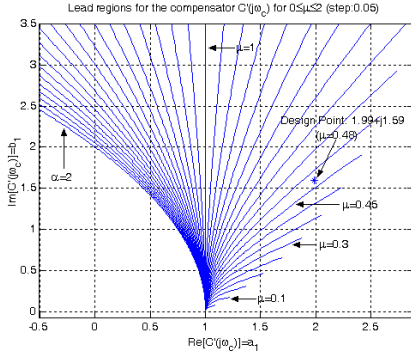


Fig. 3. Lead regions for the fractional order compensator $C'(j\omega_c)$ for $0 \leq \mu \leq 2$

$$\begin{aligned}
 G_{flat}(j\omega_c) \cdot k' \left(\frac{j\lambda_2\omega_c + 1}{jx\lambda_2\omega_c + 1} \right)^\mu &= e^{j(\pi + \varphi_m)} \Rightarrow \\
 \Rightarrow C'(j\omega_c) &= \left(\frac{j\lambda_2\omega_c + 1}{jx\lambda_2\omega_c + 1} \right)^\mu = a_1 + jb_1 \Rightarrow \\
 \Rightarrow \left(\frac{j\lambda_2\omega_c + 1}{jx\lambda_2\omega_c + 1} \right) &= (a_1 + jb_1)^{1/\mu} = a + jb,
 \end{aligned} \tag{10}$$

where $k' = k_c x^\mu = 1$ in this case, $G_{flat}(s)$ is the plant to be controlled and (a_1, b_1) is called the “design point”. After some simple calculations, the expressions for x and λ can be given by:

$$x = \frac{a - 1}{a(a - 1) + b^2}, \quad \lambda_2 = \frac{a(a - 1) + b^2}{b\omega_c}. \tag{11}$$

As explained in (Monje *et al.*, 2004), for the selection of the value of μ the curves in Fig. 3 can be used, in which the lead compensation regions in the complex plane for different positive values of μ are obtained. The zone to the right of each curve is the lead region, and any design point in this zone can be fulfilled with a fractional order compensator having a value of μ equal or bigger than the one defining the curve which passes through the design point (μ_{min}). For instance, for the design point in Fig. 3, the value of μ_{min} is 0.48. By choosing the minimum value μ_{min} , the distance between the zero and the pole of the compensator will be the maximum possible (minimum value of parameter x). In this case, the phase curve of the compensator is the most flat possible and variations in a frequency range centered at ω_c will not produce a significant phase change as in other cases, improving the robustness of the system.

Let us then sum up this design procedure. Once the system $G_{flat}(j\omega_c)$ is defined, and according to equation (10), the parameters of the fractional order compensator in (3) are obtained by simple calculations summarized next, following the robustness feature explained in this section:

- (1) Select a very small initial value of μ , for example $\mu = 0.05$. For this initial value,

calculate the value of x and λ_2 using the relations in (10) and (11).

- (2) If the value of x obtained is negative, then the value of μ is increased a fixed step and step 2 is repeated again. The smaller the fixed increase of μ the more accurate the selection of the parameter μ_{min} we are looking for. Repeat step 2 until the value of x obtained is positive.
- (3) Once a positive value of x is obtained, the value of μ must be recorded as μ_{min} . This value of x will be close to zero and will ensure the maximum flatness of the phase curve of the compensator (robustness constraint). The value of λ_2 corresponding to this value μ_{min} is also recorded.

Therefore, the parameters x , λ_2 and μ have been obtained through this iterative process and the design procedure is concluded.

3.4 Formulation of the Resulting $PI^\lambda D^\mu$ Controller

Once the parameters of the fractional order $PI^\lambda D^\mu$ controller of equation (1) are obtained by following the design methods explained above, we can identify these parameters with those ones of the standard $PI^\lambda D^\mu$ controller given by:

$$C_{std}(s) = k_p \left(1 + \frac{1}{T_i s} \right)^\lambda \left(1 + \frac{T_d s}{1 + \frac{s T_d}{N}} \right)^\mu. \tag{12}$$

Doing some calculations in (12):

$$C_{std}(s) = \frac{k_p}{(T_i)^\lambda} \left(\frac{T_i s + 1}{s} \right)^\lambda \left(\frac{T_d(1 + \frac{1}{N})s + 1}{\frac{T_d}{N}s + 1} \right)^\mu. \tag{13}$$

Comparing expressions (1) and (12), the relations obtained are $T_i = \lambda_1$, $k_p = \frac{k'}{(\lambda_1)^\lambda}$, $N = \frac{1-x}{x}$ and $T_d = \lambda_2(1-x)$.

In next section the auto-tuning method proposed here is illustrated with some examples of application.

4. AN EXAMPLE OF APPLICATION

In this section an example of application of the auto-tuning method proposed is given. The plant to control is a position servo with a time delay given by $G(s) = \frac{0.55}{s(0.6s+1)} e^{-0.05s}$. The specifications for the controlled system are a gain crossover frequency of $\omega_c = 2.3 \text{ rad/sec}$, a phase margin of $\varphi_m = 72^\circ$ and robustness to variations of the plant gain. In this case the gain of the controller $k' = k_c x^\mu$ will be fixed to 1. With the relay experiment the magnitude and the phase

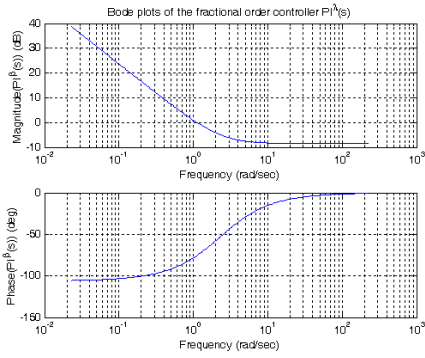


Fig. 4. Bode plots of the fractional order controller $PI^\lambda(s)$

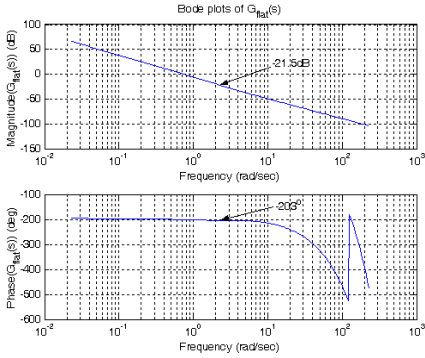


Fig. 5. Bode plots of the system $G_{flat}(s)$

of the plant at the frequency $\omega_c = 2.3rad/sec$ are obtained. To estimate the slope of the phase curve, the relay test is also experimented for the frequencies ω_{min} and ω_{max} , resulting a slope of $-0.2566deg/rad/sec$ around the frequency point ω_c . With the value of the slope and applying the criterion above for the fractional order PI^λ controller (see equations (7) and (9)), the controller to cancel the slope of the phase curve of the plant is:

$$PI^\lambda(s) = \left(\frac{0.4348s + 1}{s} \right)^{1.1803},$$

whose Bode plots are shown in Fig. 4.

The Bode plots of the system $G_{flat}(s) = G(s) \cdot PI^\lambda(s)$ are given in Fig. 5.

At the frequency $\omega_c = 2.3rad/sec$ the system $G_{flat}(s)$ has a magnitude of $-21.5dB$ and a phase of -203° . These values can be easily obtained through the value of the magnitude and phase of the plant estimated by the relay test at the frequency ω_c and the magnitude and phase of the controller $PI^\lambda(s)$ at the same frequency. It is observed that the phase of $G_{flat}(s)$ is totally flat around the frequency point ω_c due to the effect of the PI^λ controller.

The controller $PD^\mu(s)$ is now designed to fulfil the specifications of phase margin and gain crossover frequency required for the controlled system. Fol-

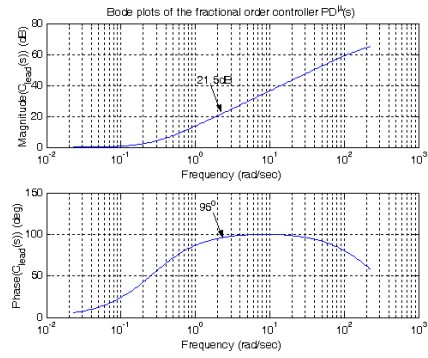


Fig. 6. Bode plots of the fractional order compensator $C_{lead}(s)$

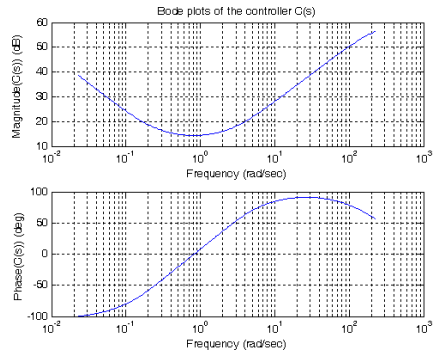


Fig. 7. Bode plots of the fractional controller $C(s)$

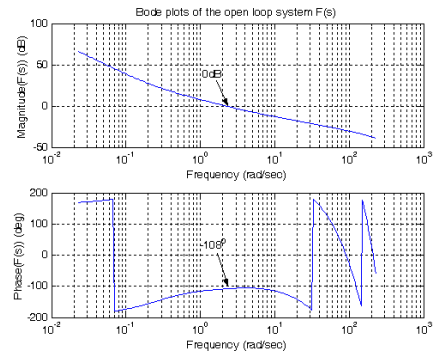


Fig. 8. Bode plots of the open loop system $F(s)$

lowing the iterative process described above, the resulting controller is given by:

$$PD^\mu(s) = \left(\frac{3.7282s + 1}{0.0037s + 1} \right)^{1.1580}.$$

The Bode plots of this controller are shown in Fig. 6. At the frequency $\omega_c = 2.3rad/sec$ the controller $PD^\mu(s)$ has a magnitude of $21.5dB$ and a phase of 95° . The Bode plots of the controller $C(s)$ and the open loop system $F(s)$ are shown in Fig. 7 and 8, respectively.

In Fig. 8 it is observed that the design specifications for the control system are fulfilled and that the phase of the open loop system $F(s)$ is the flattest possible, ensuring the maximum robustness possible to variations in the gain of the plant.

5. CONCLUDING REMARKS

In this work an auto-tuning method for the fractional order $PI^\lambda D^\mu$ controller has been proposed. This controller has two different parts: a fractional order PI^λ controller and a fractional order PD^μ controller. In this method, the PI^λ controller is used to cancel the slope of the phase curve of the plant around the frequency point ω_c . This slope will be estimated by using the relay test. Once the slope is cancelled, the PD^μ controller is designed to fulfill the specifications of gain crossover frequency, ω_c , and phase margin, φ_m , following the robustness criterion based on the flatness of the phase curve of this controller. The simplicity of this method allows to obtain in a very easy and direct way the set of parameters of the fractional order $PI^\lambda D^\mu$ controller that fulfils the design specifications and ensures the maximum robustness to plant gain variations.

REFERENCES

- Åström, K. J. and B. Wittenmark (1973). On Self Tuning Regulators. *Automatica* **9**, 185–199.
- Åström, K. J. and T. Hägglund (1984). Automatic Tuning of Simple Regulators with Specifications of Phase and Amplitude Margins. *Automatica* **20**(5), 645–651.
- Chen, Y. Q. and K. L. Moore (2005). Relay Feedback Tuning of Robust PID Controllers With Iso-Damping Property. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* **35**(1), 23–31.
- Chen, Y. Q., K. L. Moore, B. M. Vinagre and I. Podlubny (2004). Robust PID Controller Autotuning with a Phase Shaper. In: *First IFAC Workshop on Fractional Differentiation and Its Applications*. ENSEIRB, Bordeaux, France. pp. 162–167.
- Hang, C. C., K. J. Astrom and Q. G. Wang (2002). Relay Feedback Auto-Tuning of Process Controllers-A Tutorial Review. *Journal of Process Control* **12**, 143–162.
- Monje, C. A., A. J. Calderón, B. M. Vinagre and V. Feliu (2004). The Fractional Order Lead Compensator. In: *Proceedings of the IEEE International Conference on Computational Cybernetics*. Vienna, Austria. pp. 347–352.
- Monje, C. A., B. M. Vinagre, A. J. Calderón, V. Feliu and Y. Q. Chen (2005). Auto-Tuning of Fractional Lead-Lag Compensators. In: *16th IFAC World Congress*.
- Tan, K. K., S. Huang and R. Ferdous (2002). Robust Self-Tuning PID Controller for Non-linear Systems. *Journal of Process Control* **12**(7).

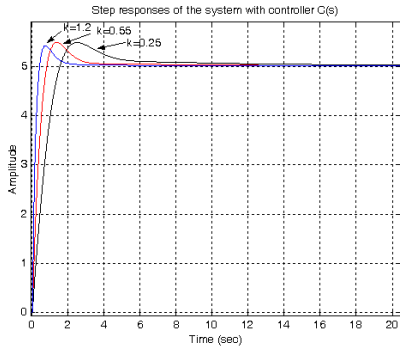


Fig. 9. Step responses of the system with controller $C(s)$

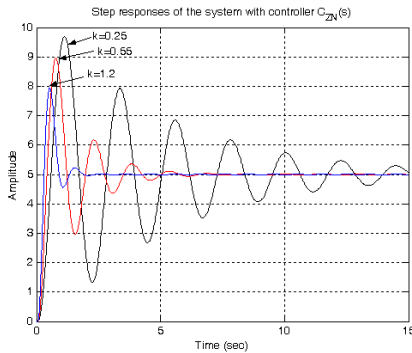


Fig. 10. Step responses of the system with controller $C_{ZN}(s)$

It has to be taken into account that a fractional-order controller is an infinite-dimensional linear filter, and that all existing implementation schemes are based on finite-dimensional approximations. In this particular case, an analogical method is used for the implementation of the fractional order controller, that is, a finite-dimensional analogical approximation is obtained. For that purpose a frequency domain identification technique is used, the one provided by Matlab function "invfreqs", which works fine for our transfer function fitting here. An integer-order transfer function $C(s)$ is obtained which fits the frequency response of the fractional order controller in the range $\omega \in (10^{-2}, 10^2)$, and whose stability is guaranteed. The step responses of the system controlled with $C(s)$ are shown in Fig. 9, where it can be observed that the system exhibits robust performance to gain variations, keeping constant the overshoot of the time response.

Comparing these responses with the ones obtained with controller $C_{ZN}(s)$ designed by the second method of Ziegler-Nichols (Fig. 10), it can be observed the better performance of the system with the fractional order controller $C(s)$.

$$C_{ZN}(s) = 22.1010 \left(1 + \frac{1}{0.55s} + 0.1375s \right).$$