Fractional Order PD and PID Position Control of an Angular Manipulator of 3DOF

Arturo Rojas–Moreno
Department of Electronic Engineering
University of Engineering and Technology
Lima 43, Peru
Email: arojas@utec.edu.pe

Victor Jara–Sandoval
Faculty of Electrical and Electronic Engineering
National University of Engineering
Lima 25, Peru
Email: jaravictor2000@yahoo.com

Abstract—This paper deals with the implementation of two position control systems: FOPID (Fractional Order Proportional Integral Derivative) and FOPD (FO Proportional Derivative), which are employed to control simultaneously angular positions of the base, arm, and forearm of an angular manipulator of 3DOF (3 Degrees of Freedom). The design of such control systems requires the derivation of the dynamic nonlinear model of the angular manipulator, as well as the determination of the corresponding control laws. Intensive simulation studies permitted to find out the initial values of the tuning parameters of the FOPD and FOPID controllers for realtime implementation. Good performances of the designed FOPD and FOPID control systems were verified via experimentation.

I. Introduction

Fractional calculus (FC) is a mathematical topic with more than 300 years old history. However, its application to physics and engineering has been published only in the recent years. Essentially, fractional calculus uses fractional integro–differential operators, which are a generalization of their integer counterparts. Fractional control system employs in its design FOPID controllers.

The number of research works related to the application of fractional control systems in many areas of science and engineering, including robotics, is continuously growing. For example, reference [1] presents the implementation of fractional order algorithms in the position/force hybrid control of robotic manipulators, while citation [2] deals with the study of the performance of FOPID controllers in a hexapod robot with joints at the legs having viscous friction and flexibility. A FOPID controller is investigated for a position servomechanism control system in [3], considering actuator saturation and the shaft torsional flexibility. In [4], the FOPID control of a 6 Degree Of Freedom hydraulic joystick–like master arm is investigated via simulation. Reference [5], develops a simulator that allows the integer and FO control of a dual-arm robotic system for cooperative manipulation of objects considering the existence of nonlinearities, while a FOPID control strategy for pneumatic position servosystem is presented in [6]. The paper [7] deals with the develop of an optimal procedure using a FO controller to control the position of a robotic system driven by DC motors.

II. Fractional Calculus (FC)

FC generalizes the notion of the derivative D^n of functions where the order of differentiation n is a non integer number [11], [10]. For the time–domain, FO derivative and integral operators are defined by means of the convolution operator, whereas for the Laplace–domain, the term s^α defines such operations, depending on the sign of the real number α. Following are presented the definitions of FO derivative and integral operators.

A. FO Integral of Riemann–Lioviulle

Let us define the following expression of the integral

$$D^{-1}f(x) = \int f(x)dx \tag{1}$$

Establishing the integration limits, integral D^{-1}, and for extension D^{-2}, can be expressed as

$$D^{-1}f(x) = \int_0^x f(t)dt$$

$$D^{-2}f(x) = \int_0^x \int_0^{t_2} f(t_1)dt_1 dt_2$$

Interchanging the order of the integration in D^{-2}, we obtain

$$D^{-2}f(x) = \int_0^x \int_{t_1}^x f(t_1)dt_2 dt_1 \tag{2}$$

Since $f(t_1)$ is not a function of t_2, then, this function can be extracted from the integral, such as

$$
\begin{aligned}
D^{-2}f(x) &= \int_0^x f(t_1)[\int_{t_1}^x dt_2]dt_1 \\
&= \int_0^x f(t)(x-t)dt \\
&= \int_0^x f(t_1)(x-t_1)dt_1
\end{aligned}
$$

Employing the same procedure, we can produce

$$
\begin{aligned}
D^{-3}f(x) &= \frac{1}{2}\int_0^x f(t)(x-t)^2 dt \\
D^{-4}f(x) &= \frac{1}{(2)(3)}\int_0^x f(t)(x-t)^3 dt \\
&\vdots \\
D^{-n}f(x) &= \frac{1}{(n-1)!}\int_0^x f(t)(x-t)^{n-1}dt \tag{3}
\end{aligned}
$$

Substituting in (3) the order $-n$ for an arbitrary λ number, as well as the factorial term for its corresponding gamma function, we obtain the following integral of Liouville

$$D^\lambda f(x) = \frac{1}{\Gamma(-\lambda)} \int_0^x \frac{f(t)}{(x-t)^{\lambda+1}} dt \qquad (4)$$

Take into account that the gamma function is defined as

$$\Gamma(z) = \int_0^\infty e^{-t} t^{z-1} dt \qquad (5)$$

which possesses the following characteristic:

$$\Gamma(z+1) = z! \qquad (6)$$

If: $\lambda \geq 0$ in (4), then this expression is non proper due to $t \to x$; therefore, $(x-t) \to 0$. The integral (4) diverges for all $\lambda \geq 0$. If: $-1 < \lambda < 0$, such integral converge, so that λ could be negative. Then, the equation (4) results a fractional integral.

Riemann generalized the integral of Liouville changing the limit 0 by b. The result is the following integral of Riemann–Louville

$$D^\lambda f(x) = \frac{1}{\Gamma(-\lambda)} \int_b^x \frac{f(t)}{(x-t)^{\lambda+1}} dt \qquad (7)$$

which is valid for $\lambda < 0$. After that, Riemann and Liouville substituted λ by $-\lambda$ in (8), resulting in the following positive fractional order integral

$$_b D_x^{-\lambda} f(x) = \frac{1}{\Gamma(\lambda)} \int_b^x \frac{f(t)}{(x-t)^{-\lambda+1}} dt \qquad (8)$$

where $\lambda > 0$ for all $t > 0$. Interchanging variables x by t and t by τ, we obtain the following generalized integral of Riemann–Liouville:

$$_b D_t^{-\lambda} f(t) = \frac{1}{\Gamma(\lambda)} \int_b^t \frac{f(\tau)}{(t-\tau)^{-\lambda+1}} d(\tau) \qquad (9)$$

which is valid for $\lambda > 0$ and $t > 0$.

B. FO Derivative of Riemann–Lioviulle

The classical derivative operator

$$D_t^n = \frac{d^n}{dt^n},$$

is generalized by the following FO derivative operator

$$_a D_t^{-\alpha}$$

where a and t are the limits and α is the order of the operation. To obtain the FO derivative, Riemann proposed to differentiate the following FO integral

$$_b D_x^{\alpha-n}$$

to produce

$$_b D_x^\alpha f(x) = D^n {}_b D_x^{\alpha-n} f(x) \qquad n > \alpha \qquad (10)$$

The last expression means to find out the fractional derivative of order α between the limits b and x. Performing the following change of variables: x by t, b by a and t by τ, results

$$_a D_t^\alpha f(t) = D^n {}_a D_t^{\alpha-n} f(t) = D^n {}_a D_t^{-(-\alpha+n)} f(t)$$

$$= D^n \frac{1}{\Gamma(n-\alpha)} \int_a^t (t-\tau)^{-\alpha+n-1} f(\tau) d(\tau)$$

$$= \frac{1}{\Gamma(n-\alpha)} \frac{d^n}{dt^n} \int_a^t (t-\tau)^{-\alpha+n-1} f(\tau) d(\tau) \qquad (11)$$

The equation (11) is known as the generalized FO derivative of Riemann–Liouville with a real positive FO value. Here, α is defined in the range: $n-1 < \alpha < n$. For $n = 1$ and $a = 0$, we have: $0 < \alpha < 1$. Then

$$_0 D_t^\alpha f(t) = \frac{1}{\Gamma(1-\alpha)} \frac{d^n}{dt^n} \int_0^t (t-\tau)^{-\alpha} f(\tau) d(\tau) \qquad (12)$$

C. The FOPID Controller

PIDOF controller arise with the purpose of improving the design specifications obtained with a conventional PID controller. The general idea is to take advantage of the simple structure and general application of PID controllers. A FOPID controller, also known as a $PI^\lambda D^\alpha$ controller, takes on the form

$$u(t) = K_p e(t) + K_i {}_0 D_t^{-\lambda} e(t) + K_d {}_0 D_t^\alpha e(t) \qquad (13)$$

where λ and α are the fractional orders of the controller and $e(t)$ is the system error. Note that the system error $e(t)$ replaces the general function $f(t)$ used before. The fundamental advantage of a FOPID controller lie in its two additional parameters: λ and α. By tuning properly such parameters, the robustness of the control system can be enhanced.

If parameters λ and α take the value of one, we obtain the conventional PID control algorithm in the time–domain

$$u(t) = P(t) + I(t) + D(t) \qquad (14)$$
$$e(t) = r(t) - y(t)$$
$$P(t) = K_p e(t) \qquad (15)$$
$$I(t) = \frac{K_p}{T_i} \int_0^t e(t) dt \qquad (16)$$
$$D(t) = K_p T_d \frac{de(t)}{dt} \qquad (17)$$

where K_p, $K_i = \frac{K_p}{T_i}$ and $K_d = K_p T_d$ represent the proportional, integral and derivative gains, respectively. Besides, T_i and T_d denote the integral and derivative times, respectively, $e(t)$ is the error signal of the system, $r(t)$ represents the reference or desired signal, and $y(t)$ is the controlled variable.

Substituting the integral and derivative parts od the PID controller given in (16) and (17) by its fractional counterparts, we obtain the following control law

$$u(t) = P(t) + I^\lambda(t) + D^\alpha(t) \qquad (18)$$
$$e(t) = r(t) - y(t)$$
$$P(t) = K_p e(t) \qquad (19)$$
$$I^\lambda(t) = K_i R_i(t) \qquad K_i = \frac{K_p}{T_i} \qquad (20)$$
$$R_i(t) = {}_0 D_t^{-\lambda} e(t)$$
$$= \frac{1}{\Gamma(\lambda)} \int_0^t (t-\tau)^{\lambda-1} f(\tau) d(\tau) \qquad (21)$$
$$D^\alpha(t) = K_d R_d(t) \qquad K_d = K_p T_d \qquad (22)$$
$$R_d(t) = {}_0 D_t^\alpha e(t)$$

$$= \frac{1}{\Gamma(m-\alpha)} \frac{d^m}{dt^m} \int_0^t (t-\tau)^{-\alpha+m-1} e(\tau) d(\tau) \tag{23}$$

The very well known PI, PD, PID controllers possess their FO counterparts: PI^λ, PD^α and $PI^\lambda D^\alpha$.

III. MANIPULATOR MODELLING

The 3DGOF (3 Degrees of Freedom) angular manipulator depicted in Fig. 1 consists of three DC servomotors driving the base, arm and forearm of such manipulator. Each servomotor possesses a reduction mechanism (a gear train) and an encoder to sense the angular position of the servo shaft. The described manipulator is a MIMO (Multiple Input Multiple Output) system due to its three inputs: control voltages u_1, u_2 and u_3 applied to the armature of each servo, and three outputs: angular positions q_1, q_2, and q_3 of the servos. Table I describes the variables and parameters of the manipulator, while Fig. 2 shows its coordinate diagram. See reference [8] for more details.
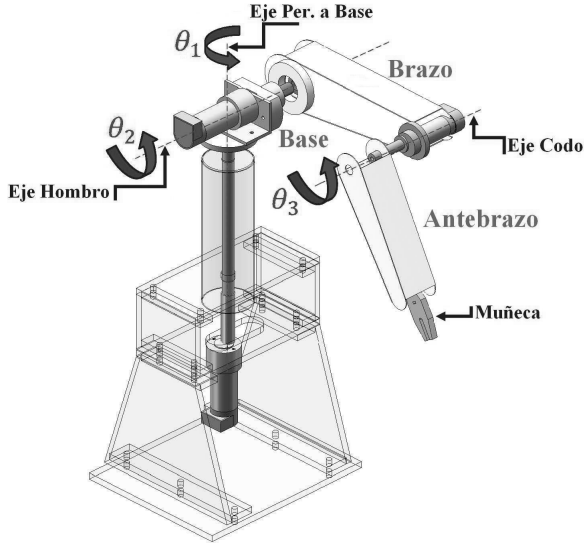

Fig. 1. Angular manipulator of 3DOF.

The dynamic model of the manipulator (the Lagrange model) can be obtained applying the Lagrange equations method, which consists in determining the kinetic (V_i) an potential (U_i) energies for each DOF q_i; that is, q_1 for the base, q_2 the arm, and q_3 the forearm, respectively.

The resulting function, known as the Lagrangian function, has the form

$$L = V(q_1, \ldots, q_r, \dot{q}_1, \ldots, \dot{q}_r) - U(q_1, \ldots, q_r, \dot{q}_1, \ldots, \dot{q}_r) \tag{24}$$

For this manipulator, the Lagrange equation are obtained as follows

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i \qquad i = 1, 2, \ldots, r \tag{25}$$

TABLE I. VARIABLES AND PARAMETERS OF THE MANIPULATOR. INDEX i TAKES VALUES 1, 2, AND 3 FOR THE BASE, ARM, AND FOREARM, RESPECTIVELY.

| Símbolo	Descripción	Valor	Unidad
x_i	Center of mass		m
q_i	Angular position		rad
H	Length of the base		m
M_i	Servomotor		N.m
T_i	Torque generated by M_i		N.m
m_1	Equivalent base mass	2.9188	Kg
m_2	Equivalent arm mass	0.89	Kg
m_3	Equivalent forearm mass	0.25	Kg
m_d	Disk mass		Kg
m_b	Bar mass		Kg
m_a	Arm mass		Kg
d	Disk width	0.0082	m
r_d	Disc radio	0.0427	m
b	Bar length	0.3378	m
a^2	Bar section		m^2
L_a	Arm length	0.24	m
L_b	Forearm length	0.24	m
L_1	Center of mass of the base	0.2936	m
L_1	Center of mass of the arm	0.1184	m
L_1	Center of mass of the forearm	0.12	m
J_{mi}	Moment of inertia (MI) of M_i,	71x10^{-7}	$Kg.m^2$
J_{eq}	Equivalent MI		$Kg.m^2$
J_{gi}	MI of the gears	0.053	$Kg.m^2$
J_1	MI of the base	0.0278	$Kg.m^2$
J_2	MI of the armo	0.0122	$Kg.m^2$
J_3	MI of the forearm	0.0015	$Kg.m^2$
B_{mi}	Friction constant of M_i	0.0001	N.m.s/rad
B_{eq}	Equivalent friction constant		N.m.s/rad
B_{gi}	Friction constant of the gers	0.01	N.m.s/rad
n	Gear rate of M_i	65.5	
r_a	Armature resistance of M_i	2.49	Ω
L_{arm}	Armature inductance of M_i	0.00263	H
V_{bi}	EM (Electro–Motriz) voltage of M_i		V
i_{ai}	Armature current		A
K_A	Amplifier gain	8.5	
K_m	Motor constant	0.0458	N.m/A
K_b	EM constant	0.0458	V.s/rad
u_i	Control voltage		V
g	gravitational constant	9.81	m/s^2
d_1	Length: base disc to M_2 shaft	0.045	m
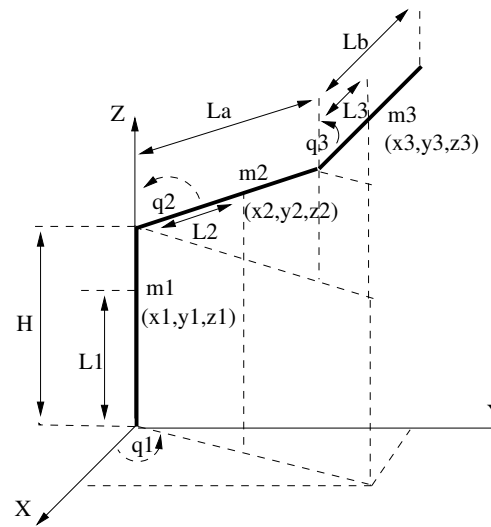
Fig. 2. Coordinates diagram of the manipulator.

where Q_i represents forces and torques external to the system.

Equations 24 and 25 takes on the form

$$L = V - U = V_1 + V_2 + V3 - (U_1 + U_2 + U_3) \quad (26)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_1}\right) - \frac{\partial L}{\partial q_1} = T_1 \quad (27)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_2}\right) - \frac{\partial L}{\partial q_2} = T_2 \quad (28)$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_3}\right) - \frac{\partial L}{\partial q_3} = T_3 \quad (29)$$

where V_1, V_2, and V_3 are the kinetic energies of the base, arm, and fore arm, respectively, U_1, U_2, and U_3 are the potential energies of the base, arm, and fore arm, accordingly, and T_1, T_2, and T_3 are the torques (neglecting losses) generated by the DC servomotors.

The kinetic and potential energies of the base are formulated as

$$V_1 = \frac{1}{2}J_1\dot{q}_1^2 \qquad U_1 = m_1 g L_1 \quad (30)$$

The expressions of the kinetic and potential energies of the arm are

$$V_2 = \frac{1}{2}J_2\dot{q}_2^2 + \frac{1}{2}m_2\left(\dot{x}_2^2 + \dot{y}_2^2 + \dot{z}_2^2\right)$$
$$U_2 = m_2 g\left(b + d + d_1 + L_2\cos q_2\right) \quad (31)$$

Also, the kinetic and potential energies of the forearm are expressed as

$$V_3 = \frac{1}{2}J_3\dot{q}_3^2 + \frac{1}{2}m_3\left(\dot{x}_3^2 + \dot{y}_3^2 + \dot{z}_3^2\right)$$
$$U_3 = m_3 g\left(b + d + d_1 + L_a\cos q_2 + L_3\cos q_3\right) \quad (32)$$

According to Fig. 2, arm positions x_2, y_2, and z_2 are given by

$$\begin{aligned}
x_2 &= (L_2\sin q_2)\cos q_1 \\
y_2 &= (L_2\sin q_2)\sin q_1 \\
z_2 &= (b + d + d_1 + L_2\cos q_2)
\end{aligned} \quad (33)$$

while the forearm positions x_3, y_3, and z_3 takes on the form

$$\begin{aligned}
x_3 &= (L_a\sin q_2 + L_3\sin q_3)\cos q_1 \\
y_3 &= (L_a\sin q_2 + L_3\sin q_3)\sin q_1 \\
z_3 &= (b + d + d_1 + L_a\cos q_2 + L_3\cos q_3)
\end{aligned} \quad (34)$$

Therefore, V_2 y V_3 result in

$$V_2 = \frac{1}{2}J_2\dot{q}_2^2 + \frac{1}{2}m_2\left(\dot{q}_1^2 + L_2{}^2\dot{q}_2^2 + L_2{}^2\dot{q}_1^2\sin^2 q_2 + 2L_2\dot{q}_1^2\sin q_2\right) \quad (35)$$

$$\begin{aligned}
V_3 &= \frac{1}{2}J_3\dot{q}_3^2 + \frac{1}{2}m_3\left(L_a^2\dot{q}_2^2 + L_3^2\dot{q}_3^2\right) \\
&+ \frac{1}{2}m_3\left(2L_3 L_a\dot{q}_2\dot{q}_3\cos(q_2 - q_3) + L_a^2\sin^2 q_2\dot{q}_1^2\right) \\
&+ \frac{1}{2}m_3\left(L_3^2\sin^2 q_3\dot{q}_1^2 + 2L_3 L_a\sin q_2\sin q_3\dot{q}_1^2\right) \quad (36)
\end{aligned}$$

Operating in concordance with (27), (28), and (29), we can obtain the corresponding torques T_1, T_2 y T_3.
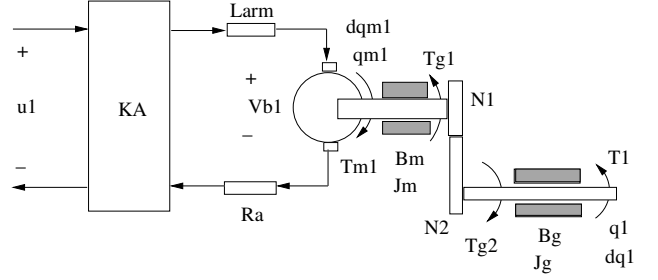

Fig. 3. Servomotor DC with its gear train.

A. Modelling the DC Servomotors

The three DC servomotors M_1, M_2 y M_3 possess the same characteristics. Fig. 3 shows one of this servomotor with its gear train. Neglecting the inductance armature L_{arm}, the voltage equation of the armature results

$$K_A u_1 = R_a i_{a1} + V_{b1} \quad (37)$$

The EMF voltage is proportional to the servomotor speed

$$V_{b1} = K_b \dot{q}_{m1} \quad (38)$$

The relation between speeds q_{m1} and q_1 is given by

$$q_{m1} = n q_1 \qquad n = \frac{N_2}{N_1} > 1 \quad (39)$$

where N_1 and N_2 are the number of tooth of each gear. The EMF can be written as

$$V_{b1} = K_b n \dot{q}_1 \quad (40)$$

On the other hand, the motor torque T_{m1} must overcome the inertia and viscous torques torques of the servomotor as follows

$$T_{m1} = J_m n\ddot{q}_1 + B_m n\dot{q}_1 + T_{g1} \quad (41)$$

where T_{g1} is the input torque to the gear train. Assuming an ideal gear train, the energy conservation principle establishes that the performed work on each side of the gear train must be the same; that is,

$$T_{g1}q_{m1} = T_{g2}\frac{q_{m1}}{n} \qquad T_{g2} = nT_{g1} \quad (42)$$

Therefore

$$nT_{g1} = J_g\ddot{q}_1 + B_g\dot{q}_1 + T_1 \quad (43)$$

where J_m and J_g represents the moment of inertia of the servomotor and the gear train, respectively, while B_m and B_g are friction constants of the servomotor and gear train, respectively. It is well known that the servomotor torque is proportional to the armature current i_{a1}

$$T_{m1} = K_m i_{a1} \quad (44)$$

Working with equations (37), (40), (43), and (44), it can be showed that T_1 has the form

$$T_1 = -J_{eq}\ddot{q}_1 - \left(B_{eq} + \frac{n^2 K_m K_b}{R_a}\right)\dot{q}_1 + \frac{nK_m K_A}{R_a}u_1 \quad (45)$$

$$J_{eq} = n^2 J_m + J_g \qquad B_{eq} = n^2 B_m + B_g$$

The procedure used to obtain T_1 for the base can also be employed to obtain T_2 and T_3; that is

$$T_2 = -J_{eq}\ddot{q}_2 - \left(B_{eq} + \frac{n^2 K_m K_b}{R_a}\right)\dot{q}_2 + \frac{nK_m K_A}{R_a}u_2 \quad (46)$$

$$T_3 = -J_{eq}\ddot{q}_3 - \left(B_{eq} + \frac{n^2 K_m K_b}{R_a}\right)\dot{q}_3 + \frac{nK_m K_A}{R_a}u_3 \quad (47)$$

Equating torques (45), (46), and (47) with torques (27), (28), and (29), we can obtain the control voltages u_1, u_2 y u_3 instead of the control torques T_1, T_2 y T_3

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{P}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{d}(\mathbf{q}) = \mathbf{u} \quad (48)$$

$$\begin{bmatrix} M_{11} & 0 & 0 \\ 0 & M_{22} & M_{23} \\ 0 & M_{32} & M_{33} \end{bmatrix}\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} +$$

$$\begin{bmatrix} P_{11} & P_{12} & P_{13} \\ P_{21} & P_{22} & P_{23} \\ P_{31} & P_{32} & P_{33} \end{bmatrix}\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ d_{21} \\ d_{31} \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$$

$$M_{11} = \frac{R_a}{nK_m K_A}\left(J_1 + J_{eq} + m_2 L_2^2\sin^2 q_2 +\right.$$

$$\left. m_3 L_a^2\sin^2 q_2 + m_3 L_3^2\sin^2 q_3 + 2m_3 L_a L_3\sin q_2\sin q_3\right)$$

$$M_{22} = \frac{R_a}{nK_m K_A}\left(J_2 + J_{eq} + m_2 L_2^2 + m_3 L_a^2\right)$$

$$M_{23} = M_{32} = \frac{R_a}{nK_m K_A}\left(m_3 L_3 L_a \cos(q_2 - q_3)\right)$$

$$M_{33} = \frac{R_a}{nK_m K_A}\left(J_3 + J_{eq} + m_3 L_3^2\right)$$

$$P_{11} = \frac{R_a}{nK_m K_A}\left(B_{eq} + \frac{n^2 K_m K_b}{R_a}\right)$$

$$P_{12} = \frac{R_a}{nK_m K_A}\left(2m_2 L_2^2\sin q_2\cos q_2\dot{q}_1 +\right.$$

$$\left. 2m_3 L_3 L_a\cos q_2\sin q_3\dot{q}_1 + 2m_3 L_a^2\sin q_2\cos q_2\dot{q}_1\right)$$

$$P_{13} = \frac{R_a}{nK_m K_A}\left(2m_3 L_3^2\sin q_3\cos q_3\dot{q}_1 +\right.$$

$$\left. 2m_3 L_3 L_a\sin q_2\cos q_3\dot{q}_1\right)$$

$$P_{21} = -\frac{R_a}{nK_m K_A}\left(m_2 L_2^2\sin q_2\dot{q}_1 +\right.$$

$$\left. m_3 L_a^2\sin q_2\cos q_2\dot{q}_1 + m_3 L_a L_3\sin q_3\cos q_2\dot{q}_1\right)$$

$$P_{22} = \frac{R_a}{nK_m K_A}\left(B_{eq} + \frac{n^2 K_m K_b}{R_a}\right)$$

$$P_{23} = \frac{R_a}{nK_m K_A}\left(m_3 L_3 L_a\dot{q}_3 \sin(q_2 - q_3)\right)$$

$$P_{31} = -\frac{R_a}{nK_m K_A}\left(m_3 L_3^2\sin q_3\cos q_3\dot{q}_1 +\right.$$

$$\left. m_3 L_3 L_a\sin q_2\cos q_3\dot{q}_1\right)$$

$$P_{32} = \frac{R_a}{nK_m K_A}\left(m_3 L_3 L_a\dot{q}_3 \sin(q_3 - q_2)\right)$$

$$P_{33} = \frac{R_a}{nK_m K_A}\left(B_{eq} + \frac{n^2 K_m K_b}{R_a}\right)$$

$$d_{21} = -\frac{R_a}{nK_m K_A}\left(m_2 g L_2\sin q_2 + m_3 g L_a\sin q_2\right)$$

$$d_{31} = -\frac{R_a}{nK_m K_A}\left(m_3 g L_3\sin q_3\right)$$

In (48), $\mathbf{M}$ and $\mathbf{P}$ are the inertia and Coriolis matrices of order three, respectively, $\mathbf{d}$ is the gravitational force vector of order three, while $\mathbf{u}$ is the voltage control vector of order three

IV.   FOPID CONTROLLER DESIGN

The expression of the FOPID (or $PI^\lambda D^\alpha$) controller was formulated in subsection II-C. The control forces corresponding to algorithms $PD^\alpha(t)$ (FOPD) and PI^λ (FOPI) have the form

$$\begin{aligned} u(t) &= P(t) + D^\alpha(t) & e(t) = r(t) - y(t) \quad (49) \\ u(t) &= P(t) + I^\lambda(t) & e(t) = r(t) - y(t) \quad (50) \end{aligned}$$

A. Computing the FO Derivative R_d

The FO Derivative R_d is given by (23) for $\alpha = 0.5$ and $m = 1$. It can be computed using the following MATLAB program [8]

```
% DFRL.m COMPUTING THE FO DERIVATIVE Rd
  clear all; close all; clc;
% RANGE: 0 < alpha < 1 AND LIMITS 0 AND t
  alpha = 0.5; e=r-y; % SYSTEM ERROR
  syms y T Vd r n; % SYMBOLIC CALCULUS
  Vd=[.1,.2,.3,.4,.5,.6,.7,.8,.98];
  n=length(Vd); % LENGTH OF VECTOR Vd
  for k=1:n
    Rd(k)=(1/gamma(1-Vd(k)))*...
    ((T-y)^(-(Vd(k))))*(r-y);
    Rd(k)=diff((int(Rd(k),y,0,T)));
    Rd(k)=VPA(Rd(k));
  end
% Rd=(0.1880*(3.0*r-2.0*T))/T^(1/2)- ...
% 0.7522*T^(1/2); % SELECTED Rd
```

B. Computing the FO Integral R_i

The FO integral R_i expressed in (21) for values of $\lambda = 0.5$ and $a = 0$, can be computed using the following MATLAB program [8]

```
% IFRL.m COMPUTING THE FO INTEGRAL Ri
  clear all; close all; clc;
% RANGE: 0 < alpha < 1 AND LIMITS 0 AND t
  alpha = 0.5; e=r-y; % SYSTEM ERROR
  syms y T Vi r n;
  Vi=[.1,.2,.3,.4,.5,.6,.7,.8,.9,1];
  n=length(Vi); % LENGTH OF VECTOR Vi
  for k=1:n
  Ri(k)=int(Ri(k),y,0,T); % FUNCTION Ri
  Ri(k)=VPA(Ri(k))
  end
% Ri=0.3761*T^(1/2)*(3.0*r-2.0*T);
% SELECTED Ri
```

V. SIMULATION STUDIES

The simulation procedure consists on applying the FOPD and FOPID control algorithms to the nonlinear dynamic model of the manipulator to demonstrate that the controlled angular positions of the base, arm and forearm met the desired design specifications: settling time less than 1 second, null steady state error, and percentage overshoot less tan 2%. All the programs where written in MATLAB code. Such programs can be found in reference [8].

The results showed in Figs. 4 and 5 demonstrate that the established design specifications are met for the designed FOPD and FOPID control systems. The FOPD control law is generated by the following portion of the simulation program:

```
% FO DERIVATIVE CONTROL LAW
  ud1=Kd1*((0.1880*(3.0*r1-2.0*T))*...
      (T.^-(1/2))-0.7522*T.^(1/2));
  ud2=Kd2*((0.1880*(3.0*r2-2.0*T))*...
      (T.^-(1/2))-0.7522*T.^(1/2));
  ud3=Kd3*((0.1880*(3.0*r3-2.0*T))*...
      (T.^-(1/2))-0.7522*T.^(1/2));
% CONTROL LAW PD^{alpha}
  u1=Kp1*e1+ud1; U1(k)=u1;
  u2=Kp2*e2+ud2; U2(k)=u2;
  u3=Kp3*e3+ud3; U3(k)=u3;
```

In the same way, the FOPID control law is generated by the following portion of the simulation program:

```
% FO DERIVATIVE CONTROL LAW
  ud1=Kd1*((0.1880*(3.0*r1-2.0*T))*...
      (T.^-(1/2))-0.7522*T.^(1/2));
  ud2=Kd2*((0.1880*(3.0*r2-2.0*T))*...
      (T.^-(1/2))-0.7522*T.^(1/2));
  ud3=Kd3*((0.1880*(3.0*r3-2.0*T))*...
      (T.^-(1/2))-0.7522*T.^(1/2));
% FO INTEGRAL CONTROL LAW
  ui1=Ki1*0.3761*(T.^(1/2))*(3.0*r1 - 2.0*T);
  ui2=Ki2*0.3761*(T.^(1/2))*(3.0*r2 - 2.0*T);
  ui3=Ki3*0.3761*(T.^(1/2))*(3.0*r3 - 2.0*T);
% CONTROL LAW PI^{gamma}D^{alpha}
  u1 = Kp1 * e1 + ui1 + ud1; U1(k)=u1;
  u2 = Kp2 * e2 + ui2 + ud2; U2(k)=u2;
  u3 = Kp3 * e3 + ui3 + ud3; U3(k)=u3;
```

VI. EXPERIMENTAL SETUP

Fig. 6 shows the experimental setup of the FOPID position control system. The base, arm and forearm of the angular manipulator (see Fig. 1) are driven by three DC servomotors. Each of them possesses a reduction mechanism and a quadrature encoder to sense the angular position of the servo shaft.

A PC was used to process the FOPID control algorithms. A Data Acquisition (DAQ) Board and a DAQ-STC (System Timing Controller) device, both from NI (National Instrument), were employed to process inputs and outputs system signals. The DAQ-STC device was used to sense three angular positions from the encoders (the inputs of the system), while the DAQ Board was employed to input to the PC three angular position signals (from the DAQ-STC device) and to output three control voltages. Each control voltage is in turn amplified by a PWM (Pulse Width Modulation) Power Amplifier. Next, eeach amplifier outputs a control force in form of a DC voltage
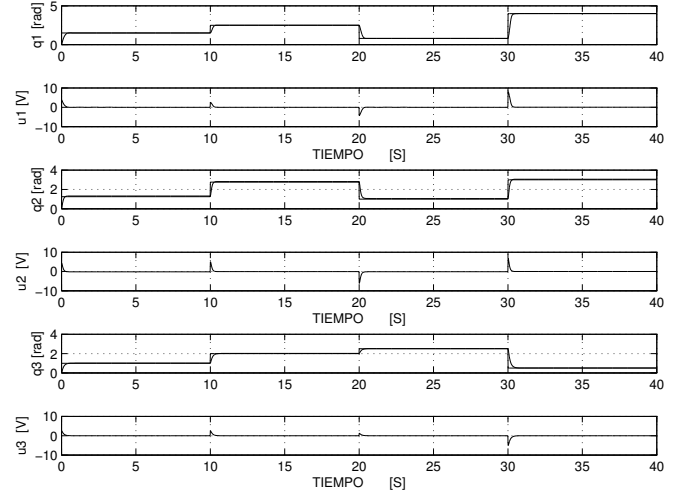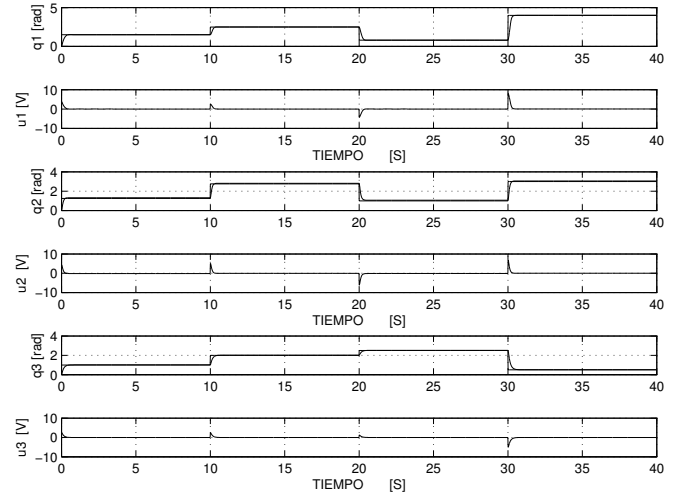

Fig. 4. Controlled angular positions q_1, q_2, and q_3 of the manipulator using FOPD control laws u_1, u_2, and u_3, respectively, for step type reference signals.

Fig. 5. Controlled angular positions q_1, q_2, and q_3 of the manipulator using FOPID control laws u_1, u_2, and u_3, respectively, for step type reference signals.

to supply the armature of its corresponding DC servomotor. Fig. 7 shows the block diagram of the FOPID control system, where r_1, r_2, and r_3 are de desired signals, y_1, y_2, and y_3 are the controlled signals (angular positions of the base, arm and forearm), and u_1, u_2, and u_3 are the control signals.

VII. CONTROL SOFTWARE

The control software was written in LabVIEW code. The structure of this program comprises five parts: the Human Machine Interface to change off– and on–line process parameters and constants (Fig. 8); the program portion to load and unload historic data; the part to storage historic data; the control algorithm (Fig. 9); and, the portion to initialize variables. Reference [8] describes in detail the control software used in this work. Some other examples of designed control software can be found in [9].

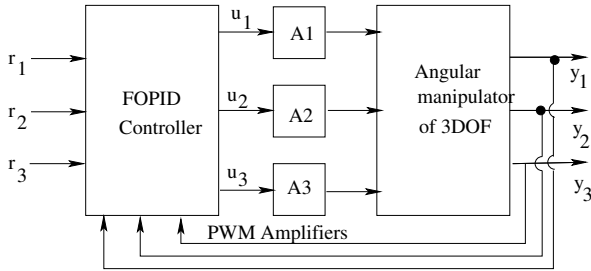Fig. 6.   The experimental setup.

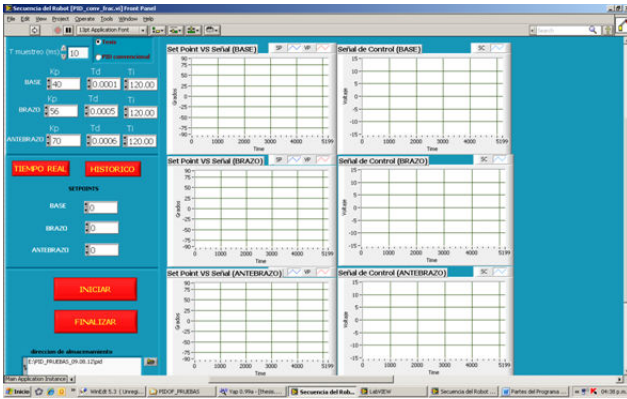Fig. 7.   Block diagram of the FOPID control system.

Fig. 8.   Human Machine Interface of the control software.

VIII.   EXPERIMENTAL RESULTS

Two designed FO control systems were verified via experimentation: position control of the manipulator employing FOPD and FOPID controllers, respectively. Design specifications were set to: null steady state error, overshoot percentage less than 2%, and settling time less than 1 second.

Fig. 10 shows the controlled positions of the base, arm, and forearm of the manipulator. The corresponding control voltages are also illustrated. Gains K_p of the FOPD controllers for the base, arm, and forearm were set to 40, 52, and 70, respectively, while the derivative time parameter T_d for the same controllers
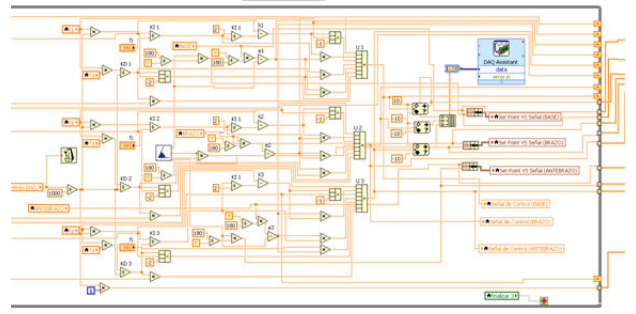

Fig. 9.   Portion of the program showing the control algorithm.

took values of 0.0001, 0.0005 and 0.0006, respectively.

The fractional parameters λ and α were set to the value of 0.5 for all cases. Such values were determined in the simulation phase using the trial–and–error method. The results illustrated in Fig. 10 demonstrate that the design specifications are fulfilled.
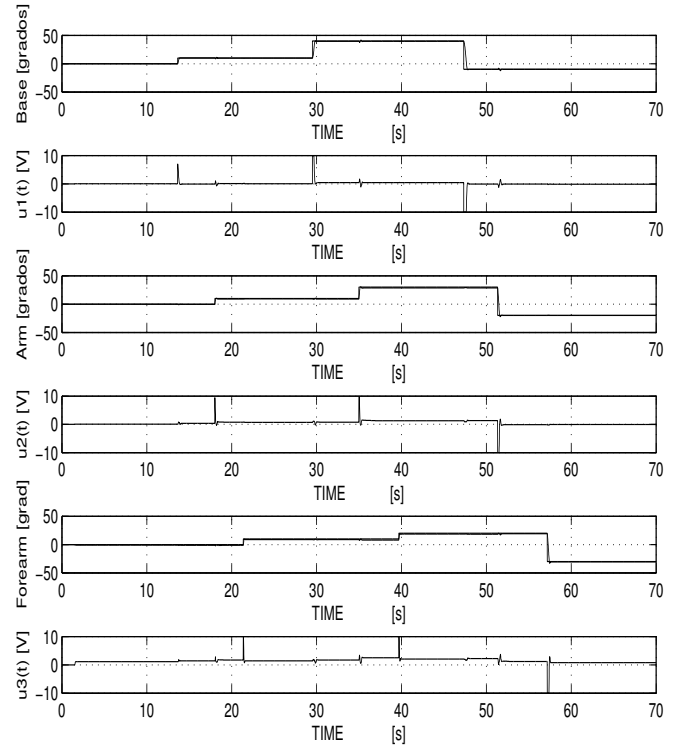

Fig. 10.   Controlled angular positions of the base, arm, and forearm by means of a FOPD control system.

Figs. 11, 12, and 13 illustrate the controlled positions of the base, arm, and forearm of the manipulator. The corresponding control voltages are also depicted. Gains K_p of the FOPD controllers for the base, arm, and forearm were set to 40, 56, and 70, respectively, while the derivative time parameter T_d for the same controllers took values of 0.0001, 0.0005, and 0.0006, respectively. The integral time parameter T_i was set to 120 for all cases.

The fractional parameters λ and α, as above, were set to

the value of 0.5 for all cases. Such values, were determined in the simulation phase using trial–and–error. The results illustrated in Figs. 11, 12, and 13 met the established design specifications.
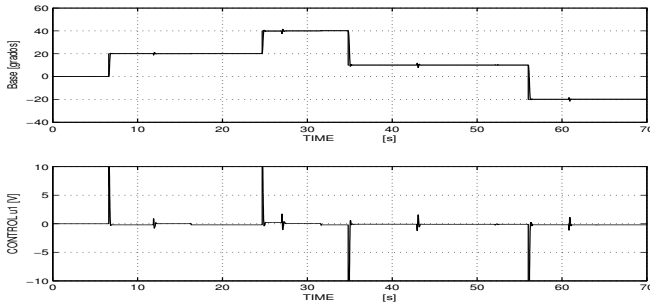

Fig. 11. Controlled angular position of the manipulator base using a FOPID controller.
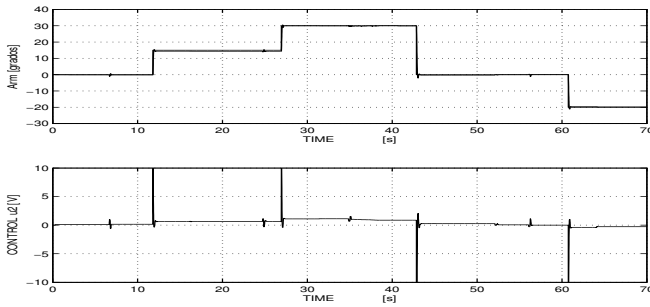

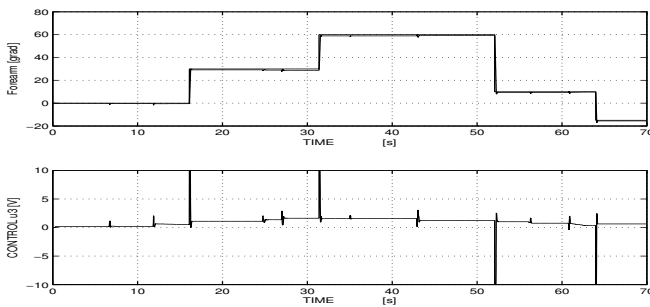Fig. 12. Controlled angular position of the manipulator arm using a FOPID controller.

Fig. 13. Controlled angular position of the manipulator forearm using a FOPID controller.

IX. Conclusions

In view of the results of section VIII, the main goal of this work has been achieved: to control simultaneously the base, arm, and forearm positions of an angular manipulator using FOPD and FOPID control systems. Figs. 10, 11, 12, and 13 demonstrate that the previously established design specifications were met: null steady state error, percentage overshoot less than 2%, and settling time less than 0.5 seconds.

The determination of the dynamic model of the manipulator (section III) was required to simulate the behavior of the

designed FOPD and FOPID control systems (Figs. 4 and 5) using the selected FO tuning parameters K_p, K_i, K_d, γ, and α, which were chosen by trial–and–error. Such parameters were also used as initial parameters in the experimentation phase.

The designed FOPID controller (section IV) is inherently nonlinear. This fact results convenient in the design of nonlinear systems. FOPID controller possess five tuning parameters. It has been demonstrated in section VIII, that a good choice of such parameters could solve regulation and robustness problems of a control system.

As a future work, it is recommendable to embed the designed FOPID control systems for industrial purposes. A CRIO (Compact Reconfigurable Input Output) device, among other possibilities, could be used for such a purpose.

References

[1] N. M. Fonseca Ferreira and J. A. Tenreiro Machado, *Fractional-Order Hybrid Control of Robotic Manipulators*, Proceedings of ICAR 2003. The 11th International Conference on Advanced Robotics Coimbra, Portugal, June 30 – July 3, 2003.

[2] M. F. Silva, J. A. T. Machado, and A. M. Lopes, *Fractional Order Control of a Hexapod Robot*, Nonlinear Dynamics 38: 417–433, 2004.

[3] Dingyü Xue, Chunna Zhao, and Yang Quan Chen, *Fractional Order PID Control of A DC-Motor with Elastic Shaft: A Case Study*, Proceedings of the 2006 American Control Conference Minneapolis, Minnesota, USA, June 14-16, 2006.

[4] A. L.Bensick, and A. M. Lopes, *Fractional Order Adaptive Control for Manipulator Systems*, Proceedings of the 9th WSEAS International Conference on Simulation, Modelling and Optimization.

[5] C. M. Figueiredo and N. M. Fonseca Ferreira, *Fractional-Order Control of a Dual-Arm Robotic System*, ENOC 2011, 24-29 July 2011, Rome, Italy.

[6] Cao Junyi and Cao Binggang, *Fractional-Order Control of Pneumatic Position Servosystems*, Hindawi Publishing Corporation, Mathematical Problems in Engineering, Volume 2011, Article ID 287565, 14 pages, doi:10.1155/2011/287565.

[7] Mihailo Lazarević, *Fractional Order Control of a Robot System Driven by DC Motors*, Scientific Technical Review, 2012,Vol.62,No.2,pp.20-29.

[8] V. O. Jara, *Implementación de un Sistema de Control de Posición PIDOF para un Manipulador Angular de 3GDL*, Tesis de Maestría en Automática e Instrumentación, Universidad Nacional de Ingeniería, FIEE, Lima, 2012.

[9] A. Rojas Moreno, *Control No Lineal Multivariable. Aplicaciones en Tiempo Real*, Editorial Universitaria UNI (EDUNI), ISBN 978–612–4072–18–5, 2012. Ingeniería, FIEE, Lima, 2012.

[10] D. Valerio, *Fractional robust system control*, PhD Thesis, Instituto Superior Tecnico, Lisboa (Portugal), 2005.

[11] M. Weilbeer, *Efficient Numerical Methods for Fractional Differential Equations and Their Analytical Background*, PhD Thesis, Technische Universitat Braunschweig, Germany, 2005.