Simulation of a Master-Slave Tele-Operated System for People with Muscular Atrophy in Upper Limbs.

Francesco García Luna, *Researcher, UACJ,* and Karla Gómez Bull, *Researcher, UACJ*

Abstract—In the present paper it was developed and programmed a simple 6 DoF tele-operated slave system that can track in real-time a master system. The arm's length parameters were calculated using a database of 196 male and female students from 18 to 22 years old. A classic control scheme were used in the slave system for the position tracking. The master system movement modifies the end effector's Euclidean position of the slave system. Experimental results of the system are provided including the desired trajectory and the robot behavior, showing the trajectory error.

Index Terms—Manipulator, Tele-operation, Anthropometry, Computer Science.

1 Introduction

Teleoperation is a set of technologies through which a person can perform tasks remotely, these systems avoid the exposure of people to dangerous and / or inaccessible sites [1]. Through tele-operation systems, it is possible to expand human senses and skills, with the aim of controlling tasks of a robot [2]. Controlling robot arms involves a number of challenges, and limitations in the capabilities of human operators and technical challenges in effectively translating human operator commands into robot actions [3].

Nowadays, robot arms have much attention in different fields [4]; as industrial security [5], service, power plant maintenance, space exploration, surgery [6], automation for industry, disaster recovery, virtual reality games and entertainment [7]. In 2016, a robot arm was designed with sixdegree of freedom in order to handle hazardous materials for chemical and nuclear industries, in order to protect workers from exposures to these risks [5].

In [6] a robot arm system applying Robot Service Network Protocol was developed, with the aim to obtain precise tasks in service fields. Later in [8], developed a robot arm controlled by manual operation of human operator using a stereo camera, considering motions for a working robot arm as rough and accurate motion.

Another application for robotic arms is for the industrial field, In [9] proposed a new method to manipulate and control a robotic arm within an industrial space in real time, which has 4 degrees of freedom on the shoulder, elbow and wrist. An exoskeletal master was designed to tele-operate a robot with two arms, it was used an industrial robot for assembly work, with nine degrees of freedom [7].

In 2013, a humanoid robot for upper limbs was performed using a motion tracking system to execute tasks in search and rescue missions, such cutting through walls, tasks that involved manual labor or require the use of power tools. It was named HUBO and had six degrees of freedom [10].

Tele-operated robot arms have been used in the surgery field too to perform a variety of invasive procedures, in 2014, there was an study with a robot focusing on the movement of surgeons' hands and arms, employing theories and methods from the study of human motor control [11]. A robotic arm was proposed too during 2014, using gesture and position tracking systems, in order to incorporate robotic systems into the home environment [12].

Robotic arms are currently enabling people with upper extremity disabilities to perform daily activities on their own [13]. According of this, [14] created a robotic platform for rehabilitation of people who suffer from progressive muscular degenerative disorders and neurological deficits.

2 Parameters Estimation

A representative sample size was calculated for finite and known populations, according to the total number of students enrolled in the University, with 95% of confidence. This sample was formed by 198 subjects, 46 women and 153 men who were invited to be part of the study.

Twelve anthropometric dimensions were taken for every participant (as shown in figure 1), upper limb length, clavicle-shoulder length, shoulder-elbow length, and shoulder-wrist length. This anthropometric data was taken according to standard methods [16]. An anthropometric kit Rosscraft model Centurion was employed to obtain the anthropometric dimensions. Wrist, arm and forearm perimeters were measured with a flexometer included in

- *F. García belongs to the Mechatronics Engineering program of the Industrial and Manufacturing Department, Ciudad Juárez Autonomous University, Campus CU, Ciudad Juárez, CHI. E-mail: francesco.garcia@uacj.mx*
- *K. Gómez belongs to the Industrial Engineering program of the Industrial and Manufacturing Department, Ciudad Juárez Autonomous University, Campus CU, Ciudad Juárez, CHI. E-mail: karla.gomez@uacj.mx*

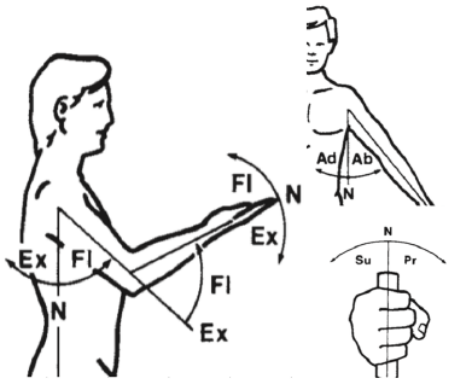

Figure 1. Range of motion for upper limb where Ex, Fl, Ad, Ab, Su and Pr means extension, flexion, adduction, abduction, supination and pronation respectively [15]

Table 1
Percentiles for anthropometric data

Anthropometric Dimension	$\mu \pm \sigma$	**P95**
Clavicle-shoulder length	19.12 ± 2.51	23.68
Shoulder-elbow length	35.85 ± 3.48	42.12
Elbow-wrist length	28.93 ± 2.07	32.7

the anthropometric kit. All measurements were taken in the Ergonomics and Methods Lab at the UACJ Campus CU.

These mentioned dimensions were taken with the purpose of simulating the main movements of the upper limb [16]:

- Shoulder: Flexion/extension, abduction/adduction.
- Elbow: Flexion/extension.
- Forearm: Pronation/supination.
- Wrist: Flexion/extension, radial/cubital deviation.

Subjects were measured wearing light clothes, as light as possible in order to have more reliable and accurate data, this procedure lasted ten minutes approximately for each person. After the anthropometric information was collected, these data were captured in Microsoft Excel and then migrated to the statistical software Minitab in order to obtain measures of central tendency and percentiles. We used (eq. 1) to calculate the percentiles of the anthropometric data (depicted in Table 1). These data helped us estimate the links' length for the master robot.

$$P = \mu \pm Z\sigma \tag{1}$$

Where:

- P: Percentile.
- μ: Sample mean.
- σ: Standard deviation of the sample.
- Z: Standard value from normal distribution.

3 Kinematic Model

The basic movements of the human arm can be simplified in three dual rotational articulations plus the end effector (depicted in figure 2):

1) clavicle - shoulder.
2) shoulder - elbow.
3) elbow - wrist (end effector).
4) end effector.

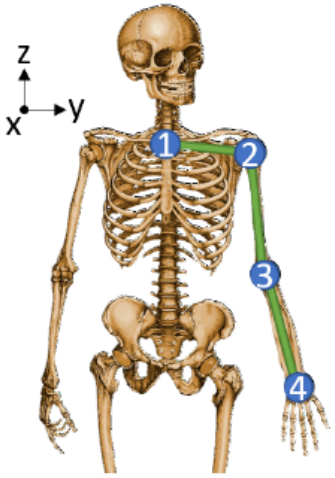

Figure 2. Key points in which the dual articulations and the end effector are located.

In this work a 6 DoF manipulator with rigid rotational joints is considered and it is depicted in (figure 3).

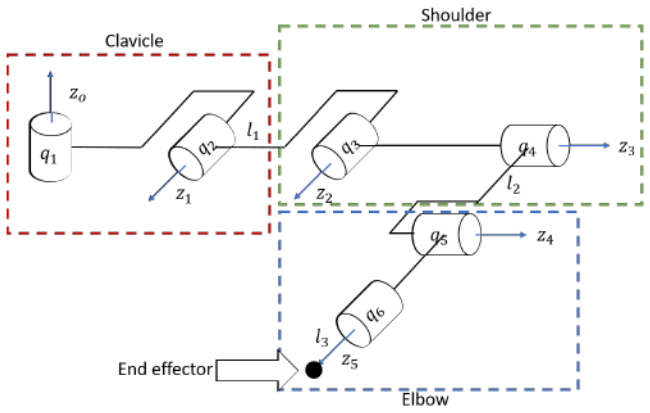

Figure 3. Manipulator scheme showing the z_i-angle, q_i-joint location and the end effector's position.

Where:

- l_1: link 1 (clavicle - shoulder).
- l_2: link 2 (shoulder - elbow).
- l_3: link 3 (elbow - wrist).
- q_i: articular joint.

Its kinematic model can be obtained using the Denavit-Hartenberg convention:

Knowing that the equation that describes the forward kinematics is given by (eq. 2)

Table 2
Denavit-Hartenberg parameters

Θ	d	a	α
q_1	5.5	0	$\pi/2$
q_2	l_1	0	0
q_3	5.5	0	0
q_4	0	9.5	0
q_5	0	l_2	0

$$\vec{x}(t) = f(q) \tag{2}$$

Where $\vec{x}$ represents the Euclidean position of the end effector and $q = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 & q_5 \end{bmatrix}^T$.

After applying Denavit-Hartenberg convention we obtain the Matrix A which is defined by:

$$A_i^0 = \begin{bmatrix} R_{3x3} & t_{3x1} \\ \vec{0}_{1x3} & 1 \end{bmatrix}_{4x4} \tag{3}$$

From (eq. 3) we can extract the end effector's pose. Looking at the block matrix t in A0 n we can determine the Euclidean position, and looking at the block matrix R in A0 n we can derive the rotation. For this to happen we need to transform a rotation matrix to a vectorial representation. The most natural would be Yaw, Pitch and Roll, depicted in (fig. 4).

Meaning that the pose is defined as:

$$x = \begin{bmatrix} x \\ y \\ z \\ \phi \\ \theta \\ \psi \end{bmatrix} \tag{4}$$

4 Control Scheme

As only the position was intended to be controlled, (eq. 4) changed to (eq. 5). The control scheme used for the robot's end effector to converge in a desired position (regardless of the orientation) was a simple PID Controller (eq. 6).

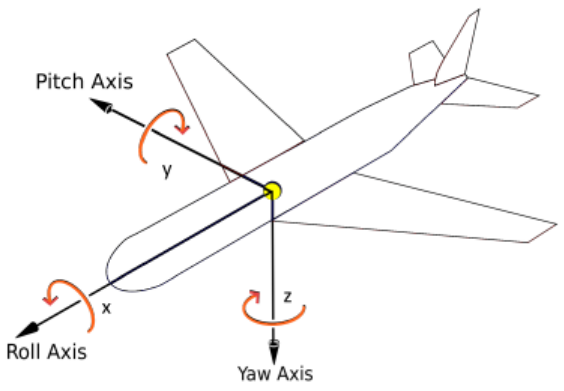

Figure 4. YPR representation where roll is rotation over the x-axis (ϕ radians), pitch is a rotation over the y-axis (θ radians) and yaw is a rotation over the z-axis (ψ radians).

$$\vec{x} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{5}$$

$$\vec{u}(t) = K_p\vec{e}(t) + K_i\vec{\dot{e}}(t) + K_d \int \vec{e}(t)dt \tag{6}$$

Where:

- $u(t) = \vec{\dot{q}}$.
- K_p: Proportional gain diagonal matrix.
- K_i: Integrative gain diagonal matrix.
- K_d: Derivative gain diagonal matrix.
- $e(t)$: error defined by $\vec{x}_d(t) - \vec{x}(t)$

It is important to notice that the robot will only converge if it exists in the manipulator's work zone (defined by eq. 7)

$$x \to x_r \forall x_r \exists \left\{\mathbb{R}^3 | x_r \leq \Psi\right\} \tag{7}$$

Where:

- $\Phi \leq \frac{1}{3}\pi \sum_{i=1}^{n} l_i$: Manipulator work zone.

In order to control the system we need the inverse kinematics given by (eq. 8).

$$\vec{q} = f(\vec{x}) \tag{8}$$

(eq. 8) can be re-written in a more appropriate form as (eq. 9)

$$\vec{q} = \int \vec{\dot{q}}dt = J_v^+ \vec{e}(t) \tag{9}$$

Where:

- $\vec{\dot{q}}$ is the articular velocities vector.
- J_v^+ is the sub-matrix of linear velocities of the geometric Jacobian's psuedo-inverse. Which is a matrix that maps the linear velocities to articular velocities.

The geometric Jacobian was calculated using (eq. 10)

$$J = \begin{bmatrix} z_1 \times (o_n - o_1) & z_2 \times (o_n - o_2) & \dots & z_n \times (o_n - o_n) \\ z_1 & z_2 & \dots & z_n \end{bmatrix} \tag{10}$$

Where z_i is the product between the rotation matrix i and the vector $\hat{k}$ $\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, and o_i is the translation of the frame i.

In this case, the end effector's orientation it's not needed, hence, the geometric Jacobian is restricted to use only linear velocities.

5 Results

The algorithm was developed in Matlab/Arduino trying to make the real-time bidirectional communication as real as it can be. We used the tic and toc Matlab's functions to establish an adaptive delay that match a fixed integration step for the control.

The adaptive delay was calculated using the algorithm's execution time and a fixed desired step for the integrator (depicted in figure 5).

The master system consisted in using an Arduino MEGA (figure 6) and three potentiometers as input for the desired position in real time.

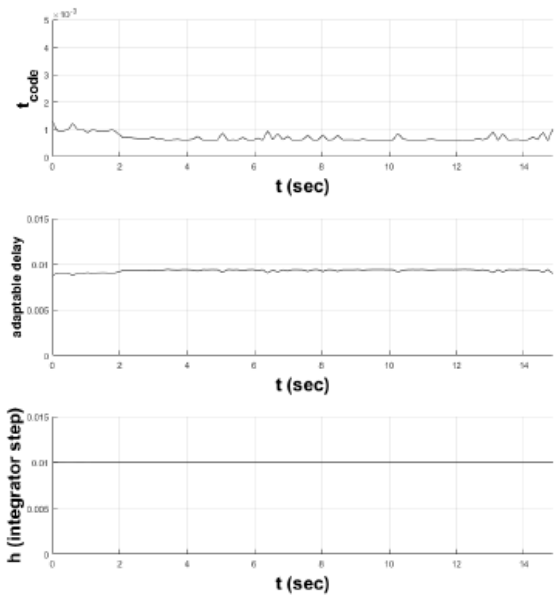

Figure 5. Algorithm's execution time over each iteration. In order to keep the integrator step fixed at 0.01s it was needed to wait for an adaptive delay $(h - t_{code})$ after each control iteration.

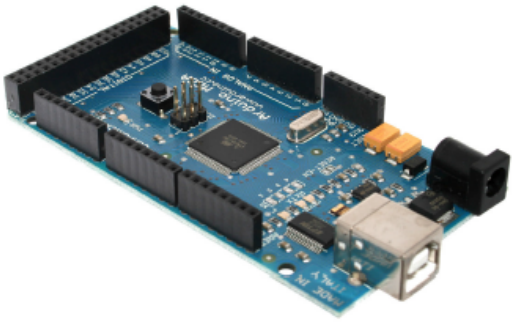

Figure 6. An Arduino MEGA was used to control the servo-motors due to its PWM output pins. The communication between the Arduino and the PC was established using Serial protocol.

$\vec{x}_d(t)$ was modified in each iteration moving the potentiometer' position (figure 7)

In (figure 8) can be seen the position error between $\vec{x}_d(t)$ and $\vec{x}(t)$. In the same way, the articular values stabilize at a certain value depicted in (figure 9).

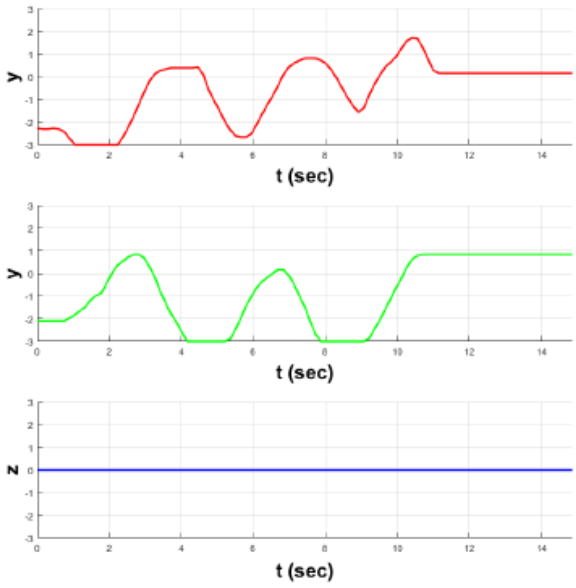

Figure 7. x_d (red), y_d (green) and z_d (blue) behavior over time as a result of modifying the potentiometer position.

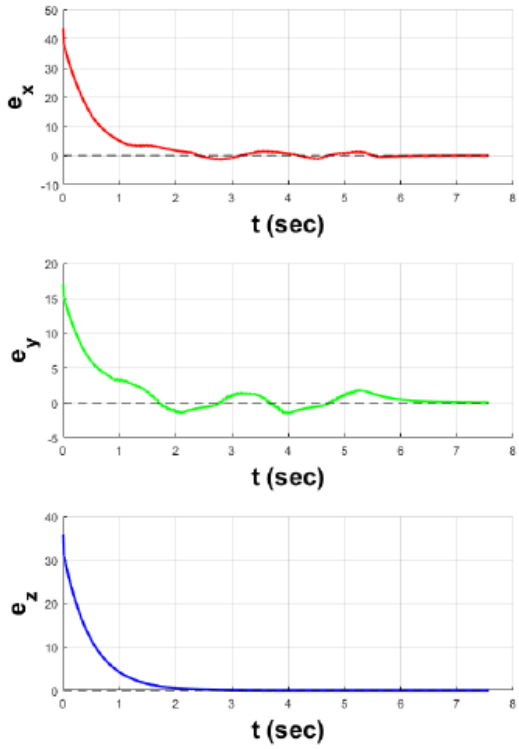

Figure 8. x-error (red), y-error (green) and z-error (blue) over time converging at 0.

6 Conclusion

It can be seen that the control scheme proposed works as expected controlling the end effector's euclidean position. Also, the bidirectional communication between Arduino and Matlab using a Serial protocol is fast enough to control

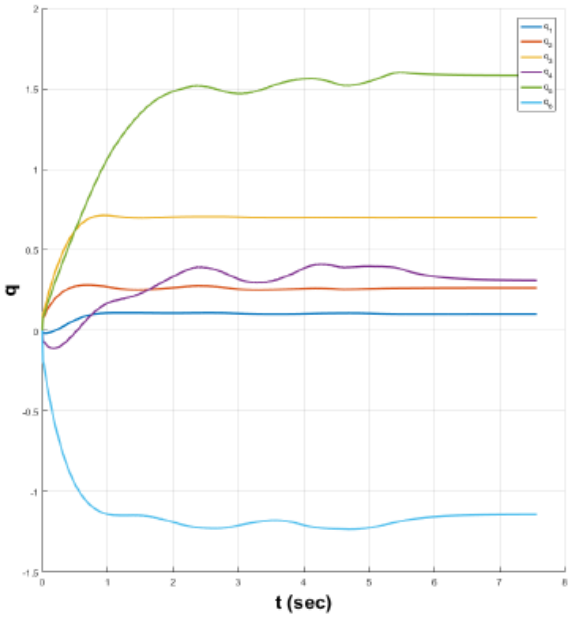

Figure 9. Manipulator's articular values over time being stabilized in a certain value where $\vec{x}_d = f(\vec{q}(t))$

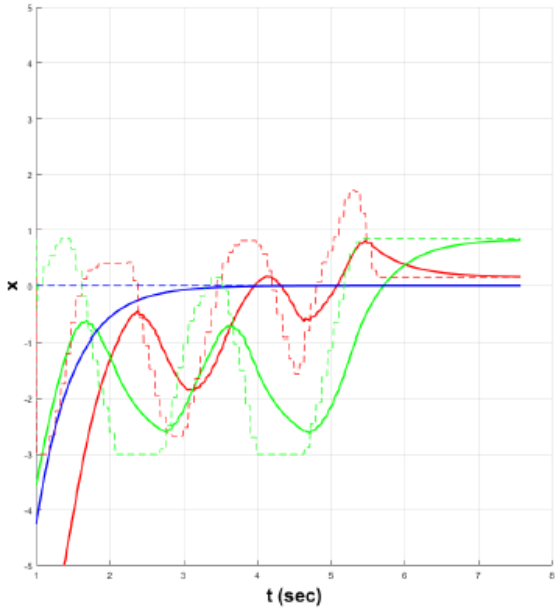

Figure 10. $\vec{x}(t)$ (solid line) evolution over time trying to converge to $\vec{x}_d(t)$ (dotted line)

in real-time te system. In the future, we might change the code from Matlab to C^{++} or ROS to speed up the communication and make h smaller.

7 Future Work

From this work we established that the next step will be the orientation control and then we will need to build the robot arm and apply the control scheme.

Once we prove the control scheme works, we then need to build the master system which will include a 6 DoF IMU (depicted in figure 11) to obtain the complete pose.

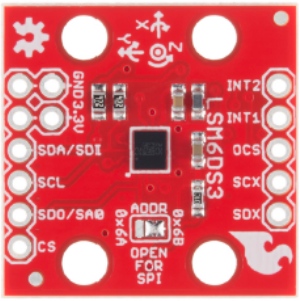

Figure 11. IMU proposed (Sparfun LSM6DS3)

References

[1] G. S. E, S. Emanuel, and M. Vicente, "Implementaciòn de sistema de teleoperaciòn multi-operador multi-robot," 2014.
[2] B. J. Montañez and S. M. Pinto, "Análisis de elementos en zona local y remota para la teleoperación del brazo robótico al5a," 2016.
[3] R. Daniel, M. Bilge, and G. Michael, "A motion retargeting method for effective mimicry-based teleoperation of robot arms," 2017.
[4] K. Fumio, K. Futoshi, and N. Hiroyuki, "Human motion caption with vision and inertial sensors for hand/arm robot teleoperation," 2016.
[5] C. Elmer, R. Hans, E. Eduardo, and H. Jerson, "Control de un brazo robótico antropomórfico con 6 grados de libertad teleoperado para la seguridad industrial en el perú," 2016.
[6] A. Takashi, Y. Motohiro, and M. Nobuto, "Development of a teleoperated robot arm system using rsnp: Precise tasks performed using a predictive display," 2016.
[7] C.-H. Lee, J. Choi, H. Lee, J. Kim, K. min Lee, and Y. bong Bang, "Exoskeletal master device for dual arm robot teaching," 2017.
[8] R. Daniel, M. Bilge, and G. Michael, "Teleoperation of robot arm with position measurement via angle-pixel characteristic and visual supporting function," 2016.
[9] P. Carlos and S. Javier, "Human machine interface hmi using kinect sensor to control a scara robot," 2013.
[10] O. Rowland, V. Peter, G. M, OhPaul, B. Aaron, E. Magnus, and S. Mike, "Humanoid robot teleoperation for tasks with power tools," 2013.
[11] N. Ilana, H. Michael, and O. Allison, "Uncontrolled manifold analysis of arm joint angle variability during robotic teleoperation and freehand movement of surgeons and novices," 2014.
[12] B. J. Montañez and S. M. Pinto, "Intuitive and adaptive robotic arm manipulation using the leap motion controller," 2014.
[13] H. L. V, H. R. M, and S. S. S, "Assistive teleoperation of robot arms via automatic time-optimal mode switching," 2016.
[14] G. A. L and B. E. E, "Using hidden markov models to track upper extremity arm motions for surface electromyographic based robot teleoperation," 2014.
[15] P. Stephen, *Bodyspace. Anthropometry, Ergonomics and the Design of Work*. Taylor and Francis, 2003.
[16] M. R and S. K, *Lehrbuch de Anthropologia*, 1957.