ISA Transactions 52 (2013) 807–815

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Research Article

Fractional adaptive control for an automatic voltage regulator

Norelys Aguila-Camacho [a,b,*], Manuel A. Duarte-Mermoud [a,b]

[a] Department of Electrical Engineering, University of Chile, Av. Tupper 2007, Santiago de Chile, Chile
[b] Advanced Mining Technology Center, Av. Tupper 2007, Santiago de Chile, Chile

ARTICLE INFO

Article history:
Received 1 December 2012
Received in revised form
18 April 2013
Accepted 6 June 2013
Available online 7 July 2013
This paper was recommended for publication by Prof. Y. Chen

Keywords:
Fractional order adaptive control
Model reference adaptive control
Automatic voltage regulator
Genetic algorithms optimization

ABSTRACT

This paper presents the application of a direct Fractional Order Model Reference Adaptive Controller (FOMRAC) to an Automatic Voltage Regulator (AVR). A direct FOMRAC is a direct Model Reference Adaptive Control (MRAC), whose controller parameters are adjusted using fractional order differential equations. Four realizations of the FOMRAC were designed in this work, each one considering different orders for the plant model. The design procedure consisted of determining the optimal values of the fractional order and the adaptive gains for each adaptive law, using Genetic algorithm optimization. Comparisons were made among the four FOMRAC designs, a fractional order PID (FOPID), a classical PID, and four Integer Order Model Reference Adaptive Controllers (IOMRAC), showing that the FOMRAC can improve the controlled system behavior and its robustness with respect to model uncertainties. Finally, some performance indices are presented here for the controlled schemes, in order to show the advantages and disadvantages of the FOMRAC.

© 2013 ISA. Published by Elsevier Ltd. All rights reserved.

1. Introduction

Adaptive control refers to the control of partially known systems. This uncertainty may be caused by unknown (fixed or time-varying) system parameters, and/or the plant being only partially modeled or subjected to external disturbances. In these cases, conventional control theory does not achieve satisfactory performance, whereas adaptive control has been a very useful tool, given its ability to adjust parameters automatically by means of adaptive laws, which allow dealing with uncertainty while achieving the desired system behavior.

One of the most popular adaptive control schemes is Model Reference Adaptive Control (MRAC), where the aim is to find a suitable control signal such that the controlled system output follows the reference model output, while at the same time the stability of the closed loop system is preserved [14].

The subject of fractional calculus (calculus of integrals and derivatives of arbitrary real or complex order) has gained considerable interest and importance during recent years, mainly due to its demonstrated applicability in numerous seemingly diverse and widespread fields of science and engineering [10].

There has been growing interest in combining classical MRAC schemes and fractional calculus in recent years. Some MRAC schemes have been proposed, in which the model of the plant to be controlled, the reference model and/or the adaptive laws for adjusting the parameters are defined by fractional order differential equations [20,12,13,22,18].

The lifestyle of modern society is deeply linked to the use of electricity. Most of the equipment used today operates on the basis of electrical energy, and is sensitive to both the continuity of the power supply, and its quality (voltage and frequency levels).

The power demand is never constant in power generation systems, and this affects the output voltage and frequency levels of the generators. For this reason, any power generation system should have a control scheme, in order to maintain the voltage and frequency levels within desired values, regardless of the demand.

The Automatic Voltage Regulator (AVR) is the controller whose main purpose is to maintain the voltage level in an electric generator at acceptable values by adjusting the generator exciter voltage.

Many control schemes have been proposed for AVR. PID controllers are the most reported control scheme for the AVR, and the difference between these works lies in the technique used to select the PID parameters. It can be cited for example PID controllers whose parameters have been adjusted using Particle Swarm Optimization (PSO) [7,16], using third order PSO [8], using Quantum-behaved PSO [2], using optimization method based in Continuous Action Reinforcement Learning Automata (CARLA) [9], using Adaptive Tabu Search algorithm [15], and using combined genetic algorithm and fuzzy logic approach [3].

* Corresponding author at: Department of Electrical Engineering, University of Chile, Av. Tupper 2007, Santiago de Chile, Chile. Tel.: +56 2 29784920; fax: +56 2 26720162.
E-mail addresses: naguila@ing.uchile.cl, norelys1980@gmail.com (N. Aguila-Camacho), mduartem@ing.uchile.cl (M.A. Duarte-Mermoud).

0019-0578/$ - see front matter © 2013 ISA. Published by Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.isatra.2013.06.005

Other control schemes have been proposed, different from PIDs, such as Fuzzy Gain Scheduled PI Controllers (FGSPIC) [17], Brain Emotional Learning Intelligent Controllers (BELBIC) [19], Nonlinear adaptive controllers [6] and Fractional order PID controllers [23]. This last one is a fractional PID, whose parameters are adjusted using PSO. However, given the importance of the control problem, this topic is still open to control solutions that would improve the performance of the controlled system, for example minimizing the overshoot and the convergence time of the control error to zero.

This paper presents a direct Fractional Order Model Reference Adaptive Controller (direct FOMRAC) for an AVR, where the parameters of the controller are adjusted using adaptive laws defined by fractional order differential equations. This FOMRAC shows an improvement in characteristics of the response of the controlled system and in robustness with respect to model uncertainties.

The paper is organized as follows: Section 2 introduces general concepts of direct FOMRAC, fractional calculus and Genetic algorithm optimization. In Section 3 the model of the plant to be controlled is presented, and the proposed fractional adaptive control scheme is introduced. Section 4 contains the results obtained through simulations of the proposed control scheme, and its comparison with other control schemes proposed in the control literature. Section 5 contains the evaluation of the system behavior for the different control schemes studied, making use of various performance indices. Finally, Section 6 presents the conclusions of the work.

2. General concepts

This section introduces some general concepts, which are used throughout the work, in order to ease the understanding of the proposed schemes.

2.1. Fractional calculus

In fractional calculus, the traditional definitions of the integral and derivative of a function are generalized from integer orders to real orders.

In the time domain, the fractional order derivative and fractional order integral operators are defined by a convolution operation.

According to Kilbas et al. [10], the Riemann–Liouville fractional integral of order $\alpha \in \mathbb{R}$, with $\alpha \geq 0$ and denoted as ${}^{R}I_0^\alpha$, is defined as

$$ {}^{R}I_0^\alpha f(t) = \frac{1}{\Gamma(\alpha)} \int_0^t \frac{f(\tau)}{(t-\tau)^{1-\alpha}}\, d\tau, \quad t>0 \tag{1}$$

where $\Gamma(\alpha)$ is the Gamma function, defined as

$$\Gamma(\alpha) = \int_0^\infty t^{\alpha-1} e^{-t}\, dt$$

Several definitions exist regarding the fractional derivative of order $\alpha \geq 0$, but the Caputo definition defined in (2) is used the most in engineering applications, since this definition incorporates initial conditions for $f(\cdot)$ and its integer order derivatives, i.e., initial conditions that are physically appealing in the traditional way:

$$ {}^{C}D_0^\alpha f(t) = \frac{1}{\Gamma(n-\alpha)} \int_0^t \frac{f^n(\tau)}{(t-\tau)^{\alpha-n+1}}\, d\tau, \quad n-1<\alpha<n, \ n \in \mathbb{Z}^+ \tag{2}$$

One of the most common ways of using fractional integrals and derivatives in simulations and practical implementations is by means of numerical approximations of these operators. The idea is to obtain integer-order transfer functions whose behavior approximates the fractional order Laplace operator:

$$C(s) = ks^\alpha \tag{3}$$

Oustaloup's method is one of the available frequency-domain methods for making this approximation, which uses a recursive distribution of N poles and N zeros [21] of the form

$$C(s) = k' \prod_{n=1}^{N} \frac{1+s/\omega_{zn}}{1+s/\omega_{pn}} \tag{4}$$

The gain k' is adjusted so that if $k=1$ then $|C(s)|=0$ dB at 1 rad/s. Zeros and poles are placed inside a frequency interval $[\omega_l, \omega_h]$.

This approximation is available in the fractional derivative block of the Ninteger Toolbox for Matlab [4], and is the one used in this work.

2.2. Fractional model reference adaptive control

According to Narendra and Annaswamy [14], the Model Reference Adaptive Control (MRAC) problem can be stated qualitatively as follows: let a linear time-invariant (LTI) plant P be defined by input–output pairs $\{u(\cdot), y_p(\cdot)\}$. Let a stable LTI reference model M be defined by its input–output pair $\{r(\cdot), y_m(\cdot)\}$ where $r: \mathbb{R}^+ \to \mathbb{R}$ is a bounded piecewise-continuous function. The aim of the MRAC is to determine the control input $u(t)$ for all $t \geq t_0$ so that

$$\lim_{t\to\infty} |y_p(t) - y_m(t)| = 0$$

In the case of direct MRAC, the parameters of the controller are directly adjusted; that is to say, no identification of the plant parameters is attempted.

For the classical direct MRAC, the controller parameters are adjusted by using a differential equation of integer order (adaptive law). In the case of direct FOMRAC, the controller parameters are adjusted using a differential equation of fractional order (fractional adaptive law), with the same structure of the adaptive laws used in the integer order MRAC [14], but the derivative order is fractional. Details of the fractional adaptive law are given in Table 1.

In this work, a direct FOMRAC has been implemented for the AVR. In general terms, the control scheme is defined as follows.

Given a known reference model, defined by the transfer function $G_m(s)$, a reference signal $r(t)$ is applied to obtain the measurable output $y_m(t)$. This output is compared with the AVR output voltage $y_p(t)$ to compute the control error defined as $e(t) = y_p(t) - y_m(t)$.

Using this control error and other available signals in the control scheme, the controller parameters are adjusted, using a fractional

Table 1
Fractional MRAC implementations details.

Reference model	$G_m(s) = \dfrac{1.2}{s^3+5.2s^2+7s+1.2}$
Control law	$u(t) = \theta(t)^T \omega(t)$
	$\theta^T(t) = [k(t) \quad \theta_1 T(t) \quad \theta_0(t) \quad \theta_2^T(t)] \in \mathbb{R}^{10}$
	$\omega(t) = [r(t) \quad \omega_1^T(t) \quad y_p(t) \quad \omega_2^T(t)]^T \in \mathbb{R}^{10}$
Auxiliary signals	$\dot{\omega}_1(t) = \Lambda\omega_1(t) + lu(t)$
	$\dot{\omega}_2(t) = \Lambda\omega_2(t) + ly_p(t)$
	$\Lambda = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -2 & 0 & 0 \\ 0 & 0 & -3 & 0 \\ 0 & 0 & 0 & -4 \end{bmatrix}$
	$l = [-1\ 1\ 3\ 4^T$
Errors	$e_1(t) = y_p(t) - y_m(t)$
	$e_2(t) = \theta^T(t)\overline{\omega}(t) - \overline{u}(t)$
	$\varepsilon(t) = e_1(t) + k_1(t)e_2(t)$
	$\overline{u}(t) = G_m(s)u(t)$
Adaptive law	$D^\alpha k_1(t) = -\gamma \dfrac{\varepsilon(t)e_2(t)}{1+\overline{\omega}(t)\overline{\omega}^T(t)}$
	$D^\alpha \theta(t) = -\gamma \dfrac{\varepsilon(t)\overline{\omega}(t)}{1+\overline{\omega}(t)\overline{\omega}^T(t)}$
	$\overline{\omega}(t) = G_m(s)\omega(t)$

adaptive law. These parameter values are then used to compute the control signal $u(t)$ applied to the AVR, in order to achieve AVR output voltage $y_p(t)$ equal to the model reference output $y_m(t)$. Specific details of the control scheme are given in Section 3.

2.3. Genetic algorithm optimization

Genetic algorithms (GA) belong to the larger class of Evolutionary Algorithms (EA), which generate solutions to optimization problems using techniques inspired by natural evolution, such as inheritance, mutation, selection, and crossover. Applied to control schemes, GA's have proved to be useful, for example, selecting the optimal controller parameters to minimize a fitness function for a controlled system.

In the real world, an organism's characteristics are encoded in its DNA. Genetic algorithms store the characteristics of artificial organisms in an electronic genotype, which mimics the DNA of natural life [1]. So GA's work with a population of potential solutions to a specific problem, in which each individual within the population represents a particular solution to the problem. The population evolves, over generations, to produce better solutions to the problem.

A fitness value is assigned to each individual within the population, in order to measure the quality of the solution it represents. Later, evolution is performed using a set of stochastic genetic operators, which manipulates the genetic code, performing, for example, crossover and mutation. This evolution generally results in better individuals, that is, solutions of the problem with better fitness values.

Usually, the evolution process stops when a limited number of generations have been reached or when the fitness function is under a prefixed value.

In this work, Matlab Genetic Algorithm Toolbox was used to find the optimal fractional orders and adaptive gains for the FOMRAC adaptive laws. Details of the GA implementation are given in Section 3.

3. AVR design using FOMRAC and Genetic algorithms

This section introduces the AVR model used in this work, and presents the design procedure of the FOMRAC.

3.1. FOMRAC for AVR

The role of an AVR is to maintain the terminal voltage magnitude of a synchronous generator at a specified value. As shown in Fig. 1(a), a simple AVR system is comprised of four main components, namely the amplifier, exciter, generator, and sensor. Reasonable transfer functions for these components are the result of a linearization procedure [7]. Fig. 1(a) shows the block diagram of the AVR with the corresponding transfer functions for each block, and Fig. 1(b) shows the generator model [11], used in this study. The system parameter values used for simulations in this work are $k_A=10$, $\tau_A=0.1$ s, $k_E=1$, $\tau_E=0.5$ s, $k_1=1.591$, $k_2=1.5$, $k_3=0.333$, $k_4=1.8$, $k_5=-0.12$, $k_6=0.3$, $\tau_3=1.91$ s, $H=3$, $K_D=0$, $\omega_0=377$ rad/s, $k_R=1$ and $\tau_R=0.06$ s. These values were taken from [23].

Thus, for fractional adaptive controller design purposes, the plant to be controlled (from control signal input to sensor output) has a sixth order transfer function. However, since the dynamics of the sensor is very fast, its influence in the transfer function can be neglected, and the plant transfer function $G_p(s)$ is then considered as one of the fifth order of the form

$$G_p(s)=\frac{b_2s^2+b_1s+b_0}{a_5s^5+a_4s^4+a_3s^3+a_2s^2+a_1s+a_0} \tag{5}$$

where $b_2=5.994$, $b_1=0$, $b_0=825.2$, $a_5=0.573$, $a_4=7.176$, $a_3=72.36$, $a_2=706.6$, $a_1=1302$ and $a_0=260.8$. Even though the sensor transfer function is neglected from the point of view of controller design, the sixth order transfer function is used in the simulations.

According to Narendra and Annaswamy [14], the reference model has to be chosen with a relative degree greater or equal to the plant relative degree, which in this case is $n^*=3$. Beyond this, selection of the reference model is the responsibility of the control designer, and will be chosen according to the requirements to be met by the control scheme. In this case, given the characteristics of the power generation process, a smooth step response is desired, a small overshoot and settling time, and zero (or minimal) steady-state error. In order to satisfy these requirements, the reference model was selected by the transfer function $G_m(s)$ shown in Table 1.

Since the relative degree of the transfer function of the plant is $n^*=3$, according to Narendra and Annaswamy [14], the direct MRAC implementation is like the one shown in Fig. 2.

The controller parameters are given by the vector $\theta^T=[k\ \theta_1^T\ \theta_0\ \theta_2^T]\in\mathbb{R}^{10}$ and the scalar $k_1\in\mathbb{R}$. However, they are not adjusted using differential equations of integer order as in [14], but by using fractional order differential equations, with order $0<\alpha<1$. Table 1 summarizes the design of the direct FOMRAC and the corresponding values used in the implementation. The control signal generated by the FOMRAC corresponds to the field voltage.

According to Narendra and Annaswamy [14], the number of parameters to be adjusted is $2n+1$, where n is the order of the plant transfer function. So in this case, where $n=5$, the total number of parameters to be adjusted is 11.

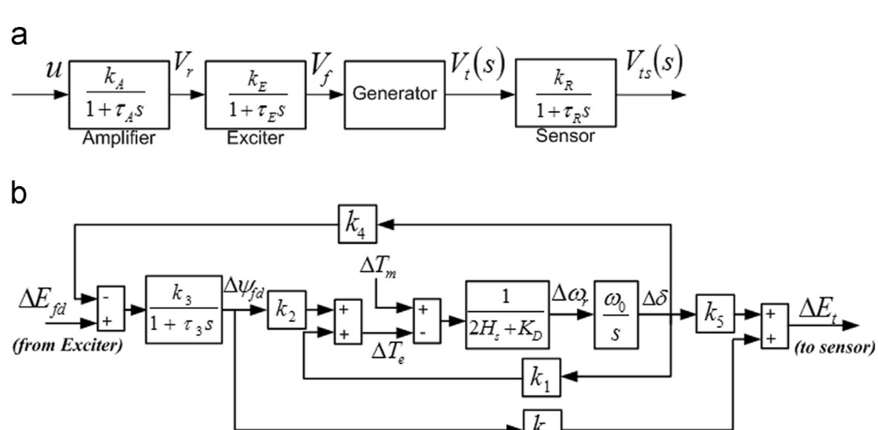

Fig. 1. Block diagrams for AVR and generator. [23]. (a) Block diagram of the AVR. (b) Block diagram of the generator.

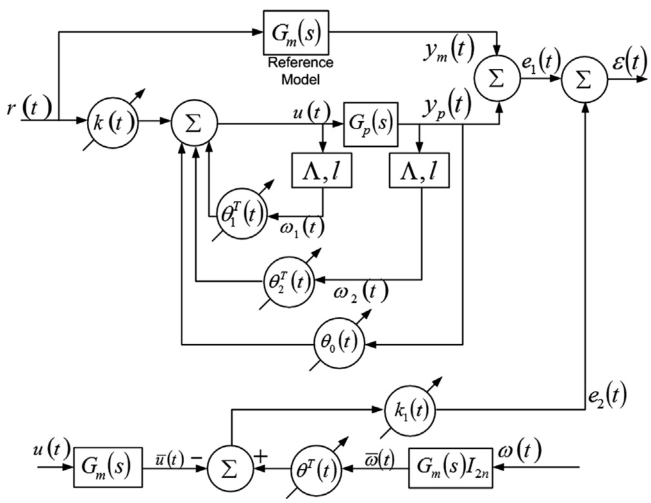

Fig. 2. Block diagram for the implementation of the FOMRAC for the AVR [14].

Since the derivative order α of the adaptive laws is fractional, different derivative orders could be used to adjust each one of the parameters. The same comment can be made for the adaptive gain γ, in the sense that different values for each parameter could also be used. Thus, in this control scheme, different fractional orders and adaptive gains are considered for each one of the 11 parameters to be adjusted.

3.2. Optimization of FOMRAC design parameters

Genetic algorithm (GA) optimization is used to select the optimal derivative order and adaptive gain for each controller parameter adaptive law, so that the controlled system exhibits the desired behavior, measured through the proposed performance criterion.

The parameter vector optimized by the GA corresponds to $x=[\alpha^T \ \ \gamma^T] \in \Re^{22}$, where α and γ are vectors of dimension 11.

Simulation studies suggest that when using fractional order in the interval $0<\alpha<2$, the FOMRAC remains stable, however when using $1<\alpha<2$, transient behavior of the controlled system is quite oscillatory. That is why a lower bound of 0 and an upper bound of 1 for the optimization of α were considered, and the search space of the GA was restricted to this interval. In the case of the adaptive gains, lower and upper bounds of 0 and 100 were used. Zero initial conditions were chosen for every controller parameter.

In order to achieve system performance in accord with control specifications and based on [23], the performance criterion used in the optimization process was defined as

$$J(x)=w_1 M_p+w_2 t_s+w_3 E_{ss}+w_4 \int_0^{t_f} |e_c(t)|\, dt+w_5 \int_0^{t_f} u^2(t)\, dt \qquad (6)$$

where M_p is the overshoot, t_s is the settling time, E_{ss} is the steady-state control error, $e_c(t)=r(t)-y_p(t)$ is the control error (difference between the reference voltage and the output voltage), and $u(t)$ corresponds to the control signal generated by the FOMRAC. The importance of each of these elements in the performance criterion function is given by weighting factors $w_i,\ i=1,\dots,5$ and it is up to the designer to select these values.

The choice of the weighting factors is not an easy task, and there are many ways to do it. In the case of this paper, the values used in Zamani et al. [23] were used as a starting point. Several trials were performed for the optimization process, using values for the weighting factors around the values used in Zamani et al. [23]. The smallest value of J was obtained for $w_1=w_2=w_4=1$, $w_3=1000$ and $w_5=7$. The integration limit t_f in (6) was set to 100 s.

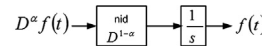

Fig. 3. Block diagram used for the implementation of the fractional adaptive law.

The optimization process was carried out using the Matlab GA toolbox. The most representative GA parameters used in the optimization procedure are

- Population type: double vector.
- Population size = 25.
- Number of generations = 130.

The remaining parameters were chosen at their default values.

The fractional adaptive laws were implemented using the Ninteger Toolbox for Matlab [4], specifically the NID block. In order to include the initial conditions, which are not included in the NID block, the definition of the Caputo fractional derivative (2) and a property of the fractional integrals were used. When $0<\alpha<1$, as it is in our case, the block diagram used to generate the estimated parameters in the FOMRAC is shown in Fig. 3.

The NID block used in the scheme is based on the Oustaloup's approximation method mentioned in Section 2.1, with 5 poles, 5 zeros and a frequency interval $[0.001,\ 1000]$ rad/seg.

In general terms, the optimization process is carried out in the following way:

1. In the first generation the population is randomly initialized. Every individual contains the values of the derivative orders and the adaptive gains for the adaptive law.
2. For every individual of the population, the value of the performance criterion (6) is calculated. This means that at every iteration, the simulation of the controlled system (plant + FOMRAC) is performed.
3. Evolution is performed and new individuals result.
4. If the number of iterations reaches the maximum, then go to step 5, otherwise go to step 2.
5. The optimal controller parameters have been found with the lowest performance criterion.

4. Simulation results

This section presents the simulation results obtained for the AVR controlled by the FOMRAC designed in Section 3. These results are compared with those obtained using fractional PID and classical PID, which are reported in the technical literature. Besides, some other less complex FOMRAC's are designed and presented through simulations.

4.1. Behavior of the FOMRAC

Several trials were performed for the optimization process in order to find the best set of controller parameters. The best case gave $J=130.93$ and the following optimal values for the fractional differential orders and the adaptive gains:

$\alpha_1=0.1508,\quad \alpha_2=0.4152,\quad \alpha_3=0.6,\quad \alpha_4=0.1844$
$\alpha_5=0.7627,\quad \alpha_6=0.2944,\quad \alpha_7=0.8110,\quad \alpha_8=0.9998$
$\alpha_9=0.7024,\quad \alpha_{10}=0.1446,\quad \alpha_{11}=0.9885$

$\gamma_1=3.4345,\quad \gamma_2=1.8095,\quad \gamma_3=0.4423,\quad \gamma_4=1.9116$
$\gamma_5=1.8242,\quad \gamma_6=0.7816,\quad \gamma_7=0.0582,\quad \gamma_8=2.3671$
$\gamma_9=0.1756,\quad \gamma_{10}=0.7980,\quad \gamma_{11}=2.9059$

Fig. 4 shows the step responses of the controlled system using FOMRAC, as well as the system being controlled by a fractional order proportional–integral-derivative controller (FOPID), a classical PID and an Integer Order Model Reference Adaptive Controller (IOMRAC).

The results corresponding to FOPID and PID were reported in [23], and they were also tuned optimally. In the case of the IOMRAC, it was designed and analyzed in this work for comparison purposes, and it was followed the same design procedure as in the FOMRAC. In this case, due to the fact that derivative orders are integers, only the optimization of the adaptive gains ($\gamma_{ie},\ i=1,2,\ldots,11$) was required. The optimized values for the adaptive gains in the IOMRAC case were found to be

$$\gamma_{1e}=1.8138,\quad \gamma_{2e}=10.179,\quad \gamma_{3e}=11.4493,\quad \gamma_{4e}=9.9687$$
$$\gamma_{5e}=4.5219,\quad \gamma_{6e}=3.8443,\quad \gamma_{7e}=1.2544,\quad \gamma_{8e}=5.1298$$
$$\gamma_{9e}=0.3906,\quad \gamma_{10e}=1.8225,\quad \gamma_{11e}=20.3988$$

As can be seen from Fig. 4, the settling time is shorter for the IOMRAC and the FOMRAC, but a more demanding control signal is required as compared with the cases of FOPID and PID. The control signal of the IOMRAC, however, is more demanding than the FOMRAC control signal, that is to say, the FOPID delivers a good balance between the transient response and the control signal behavior.

The FOMRAC better transient response relies on the possibility of selecting different fractional orders for the adaptive laws, through the optimization process. The fractional order adaptive laws, depending on the value of alpha, allow obtaining smoother transient responses than the IOMRAC, and using the optimization procedure the best combination of fractional orders as well as adaptive gains was found.

As far as the controller dimension is concerned, and consequently the complexity of the FOMRAC, some other simpler approaches can be implemented. The transfer function of the plant corresponds to a fifth order model plus a first order sensor dynamic, which is usually neglected. However, this transfer function can be approximated quite well by a fourth order transfer function ($G_{p4}(s)$), by a third order transfer function ($G_{p3}(s)$) and even by a second order transfer function ($G_{p2}(s)$). Details of these reduced order models are shown in (7). Fig. 5 shows the step response of the plant (fifth order transfer function plus first order sensor dynamic) together with the step response of three reduced order models (fourth, third and second), which supports the previous statement. The step response of the fifth order transfer function is plotted as well, in order to show that the influence of the sensor dynamic can be neglected.

The reduced order transfer functions are as follows:

$$G_{p4}(s)=\frac{82.52s+825.2}{5.73s^4+714.46s^3+579s^2+1276s+260.8}$$
$$G_{p3}(s)=\frac{13.8}{s^3+12.21s^2+22.6s+4.34}$$
$$G_{p2}(s)=\frac{1.375}{s^2+2.21s+0.4348} \tag{7}$$

For this reason, simpler FOMRAC's can be designed, considering the reduced order transfer functions rather than the original one, adjusting fewer parameters than in the case already presented in Fig. 4. Following the same design procedure presented in Section 3, three other FOMRAC's were proposed. These correspond to a fourth order controller (FOMRAC$_4$), a third order controller (FOMRAC$_3$) and a second order controller (FOMRAC$_2$). In the simulation of these cases, the plant to be controlled is the real one, that is to say, a fifth order model with a first order sensor dynamic.

Implementation details for each scheme are given in Table 2. The rest of the implementation details is the same as those used in the simulations of Section 3 and shown in Table 1. The resulting value of functional J and the optimal values for each controller adaptive law are given in Table 3. The controller designed in Section 3 is referenced here as FOMRAC$_5$.

Similarly, three other IOMRAC's were proposed for comparison purposes, following the same design procedure than for the fractional case and the specifications given in Table 3, but with integer order adaptive laws. Those controllers are referenced as IOMRAC$_4$, IOMRAC$_3$ and IOMRAC$_2$. The resulting values of functional J and the optimal values for each integer order controller adaptive law are given in Table 4.

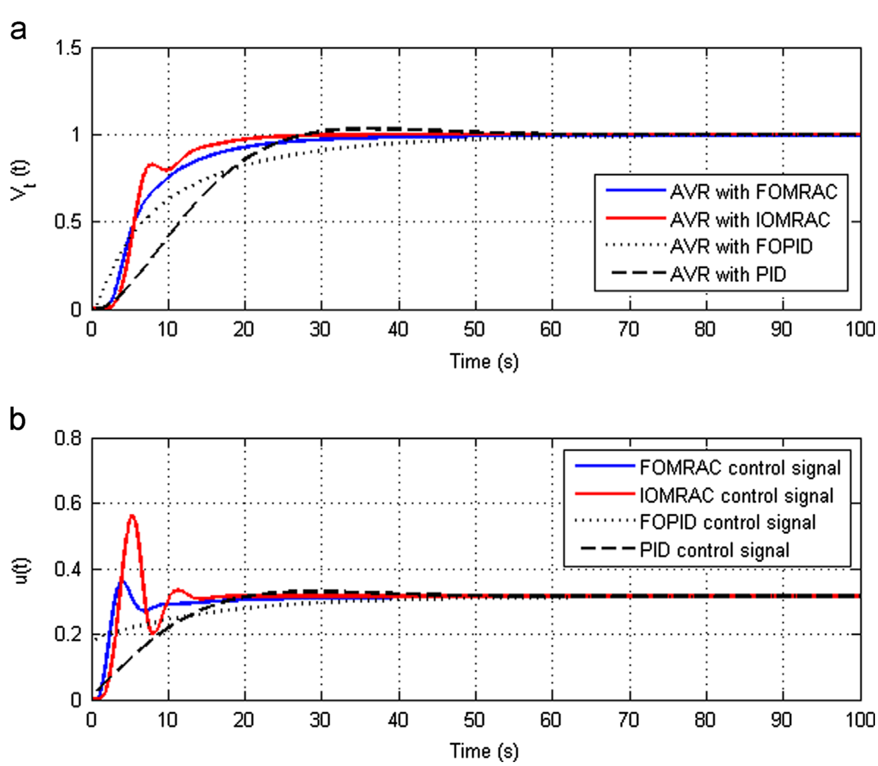

Fig. 4. Step responses (a) and control signals (b) of the AVR controlled by FOMRAC, IOMRAC, FOPID and PID.

Fig. 6 shows the step response of the controlled system for each of the four FOMRAC's designed, and compared again with the controlled system using FOPID and integer order PID, reported in [23]. Fig. 7 shows the control signal for each case.

The behavior of the controlled system is similar for each case when using FOMRAC, and in all cases the settling time is shorter than in the cases using FOPID and PID. This result confirms that it is possible to use a reduced order controller to control the plant. The control signal, however, is more demanding for the fractional order adaptive controllers than for the PID and PIDF.

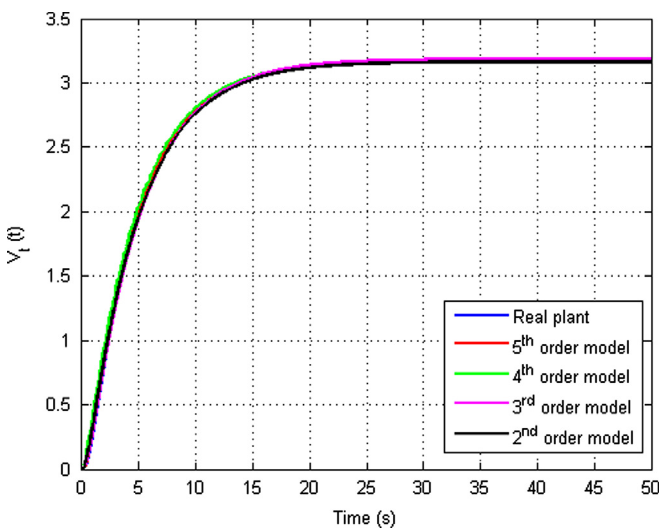

Fig. 5. Step responses of the plant model compared with the reduced order models

Fig. 8 shows the step response of the controlled system for each of the four IOMRAC's designed, compared with the controlled system using FOPID and integer order PID. In this case the behavior of the controlled system is similar for each case when using $IOMRAC_5$, $IOMRAC_4$ and $IOMRAC_3$, with settling times shorter than in the cases of using FOPID and integer order PID. However, in the case of using $IOMRAC_2$, the controlled system has an oscillatory transient response, exhibiting an important difference with the $FOMRAC_2$ case.

As can be seen from Fig. 9, the control signal for the integer order controllers is, in all the cases, more demanding than for the PID and the PIDF. Comparing Figs. 7 and 9, it can be seen that the control signal for the integer order controllers is, in all the cases, more demanding than for the fractional order adaptive controllers.

4.2. Robustness of the FOMRAC

Due to the nature of the electric generation process, variations in parameter values usually occur. For example, changes in load conditions are presented. In order to check the robustness of the FOMRAC with respect to parameter changes, some simulations were performed, using the same parameter changes reported in [23].

First it is assumed that at $t=100$ s, parameter $K_1=1.59$, changes to $K_1=1$ due to changes in load conditions. Fig. 10 shows the terminal voltage response of the AVR using the $FOMRAC_3$ designed in this study, (which considers that the plant can be modeled as a third order model) and compared with FOPID, PID and $IOMRAC_3$. The $FOMRAC_4$ and the $FOMRAC_5$ present quite similar behaviors to the $FOMRAC_3$, therefor only the controller with less dimension was plotted ($FOMRAC_3$). The $FOMRAC_2$ presents some oscillations in the transient responses under parameter changes, so it was not considered as a good choice.

Table 2
FOMRAC implementations details considering reduced order controllers.

Controller Order	Second order controller	Third order controller	Fourth order controller
Reference model	$G_m(s)=\frac{1.2}{s^3+5.2s^2+7s+1.2}$	$G_m(s)=\frac{1.2}{s^3+5.2s^2+7s+1.2}$	$G_m(s)=\frac{1.2}{s^3+5.2s^2+7s+1.2}$
Control law	$u(t)=\theta(t)^T\omega(t)$ $(\theta,\omega\in\mathbb{R}^4)$	$u(t)=\theta(t)^T\omega(t)$ $(\theta,\omega\in\mathbb{R}^6)$	$u(t)=\theta(t)^T\omega(t)$ $(\theta,\omega\in\mathbb{R}^8)$
Auxiliary signals	$\dot{\omega}_1(t)=\Lambda\omega_1(t)+lu(t)$ $\dot{\omega}_2(t)=\Lambda\omega_2(t)+ly_p(t)$ $\Lambda=-2$ $l=1$	$\dot{\omega}_1(t)=\Lambda\omega_1(t)+lu(t)$ $\dot{\omega}_2(t)=\Lambda\omega_2(t)+ly_p(t)$ $\Lambda=\begin{bmatrix}-1 & 0\\ 0 & -2\end{bmatrix}$ $l=[1\ -1]^T$	$\dot{\omega}_1(t)=\Lambda\omega_1(t)+lu(t)$ $\dot{\omega}_2(t)=\Lambda\omega_2(t)+ly_p(t)$ $\Lambda=\begin{bmatrix}-1 & 0 & 0\\ 0 & -2 & 0\\ 0 & 0 & -3\end{bmatrix}$ $l=[1\ -1\ 3]^T$

Table 3
Optimal controller design parameters for FOMRAC implementation considering reduced order controllers.

Controller Order	Second order controller	Third order controller	Fourth order controller
Optimal value of J	130.08	129.19	133.06
Optimal derivative orders	$\alpha_1=0.1076,\ \alpha_2=0.7375$ $\alpha_3=0.1892,\ \alpha_4=0.9969$ $\alpha_5=0.6272$	$\alpha_1=0.1494,\ \alpha_2=0.4447$ $\alpha_3=0.5170,\ \alpha_4=0.3393$ $\alpha_5=0.9609,\ \alpha_6=0.8717$ $\alpha_7=0.9084$	$\alpha_1=0.1015,\ \alpha_2=0.2421$ $\alpha_3=0.4792,\ \alpha_4=0.4888$ $\alpha_5=0.6906,\ \alpha_6=0.9668$ $\alpha_7=0.8197,\ \alpha_8=0.1649$ $\alpha_9=0.9110$
Optimal adaptive gains	$\gamma_1=3.2318,\ \gamma_2=0.2691$ $\gamma_3=0.4442,\ \gamma_4=1.8235$ $\gamma_5=4.3721$	$\gamma_1=3.4581,\ \gamma_2=3.0186$ $\gamma_3=1.5211,\ \gamma_4=0.8351$ $\gamma_5=0.6364,\ \gamma_6=0.2690$ $\gamma_7=5.0180$	$\gamma_1=3.3963,\ \gamma_2=5.1532$ $\gamma_3=5.3037,\ \gamma_4=4.9035$ $\gamma_5=0.0582,\ \gamma_6=0.5305$ $\gamma_7=0.8161,\ \gamma_8=0.9677$ $\gamma_9=2.0426$

Table 4
Optimal controller design parameters for IOMRAC implementation considering reduced order controllers.

Controller Order	Second order controller	Third order controller	Fourth order controller
Optimal value of J	173.03	118.99	118.85
Optimal adaptive gains	$\gamma_{1e}=0.1074,\ \gamma_{2e}=0.4772$ $\gamma_{3e}=0.3272,\ \gamma_{4e}=0.4377$ $\gamma_{5e}=0.2062$	$\gamma_{1e}=0.7501,\ \gamma_{2e}=10.9832$ $\gamma_{3e}=3.4511,\ \gamma_{4e}=3.9595$ $\gamma_{5e}=0.0606,\ \gamma_{6e}=0.06$ $\gamma_{7e}=11.9$	$\gamma_{1e}=0.9633,\ \gamma_{2e}=8.6064$ $\gamma_{3e}=4.4762,\ \gamma_{4e}=4.002$ $\gamma_{5e}=3.4331,\ \gamma_{6e}=0.06$ $\gamma_{7e}=0.2612,\ \gamma_{8e}=0.3744$ $\gamma_{9e}=12.9981$

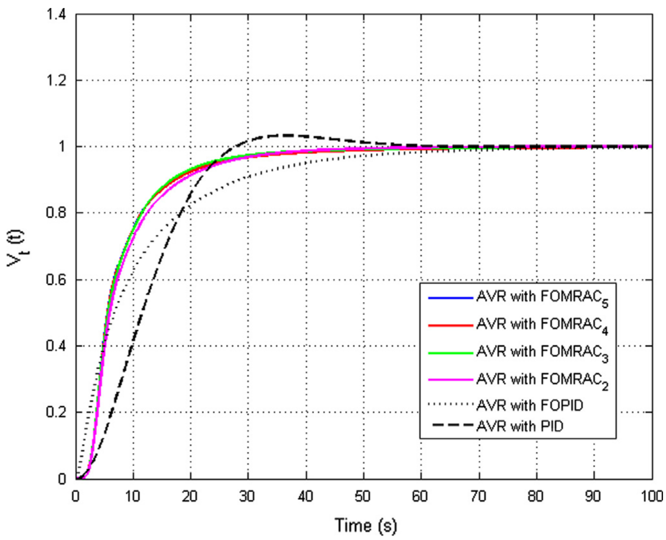

Fig. 6. Step responses of the AVR controlled by the four FOMRACs, PID and FOPID.

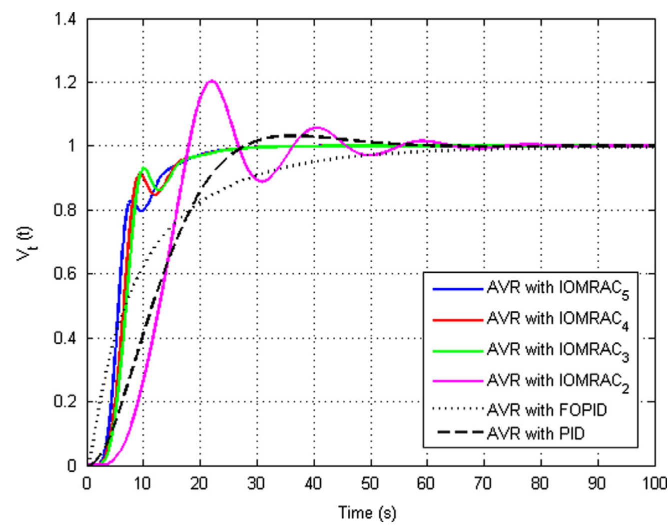

Fig. 8. Step responses of the AVR controlled by the four IOMRACs, PID and FOPID.

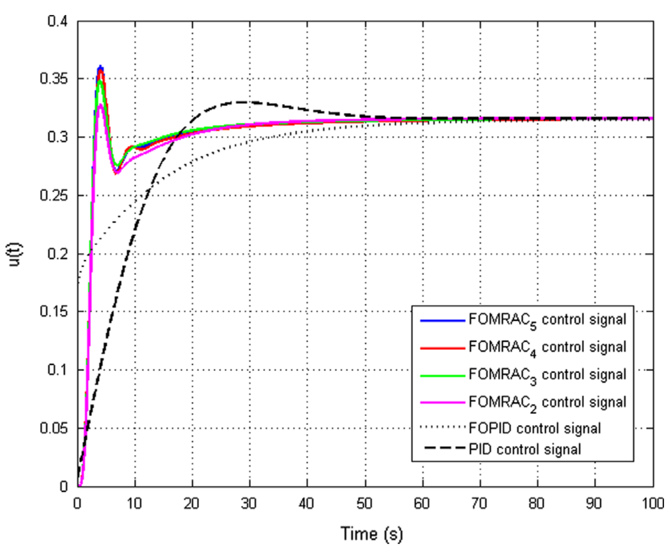

Fig. 7. Control signals of the AVR controlled by the four FOMRACs, PID and FOPID.

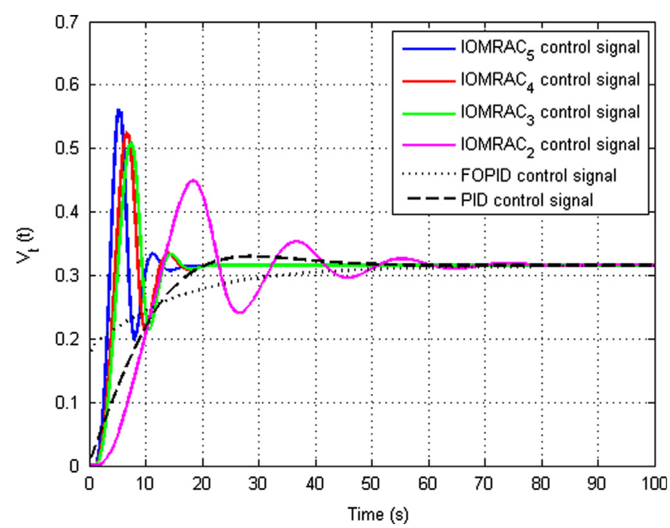

Fig. 9. Control signals of the AVR controlled by the four IOMRACs, PID and FOPID.

As can be seen in Fig. 10, $FOMRAC_3$ and $IOMRAC_3$ are similar, in the sense that both have a lesser overshoot and settling time than the FOPID and PID, when the change occurs. Both characteristics are highly desirable in an electric generation process control scheme. This improvement is obtained at the expense of a more demanding control signal, as can be seen in Fig. 11. The $IOMRAC_3$ control signal is, however, more demanding than the $FOMRAC_3$ control signal.

In the second robustness test, another uncertainty in the exciter model was assumed, where the transfer function varies from $V_f(s)/V_r(s)=1/0.5s+1$ to $V_f(s)/V_r(s)=1/0.5s+0.5$, at $t=100$ s.

Fig. 12 shows the terminal voltage response of the AVR with the $FOMRAC_3$ designed in this work and compared with FOPID, PID and $IOMRAC_3$, under parameter changes in the generator and the exciter. Again in this case, the fractional order adaptive controller

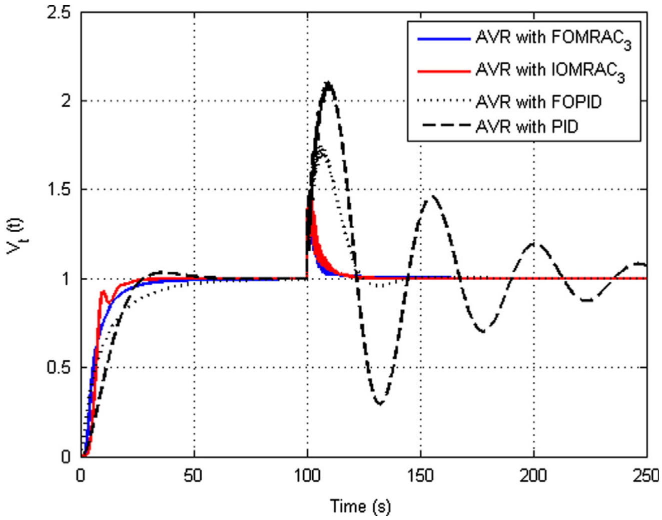

Fig. 10. Step responses of the AVR controlled by the $FOMRAC_3$, $IOMRAC_3$, PID and FOPID, under a parameter variation in the generator.

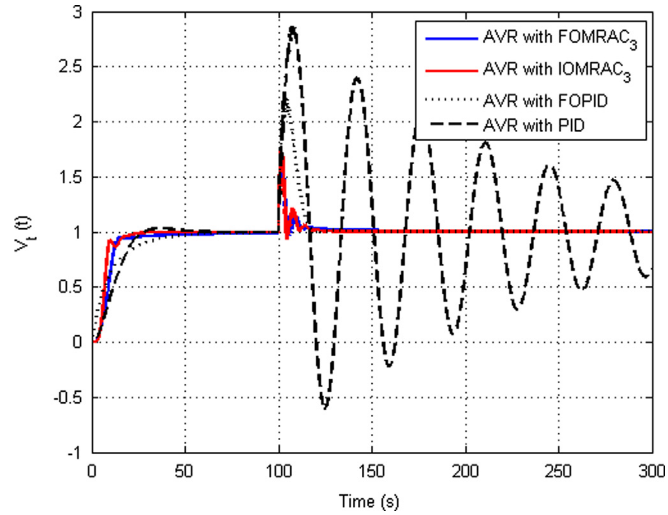

Fig. 12. Step responses of the AVR controlled by the $FOMRAC_3$, $IOMRAC_3$, PID and FOPID, under variations in the parameters of the generator and the exciter transfer function at $t=100$ s.

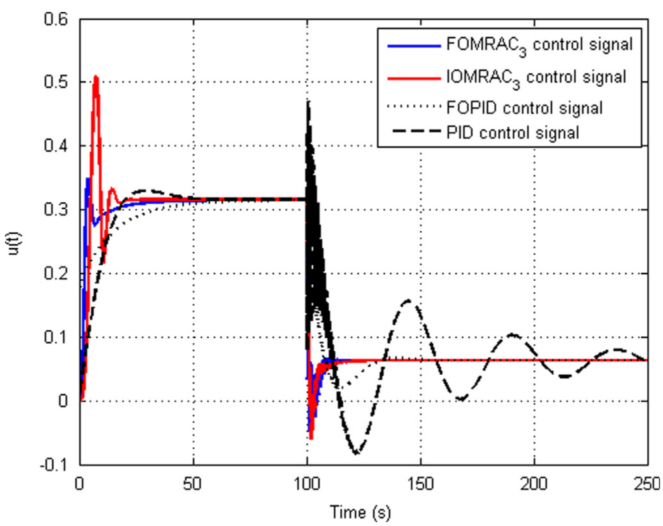

Fig. 11. Control signals of the AVR controlled by the $FOMRAC_3$, $IOMRAC_3$, PID and FOPID, under a parameter variation in the generator.

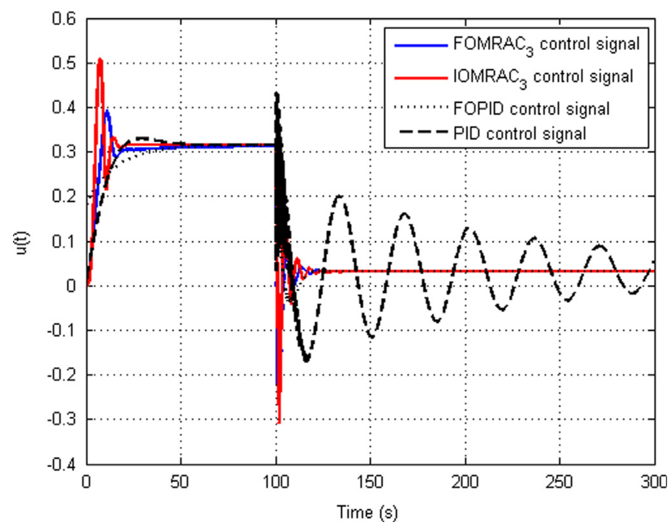

Fig. 13. Control signals of the AVR controlled by the $FOMRAC_3$, $IOMRAC_3$, PID and FOPID, under variations in the parameters of the generator and the exciter transfer function at $t=100$ s.

and the integer order adaptive controller have lesser overshoot and settling time than the FOPID and PID, and the control signal is more demanding in the case of the $IOMRAC_3$, as can be seen in Fig. 13.

5. Performance evaluation for the control scheme

Measuring the quality of a system response is a tough problem. One of the most serious difficulties is deciding what characteristics of the system response are important to take into account, and how they should be weighted [5].

In order to measure the quality of the FOMRAC proposed in this work, some performance indices taken from [5] are used. These performance indices are overshoot (M_p), rise time (t_r), settling time (t_s), steady-state error (E_{ss}), integral of the absolute error (*IAE*), integral of the squared error (*ISE*), integral of the time-weighted squared error (*ITSE*), integral of the squared input (*ISI*), and the sum of all these as another performance index.

Table 5 presents the corresponding values of these indices for the four FOMRAC's designed in this work, as well as for the four IOMRAC's designed, the FOPID and PID reported in [23]. In Table 5 it is observed that the fractional order adaptive controllers have smaller values than the FOPID and PID on almost all the performance indices. However, the ISI is smaller for the FOPID.

The four IOMRAC's designed have smaller indices values than the FOMRAC's, in almost all the performance indices, but the difference is not meaningful. The ISI has smaller values for the FOMRAC's than for the integer order counterparts, which is in agreement with the behavior obtained in the simulations.

It is interesting to note that the FOPID is not better than the PID in all performance indices, as can be seen in Table 5. This is due to the fact that they were tuned in Zamani et al. [23] using an specific criterion function, which includes only some of the performance indices of Table 5. However, a better result could be probably obtained for the FOPID over the PID, according to our performance indices, if the optimization procedure was made using a different functional J and/or different values for the weighting factors.

Table 5
Performance indices values for the four direct FOMRACs, the four direct IOMRACs, PID and FPID.

Performance Indice	M_p	t_r	t_s	E_{ss}	IAE	ITAE	ISE	ITSE	ISI	$\sum$
$FOMRAC_5$	0	13.19	28.83	0.02	7.82	59.10	4.59	14.65	9.59	137.79
$FOMRAC_4$	0	11.72	25.67	0.02	7.65	52.40	4.76	14.73	9.63	126.58
$FOMRAC_3$	0	11.36	26.60	0.02	7.89	54.99	5.02	15.87	9.61	131.36
$FOMRAC_2$	0	11.40	31.11	0.02	9.93	81.47	6.75	27.04	9.32	177.04
FOPID	0	26.69	53.57	0.02	11.87	172.57	5.37	31.52	8.75	310.36
PID	3.23	17.35	45.42	0.02	12.89	120.97	8.65	49.07	9.09	266.69
$IOMRAC_5$	0	9.15	21.63	0.02	7.08	35.87	5.03	14.46	9.93	103.17
$IOMRAC_4$	0	4.61	22.15	0.02	7.58	38.66	5.67	17.78	9.87	106.34
$IOMRAC_3$	0	4.74	22.31	0.02	7.88	40.71	6.00	19.73	9.83	111.22
$IOMRAC_2$	20.30	9.60	52.40	0.02	14.92	164.28	10.66	66.05	6.42	344.65

These performance indices offer a valuable tool for evaluating the controlled systems, but also offer a tool for control engineers for choosing a suitable controller for the AVR, based on the most relevant aspects they want to preserve in the controlled system.

6. Conclusions

This paper presents the application of a FOMRAC to an AVR. Selection of the controller design parameters is made through an optimization procedure using GA's. Considering different orders for the plant model, four FOMRAC's were designed and implemented, reducing the number of parameters adjusted in the control scheme. Simulation studies show an improvement in characteristics of the response of the controlled system and in robustness with respect to model uncertainties when using the FOMRAC, compared with the controlled system using FOPID and PID reported by other authors and using IOMRAC's designed in this work. Several performance indices used to evaluate the behavior show the advantages and disadvantages of the FOMRAC, making evident the utility of its use.

Acknowledgments

The results presented in this work were supported by CONICYT—Chile, under the Grants FB009 “Centro de Tecnología para la Minería” and FONDECYT 1120453, “Improvements of Adaptive Systems Performance by using Fractional Order Observers and Particle Swarm Optimization”.

References

[1] Chambers L. Practical handbook of genetic algorithms applications, vol. 1. CRC Press; 1995.
[2] Coelho LDS, de Meirelles Herrera B. Quantum gaussian particle swarm optimization approach for PID controller design in AVR system. In: IEEE international conference on systems, man and cybernetics, SMC 2008, 2008. p. 3708–13.
[3] Devaraj D, Selvabala B. Real-coded genetic algorithm and fuzzy logic approach for real-time tuning of proportional–integral–derivative controller in automatic voltage regulator system. IET Generation, Transmission and Distribution 2009;3:641–9.
[4] Duarte V, Costa JSD. Ninteger: a non-integer control toolbox for Matlab. In: Fractional derivatives and applications. Bordeaux: IFAC; 2004.
[5] Duarte-Mermoud MA, Prieto RA. Performance index for quality response of dynamical systems. ISA Transactions 2004;43:133–51.
[6] Fusco G, Russo M. Nonlinear control design for excitation controller and power system stabilizer. Control Engineering Practice 2011;19:243–51.
[7] Gaing Z. A particle swarm optimization approach for optimum design of PID controller in AVR system. IEEE Transactions on Energy Conversion 2004;19: 384–91.
[8] Kang HI, Kwon MW, Bae HG. PID coefficient designs for the automatic voltage regulator using a new third order particle swarm optimization. In: 2010 International conference on electronics and information engineering (ICEIE), 2010. p. 179–83.
[9] Kashki M, Abdel-Magid Y, Abido M. A reinforcement learning automata optimization approach for optimum tuning of PID controller in AVR system. In: Advanced intelligent computing theories and applications with aspects of artificial intelligence. Berlin, Heidelberg: Springer; 2008. p. 684–92.
[10] Kilbas A, Srivastava H, Trujillo J. Theory and applications of fractional differential equations. Elsevier; 2006.
[11] Kundur P. Power system stability and control. New York: McGraw-Hill; 1994.
[12] Ladaci S, Loiseau J, Charef A. Using fractional order filter in adaptive control of noisy plants. In: Third international conference on advances in mechanical engineering and mechanics, 2006.
[13] Ma J, Yao Y, Liu D. Fractional order model reference adaptive control for a hydraulic driven flight motion simulator. In: 41st southeastern symposium on system theory, 2009. p. 340–3.
[14] Narendra KS, Annaswamy AM. Stable adaptive systems. Dover Publications Inc.; 2005.
[15] Oonsivilai A, Pao-La-Or P. Optimum PID controller tuning for AVR system using adaptive tabu search. In: 12th WSEAS international conference on computers, 2008.
[16] Rahimian M, Raahemifar K. Optimal PID controller design for AVR system using particle swarm optimization algorithm. In: 24th Canadian conference on electrical and computer engineering (CCECE), 2011. p. 337–40.
[17] Shyama TR, Kumar RS, Shanmugasundaram V. Design of FGSPI controller based combined LFC and AVR of two area interconnected power generating system. International Journal of Engineering and Advanced Technology 2012;1:135–9.
[18] Suarez J, Vinagre BM, Chen Y. A fractional adaptation scheme for lateral control of an AGV. Journal of Vibration and Control 2008;14(9–10):1499–511.
[19] Valizadeh S, Jamali M-R, Lucas C. A particle-swarm-based approach for optimum design of BELBIC controller in AVR system. In: International conference on control, automation and systems, ICCAS 2008, 2008. p. 2679–84.
[20] Vinagre B, Petrás I, Podlubny I, Chen Y. Using fractional order adjustment rules and fractional order reference models in model-reference adaptive control. Nonlinear Dynamics 2002;29:269–79.
[21] Vinagre B, Podlubny I, Hernández A, Feliu V. Some approximations of fractional order operators used in control theory and applications. Fractional Calculus and Applied Analysis 2000;3:231–48.
[22] YaLi H, RuiKun G. Application of fractional-order model reference adaptive control on industry boiler burning system. In: International conference on intelligent computation technology and automation (ICICTA), 2010. p. 750–3.
[23] Zamani M, Karimi-Ghartemani M, Sadati N, Parniani M. Design of a fractional order PID controller for an AVR using particle swarm optimization. Control Engineering Practice 2009;17:1380–7.