MENSAJE DEL PRESIDENTE

A nombre de la Asociación Mexicana de Robótica e Industria A.C., les damos la más cordial bienvenida a este XXIII Congreso Mexicano de Robótica 2021, segundo en modalidad virtual e idóneo para compartir el conocimiento generado entre estudiantes, investigadores y profesionales que participan en diferentes áreas en el campo de la robótica, tales como: control automático, humanoides y sistemas de visión, robots manipuladores, robótica móvil y teleoperación, robótica aérea y educacional, entre otras. Para ello, fueron sometidos a una revisión con un arbitraje para garantizar calidad y éxito en el congreso. Por consiguiente, los autores y los revisores son los pilares de este congreso.

En esta ocasión contamos con tres conferencias magistrales, veinte ponencias y una plática informativa de la IEEE, sección Monterrey, México. Durante el congreso se seleccionó al ganador del premio Rafael Kelly, el cual es otorgado al mejor trabajo presentado por un estudiante, siendo evaluado por el comité de expresidentes, así como a los actuales presidente y vicepresidente de la Asociación y al presidente del Comité Organizador. De la misma manera, y en su novena edición, se entregará el premio AMRob a Tesis como un reconocimiento al trabajo de investigación realizado en instituciones de educación superior nacionales e internacionales.

Agradecemos a nuestros conferencistas magistrales: Dr. Luis Enrique Sucar Succar del INAOE, Dr. José Alfonso Pámanes del TecNM/I. T. de La Laguna y al Dr. Luis Rodolfo García Castillo de la Universidad Estatal de Nuevo México, por compartir sus conocimientos y recientes contribuciones con toda nuestra comunidad académica. Reconocemos también el trabajo realizado por el presidente del programa, el Dr. Javier Moreno Valenzuela, y el comité de revisores. Sus evaluaciones y sugerencias realizadas en tiempo y forma enriquecieron los trabajos presentados.

En esta vigésima tercera edición del congreso, queremos agradecer al Tecnológico Nacional de México Campus Instituto Tecnológico de Tijuana por ser la sede virtual de este evento por segundo año consecutivo. En especial al Dr. Luis Néstor Coria de Los Ríos, presidente del comité organizador, así como a su equipo de trabajo por su disposición, compromiso y esfuerzo para organizar el evento por segundo año consecutivo. Al Ing. José Guillermo Cárdenas López, Director del TecNM-Instituto Tecnológico de Tijuana, gracias por su apoyo financiero y logístico. No podemos dejar de mencionar al comité honorífico formado por los expresidentes de la Asociación que realizaron el primer congreso en el año 1999, y que desde entonces han trabajo intensamente para que lleguemos hoy a la vigésima tercera edición. Pero sobre todo reitero mi agradecimiento a todos los autores quienes son la esencia de este congreso.

Por último, quiero extender un agradecimiento a todos los asistentes a las conferencias magistrales y sesiones regulares, ya que uno de los principales objetivos de la Asociación Mexicana de Robótica e Industria, A. C. es promover sin fines de lucro la robótica en nuestro país.

¡Espero hayan disfrutado del congreso y les deseo el mejor de los éxitos!

Jesús Sandoval

Presidente de la Asociación Mexicana de Robótica (2020-2023)

XXIII CONGRESO MEXICANO DE ROBÓTICA
COMROB
2021

“Robótica en México: Retos y desafíos ante la nueva realidad”

Memorias del

XXIII Congreso Mexicano de Robótica 2021

Asociación Mexicana de Robótica e Industria AC
TecNM-Instituto Tecnológico de Tijuana

comrob.tijuana.tecnm.mx

Tijuana, Baja California, México, 27 al 29 de Octubre de 2021

Dr. Jesús Alberto Sandoval Galarza
Presidente de la AMRob

Dr. Luis Néstor Coria de los Ríos
Presidente del Comité Organizador

ISBN: 978-607-99446-1-2

TECNOLÓGICO NACIONAL DE MÉXICO

ROBÓTICA EN MÉXICO:
RETOS Y DESAFÍOS ANTE LA NUEVA REALIDAD

Memorias COMRob 2021

ROBÓTICA EN MÉXICO:
RETOS Y DESAFÍOS ANTE LA NUEVA REALIDAD

Memorias COMRob 2021

Eduardo Javier Moreno Valenzuela

Iliana Castro Liera

Jesús Alberto Sandoval Galarza

Alejandro Enrique Dzul López

Paul Antonio Valle Trujillo

Luis Néstor Coria de los Ríos

(EDITORES)

Primera Edición: Diciembre 2021
D.R. © Eduardo Javier Moreno Valenzuela
D.R. © Iliana Castro Liera
D.R. © Jesús Alberto Sandoval Galarza
D.R. © Alejandro Enrique Dzul López
D.R. © Paul Antonio Valle Trujillo
D.R. © Luis Néstor Coria de los Ríos
(EDITORES)

D.R. © UNIVERSIDAD AUTÓNOMA DE SINALOA
Ángel Flores s/n, colonia Centro, Culiacán, 80000 (Sinaloa)
DIRECCIÓN EDITORIAL

D.R. © INSTITUTO TECNOLÓGICO DE LA PAZ
Blvd. Forjadores de B.S.C. #4720, 23080 La Paz, B.C.S.
SELLO EDITORIAL
Instituto Tecnológico de La Paz (978-607-99446)

ISBN:
978-607-99446-1-2
Hecho en México

Índice

Capítulo 1. Control

Capítulo 1.

Control

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Robust stabilization of an inverted cart-pendulum: A second order continuous sliding mode approach

Samy Kharuf-Gutierrez, Alejandra Ferreira de Loza, Luis T. Aguilar, and Luis N. Coria

***Abstract*—This manuscript uses two continuous sliding mode controllers to stabilize a cart-pendulum system. The nonlinear model of the cart-pendulum system is, at the first stage, transformed into its regular form. Then, the control input designed via continuous sliding mode strategies uses a sliding variable with relative degree two, which reduces the number of stages of the design. The implemented control techniques are the discontinuous integral action and a *continuous* nested algorithm. The proposed robust control strategies are insensitive against unknown matched perturbations with bounded derivatives. Since the proposed controllers require the knowledge of the successive time derivatives of the sliding surface, a Levant's Differentiator is introduced to estimate such derivatives. The performance of both controllers is contrasted using the mean square error of the sliding surface as comparison criteria.**

***Index Terms*—High-order sliding mode control, Nonlinear uncertain systems, Cart-pendulum system.**

I. INTRODUCTION

The study of the inverted pendulum has been one of the most widely discussed topics, considering the multiple challenges it presents. This system has nonlinear dynamics, is unstable and underactuated, so thus an excellent benchmark for testing different control algorithms. A good number of control schemes such as passivity-based method, intelligent controllers [1], and coupled sliding-mode [2] have been employed for the cart–pendulum system (CPS). In [3], a passivity-based method is proposed. In this work, it was claimed that the total energy of the system can be used to achieve equilibrium. However, the solutions of some particular partial differential equations are generally very difficult to find.

In the literature exist a large number of intelligent controllers for the CPS, for instance [4]. The main drawback of the previous works is that they are based on heuristic approaches. In [2], the authors propose a coupled sliding surface with relative degree one. Unlike [4], the closed-loop stability is ensured.

A common procedure for the design of controllers for underactuated systems, relies on a transformation of the dynamics into a regular form [5]. As a result, the system is split into a series of cascaded blocks. The control can be thus designed using, for instance, a backstepping procedure [6]. However, the backstepping control design can be cumbersome and lacks robustness in the presence of external disturbances. Instead, combining the backstepping approach with sliding mode control (SMC) simplifies the design while adding robustness against matched disturbances. The work in [7] proposes a backstepping procedure for designing a sliding surface, whereas a discontinuous first-order sliding mode control drives the trajectories toward the sliding surface. Such a simplification in the design, comes from the fact that the discontinuous control action produces the collapse of the system's trajectories. In this regard, [8] proposes a discontinuous second-order sliding-modes approach. Discontinuous approaches, however, suffer from chattering and it may be a drawback for certain applications. To overcome chattering, some authors have exploited higher-order sliding-mode approaches to simplify even more the design [9].

HOSM controllers can be exploited to simplify the control synthesis and provide robustness, such as continuous twisting [10], quasi-continuous and nested [11], among others. For instance, in [10] the authors propose diverse HOSM controllers using optimal sliding surfaces of diverse relative degrees. The optimal sliding surface design, however, uses a linear approximation. Recently in [12], the authors proposed a new HOSM approach: the discontinuous integral algorithm (DIA). The DIA controllers are robust against Lipschitz perturbations. Also, the DIA approach is an intermediate solution between discontinuous HOSM controllers and the continuous PID [13].

The goal of the manuscript is to stabilize the pendulum in the upright position, despite disturbances acting on the control channel. Toward this aim, first, the system is taken to the regular form, and then, two-second order sliding mode approaches are applied: the DIA and the nested algorithm [11]. To do a fair comparison between continuous controllers, the nested algorithm is designed to provide a continuous control signal, increasing arbitrarily the system's relative degree. The use of second-order sliding mode controllers simplifies the design reducing the number of steps involved in the control synthesis [14]. In addition, second-order siding mode approaches shed benefits in terms of accuracy and chattering attenuation. Numerical simulations are carried to contrast the controllers, outcomes using the mean square error (MSE).

The manuscript contribution is two-fold: i) The application

This work was partially supported by the Mexican Council of Science and Technology CVU 1030388 and Grant 285279, by Tecnológico Nacional de México Grant 13391.21-P, by Cátedras CONACYT under Grant CVU 166403, Project 1537 and by Instituto Politécnico Nacional under Grant SIP 2021-0356.

Samy Kharuf-Gutierrez and Luis N. Coria are with IT de Tijuana, Tecnológico Nacional de México, Tijuana, México; (`samy.gutierrez201@tectijuana.edu.mx`; `luis.coria@tectijuana.edu.mx`).

A. Ferreira de Loza is with Instituto Politécnico Nacional CITEDI, Tijuana 22435, Mexico, and also with Cátedras CONACYT, Mexico City 03940, México (`daferreirade@conacyt.mx`).

L. Aguilar is with Instituto Politécnico Nacional, Avenida Instituto Politécnico Nacional 1310, Col. Nueva Tijuana, Tijuana, B.C., 22435, México (`laguilarb@ipn.mx`).

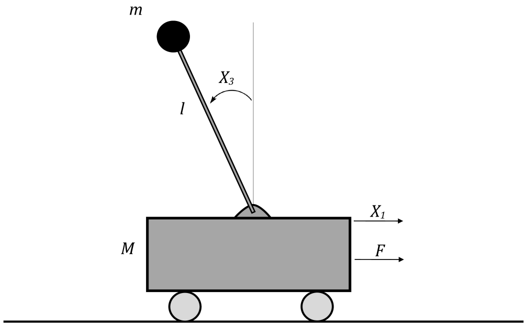


Fig. 1. Cart-pendulum diagram.

of two second-order sliding mode controllers to an inverted cart-pendulum, to ensure the asymptotical stabilization of the pendulum and cart position despite matched perturbations. $ii)$ A comparison analysis using the MSE is proposed to contrast the performance of both controllers.

The paper is organized as follows. Section II is concerned with modeling issues as well as the problem statement. Section III is dedicated to present a regular form of the system and then explain the design of the sliding surface to be used. Section IV presents the design of the robust controllers; unmatched uncertainties are not considered. In Section V we give the numerical simulations and in Section VI some conclusions are given.

Notation. For a real variable $a \in \mathbb{R}$ and a real number $b \in \mathbb{R}$ the symbol $\lceil a \rfloor^b = |a|^b sign(a)$, i.e. the signed power b of a; according to this notation $\lceil a \rfloor^0 = sign(a)$.

II. PROBLEM STATEMENT

Consider the cart-pendulum system (see Figure 1), described by the following state-space representation [8]:

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix} x_2 \\ \frac{(3mgl\cos x_3-4ml^2x_4^2)\sin x_3}{D} \\ x_4 \\ \frac{3((M+m)g-ml\cos x_3\; x_4^2)\sin x_3}{D}\end{bmatrix} + \begin{bmatrix}0\\ \frac{4l}{D}\\ 0\\ \frac{3\cos x_3}{D}\end{bmatrix}\Big(\tau - \big(f_s \operatorname{sign} x_2 + f_c x_2\big) + \omega_1(t)\Big) \tag{1}$$

where x_1 represents the position of the cart, x_2 is the cart velocity, x_3 is the pendulum angular position, and x_4 is the pendulum angular velocity, τ is the control input, $\omega_1(t)$ is a unknown bounded disturbances, and $D = l(4M + 4m - 3m\cos^2 x_3) > 0$. The cart friction is compound of two terms: the Coulomb and the viscous friction, $f_s \operatorname{sign} x_2$ and $f_c x_2$, respectively. Table I presents the CPS parameters given by the manufacturer INTECO.

The **control objective** of this paper is to stabilize the unstable equilibrium point of the cart-pendulum $x = [0, 0, 0, 0]$ by means of two continuous sliding mode controllers, operating under matched disturbances.

III. SYSTEM TRANSFORMATION

The CPS is an underactuated system with one actuator to control the system. The system (1) can be taken to its regular form with a coordinate transformation $\eta = \phi(x)$, which is a continuously differentiable function, so that the control input τ only appears in $\dot{x}_4$ [8]. Defining $\tilde{B}_1$ and $\tilde{B}_2$, which correspond to the components of the input distribution matrix from (1):

$$\tilde{B}_1 = \frac{4l}{D}, \quad \tilde{B}_2 = \frac{3\cos x_3}{D}.$$

Then, the transformation variable is proposed below:

$$\eta = \phi(x) = x_1 - \vartheta(x_3) \tag{2}$$

where $\vartheta(x_3)$ represents a solution to the equation:

$$\frac{d\vartheta}{dx_3} = \frac{\tilde{B}_1}{\tilde{B}_2} = \frac{4l}{3\cos x_3}. \tag{3}$$

The transformation (2) is valid $\forall \; x_3 \in (-\frac{\pi}{2}, \frac{\pi}{2})$. By differentiating (2) twice, one gets:

$$\ddot{\eta} = \dot{x}_2 - \frac{4}{3}l\frac{\dot{x}_4}{\cos x_3} - \frac{4}{3}l\,\frac{\sin x_3\; x_4^2}{\cos^2 x_3}, \tag{4}$$

and evaluating (1) into (4) it is possible to transform the system into the following regular form:

$$\begin{bmatrix}\dot{\eta}\\ \ddot{\eta}\\ \dot{x}_3\\ \dot{x}_4\end{bmatrix} = \begin{bmatrix}\dot{\eta}\\ -G(x_3,x_4)\tan x_3\\ x_4\\ \frac{3((M+m)g-ml\cos x_3 x_4^2)\sin x_3}{D}\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 0\\ \frac{3\cos x_3}{D}\end{bmatrix}\Big(\tau - \big(f_s \operatorname{sign} x_2 + f_c x_2\big) + \omega_1(t)\Big) \tag{5}$$

where:

$$G(x_3, x_4) = g + \frac{4}{3}l\frac{x_4^2}{\cos x_3}. \tag{6}$$

A. Sliding surface design for the cart-pendulum system

Any nonlinear control technique can be applied to stabilize (5). Due to its nonlinearity, however, the control design is achieved step by step from η to the controlled coordinate $\dot{x}_4$. The use of first-order sliding mode control diminishes the number of steps from four to three: the sliding surface is designed in three steps and, in the last step, a discontinuous controller takes care of the unknown dynamics [15]. Alternative, in the following section we exploit second-order sliding modes controllers. Thus, a second-order relative degree sliding surface is obtained in two steps and thus, a robust continuous controller stabilizes the remained dynamics on finite time [14].

In (5), the function $G(x_3, x_4)$ is positive for any value of its arguments if $-\frac{\pi}{2} < x_3 < \frac{\pi}{2}$. According to the mechanical characteristics of the CPS, the motor power is bounded. Therefore, the following assumptions are introduced:

Assumption 1: The perturbation term $\omega_1(t)$ is a bounded and Lipschitz continuous function, i.e. $|\omega_1(t)| < \gamma_{\omega_1}$ and $|\dot{\omega}_1(t)| < L_{dw1}$; where γ_{ω_1} and L_{dw1} are known constants.

Assumption 2: The angular speed of the pendulum is assumed to be bounded i.e. $|x_4| \leq \gamma_{x_4}$, where γ_{x_4} is a known constant.

To stabilize (5), and to ensure that the corresponding zero dynamics is stable, the $\tan x_3$ function is selected as a virtual control, i.e.,

$$\tan x_3 = \alpha_1 s_2, \qquad \alpha_1 > 0, \qquad s_2 = \eta + \dot{\eta}. \tag{7}$$

The intermediate control $\tan x_3$ in (7) is selected as a linear combination of η and $\dot{\eta}$, as presented in [15]. Then, the transformed dynamics of $\ddot{\eta}$ can be represented as follows:

$$\dot{\eta} = -\eta + s_2, \tag{8}$$

$$\dot{s}_2 = -\eta + s_2\big(1 - \alpha_1 G(x_3, x_4)\big). \tag{9}$$

Let us introduce the following Lyapunov candidate function:

$$V = \frac{1}{2}(\eta^2 + s_2^2), \tag{10}$$

whose derivative along the trajectories of the auxiliary system (8)-(9) is:

$$\dot{V} = \eta(s_2 - \eta) + s_2\big(-\eta + s_2 - \alpha_1 G(x_3, x_4) s_2\big),$$

$$\dot{V} = -\eta^2 - s_2^2\big(\alpha_1 G(x_3, x_4) - 1\big), \tag{11}$$

which is negative definite if and only if:

$$\alpha_1 > \frac{1}{G(x_3, x_4)} > 0. \tag{12}$$

Therefore, the system (8)-(9) is asymptotically stable, that is, $\eta \to 0$ and $s_2 \to 0$, when $t \to \infty$. In order to implement intermediate control (7) a fictitious output s_1, of the relative degree two, is to be chosen to ensure that the $(\eta, \dot{\eta})$ subsystem is minimum phase with respect to this output:

$$s_1 = \tan x_3 - \alpha_1(\eta + \dot{\eta}). \tag{13}$$

Thus, enforcing a sliding motion on $s_1 = 0$ solves the problem of the asymptotic stabilization of x_1 and x_3. On the other hand, if s_1 is differentiated we get one sliding variable of relative degree one. But this implies an increase in the complexity of the design. The problem is now to design a switching controller which guarantees the finite-time convergence of s_1 to zero.

IV. CONTINUOUS SLIDING MODE CONTROLLERS

The nominal control τ will be chosen to ensure that $s_1 \to 0$. The relative degree of s_1, with respect to the control input, is equal to two. This can be verified by differentiating s_1 twice i.e.,

$$\ddot{s}_1 = p(x_3, x_4) + \Delta(x_3, x_4, \omega_1) + \kappa(x_3, x_4)\tau, \tag{14}$$

and the term to be precompensated $p(x_3, x_4)$ is given by:

$$p(x_3, x_4) = 2\frac{\tan x_3}{\cos^2 x_3}x_4^2 + T_1(x) + T_2(x) \tag{15}$$

where:

$$T_1(x) = \Big(\frac{1}{\cos^2 x_3} + \frac{8\;l\;\alpha_1 x_4\;\tan x_3}{3\cos x_3}\Big) \times \left[\frac{3[(M+m)g - ml\cos x_3 x_4^2]\sin x_3 - 3\cos x_3 \psi(x_2)}{D}\right],$$

$$\psi(x_2) = f_s\,\mathrm{sign}\,x_2 + f_c x_2,$$

$$T_2(x) = \left[g + \frac{4}{3}\frac{lx_4^2}{\cos x_3}\right]\alpha_1 \tan x_3 + \left[g + \frac{4}{3}\frac{lx_4^2(1+\sin^2 x_3)}{\cos x_3}\right] \times \alpha_1 \frac{x_4}{\cos^2 x_3},$$

$$\kappa(x_3, x_4) = \left[\frac{3ml + 8ml^2\alpha_1 x_4 \sin x_3}{mlD\cos x_3}\right], \tag{16}$$

and the uncertain term:

$$\Delta(x_3, x_4, \omega_1) = \frac{3}{\cos x_3 D}\omega_1 + \frac{8l\alpha_1 x_4 \tan x_3}{D}\omega_1. \tag{17}$$

Following [12], we split the uncertain term into two parts:

$$\Delta_1(x_3, x_4, \omega_1) = \frac{8l\alpha_1 x_4 \tan x_3}{D}\omega_1, \tag{18}$$

$$\Delta_2(x_3, \omega_1) = \frac{3}{\cos x_3\; D}\omega_1. \tag{19}$$

Since we are targeting a regulation goal, the term $\Delta_1(x_3, x_4, \omega_1)$ vanishes to the origin when the state x_3 reaches the sliding surface and becomes zero, i.e. $\Delta_1(0, 0, \omega_1) = 0$ for all $t > 0$, and bounded by:

$$|\Delta_1(x_3, x_4, \omega_1)| \leq \frac{8\alpha_1\; c\; ||\;[\;x_3, x_4\;]^T||\;\gamma_{\omega_1}}{4M + 7m} \tag{20}$$

where the parameter c is a constant. Due to **Assumption 1**, $\Delta_2(x_3, x_4, \omega_1)$ is Lipschitz non-vanishing term, bounded by:

$$\left|\frac{d}{dt}\Delta_2(x_3, \omega_1)\right| \leq L, \tag{21}$$

$$L = \frac{(12lM + 21lm)L_{dw1} + (12lM + 21lm + 18lm\gamma_{\omega_1})\gamma_{x_4}}{(4lM + 7lm)^2}. \tag{22}$$

To design a controller that guarantees the finite-time convergence of s_1 to zero, the following nominal controller is proposed:

$$\tau = \frac{1}{\kappa(x_3, x_4)}(-p(x_3, x_4) - u), \tag{23}$$

and substituting (23) into (14) results in:

$$\dot{\eta} = \frac{\tan x_3 - s_1}{\alpha_1} - \eta, \tag{24}$$

$$\ddot{s}_1 = -u + \Delta_1(x_3, x_4, \omega_1) + \Delta_2(x_3, \omega_1). \tag{25}$$

According to the relative degree of the sliding surface, in the following subsections, we propose two different ways to design the robust control "$u(x)$".

A. Discontinuous integral action for arbitrary relative degree

To guarantee the finite-time convergence of s_1 to zero, by means of a continuous control signal, a sliding-mode control through a homogeneous discontinuous integral action (DIA) [12] is given by:

$$u = -k_2 \left\lceil \lceil \dot{s}_1 \rfloor^{\frac{1-2d}{1-d}} + k_1 \lceil s_1 \rfloor \right\rfloor^{\frac{1}{1-2d}} + \varrho, \tag{26}$$

$$\dot{\varrho} = -k_{I1} \left\lceil s_1 + k_{I2} \lceil \dot{s}_1 \rfloor^{\frac{1-2d}{1-d}} \right\rfloor^{\frac{1+d}{1-2d}}. \tag{27}$$

Then, with a homogeneity degree $d = -1$, defining $\Upsilon = \Delta_2(x_3, \omega_1) + \varrho$ and substituting (26)-(27) into (24)-(25) we get the closed-loop system:

$$\dot{\eta} = \frac{\tan x_3 - s_1}{\alpha_1} - \eta, \tag{28}$$

$$\begin{aligned} \ddot{s}_1 = & -k_2 \left\lceil \lceil \dot{s}_1 \rfloor^{\frac{1-2d}{1-d}} + k_1 \lceil s_1 \rfloor \right\rfloor^{\frac{1}{1-2d}} \\ & + \Delta_1(x_3, x_4, \omega_1) + \Upsilon, \end{aligned} \tag{29}$$

$$\dot{\Upsilon} = -k_{I1} \left\lceil s_1 + k_{I2} \lceil \dot{s}_1 \rfloor^{\frac{1-2d}{1-d}} \right\rfloor^{0} + \frac{d}{dt} \Delta_2(x_3, \omega_1). \tag{30}$$

Theorem 1 [12]: Let **Assumptions 1–2** be satisfied. Consider the uncertain term $\Delta_2(x_3, \omega_1)$ as in (21) and the closed-loop system (28)-(30). Under these conditions, for any values of $k_{I2} \in \mathbb{R}$, any $k_1 > 0$ it is possible to choose the gains k_2 sufficiently large and $k_{I1} > L$ such that s_1 is finite-time stable.

B. Nested Sliding Controller

The relative degree of the system is two, but we artificially increased the relative degree adding an integrator [11], i.e. defining a new state $\Upsilon = u + \Delta_2(x_3, \omega_1)$, with $\dot{u} = \nu$ and designing a third-order sliding mode controller for the new control variable ν. The following nominal controller is used:

$$\tau = \frac{1}{\kappa(x_3, x_4)} \big(u - p(x_3, x_4) \ \big), \qquad \dot{u} = \nu, \tag{31}$$

with the following 3-nested sliding controller:

$$\nu = -h \ sign\Big(\ddot{s}_1 + 2 \ \left(|\dot{s}_1|^3 + |s_1|^2 \right)^{\frac{1}{6}} sign\big(\dot{s}_1 + \lceil s_1 \rfloor^{\frac{2}{3}} \big) \Big). \tag{32}$$

Substituting (31) into (24)-(25), we get the following closed-loop system:

$$\dot{\eta} = \frac{\tan x_3 - s_1}{\alpha_1} - \eta, \tag{33}$$

$$\begin{aligned} \ddot{s}_1 &= \Delta_1(x_3, x_4, \omega_1) + \Upsilon, \\ \dot{\Upsilon} &= -h \ \mathrm{sign} \left(\ddot{s}_1 + 2 \ \left(|\dot{s}_1|^3 + |s_1|^2 \right)^{\frac{1}{6}} \mathrm{sign} \left(\dot{s}_1 \right.\right. \\ & \left.\left. + \lceil s_1 \rfloor^{\frac{2}{3}} \right) \right) + \frac{d}{dt} \big(\Delta_2(x_3, \omega_1) \big). \end{aligned} \tag{34}$$

Theorem 2 [11]: Let **Assumptions 1–2** be satisfied. Consider the uncertain term $\Delta_2(x_3, \omega_1)$ as in (21) and the closed-loop system (33)-(34). Under these conditions, for a value sufficiently large $h > L$, then the origin of the closed-loop system is asymptotically stable. In particular, the variable s_1 converge to zero in finite time.

C. Levant's Arbitrary Order Exact Differentiator

The controllers DIA (26)-(27) and nested (32) require the knowledge of the successive time derivatives of the sliding surface $(\dot{s}_1, \ddot{s}_1)$. Therefore, a finite time robust and exact differentiator of second-order is proposed [11]:

$$\dot{z}_1 = -k_1 \left\lceil z_1 - s_1 \right\rfloor^{\frac{2}{3}} + z_2, \tag{35}$$

$$\dot{z}_2 = -k_2 \left\lceil z_1 - s_1 \right\rfloor^{\frac{1}{3}} + z_3, \tag{36}$$

$$\dot{z}_3 = -k_3 \left\lceil z_1 - s_1 \right\rfloor^{0}. \tag{37}$$

In accordance with [16] the differentiator gains are chosen as $k_3 = 1.1Q$, $k_1 = 3.34(k_3)^{\frac{1}{3}}$ and $k_2 = 5.3(k_3)^{\frac{2}{3}}$, where $Q \geq |\ddot{s}_1|$. Consequently, the variable z_2 of the differentiator (35)-(37) is a finite time estimate of $\dot{s}_1$ and the variable z_3 is a finite time estimate of $\ddot{s}_1$.

V. SIMULATIONS

To illustrate the effectiveness of the proposed controllers (DIA and nested), numerical simulations were conducted on a CPS (1) with a fixed sampling step of 10 [ms], using the Euler integration method and initial conditions $x(0) = [0, 0.1, 0, 0]$. An external Lipschitz perturbation was applied $\omega_1 = 0.1 \sin(t)$. The controller is turned on at $t = 1$ $[sec]$, once the differentiator (35)-(37) has converged. Both controllers presented in this paper are designed assuming that all the states are available, with a differentiator gain of $Q = 14.5455$.

The mean square error (MSE) of the sliding variable (s_1) is taken as criteria for performance analysis. A simulation is designed, in which the gains of the two controllers were chosen such that the MSE of s_1 is similar for both controllers. Table I presents the controller gains. In Figure 3, the MSE of the system subjected to matched disturbances is compared. For both controllers (DIA and nested) the MSE of s_1 is equivalent. Figures 2 c) and 2 a) shows the evolution of the position variables (x_3 and x_1). Both, the angular and translational positions reached the origin; although exist slight oscillations in x_3 and a slight over-peak in x_1 when the nested control was used.

Both controllers are robust against Lipschitz perturbations and guarantee that the trajectories tend to the set $s_1 = 0$ in finite-time. The differences of the proposed approaches, are the needed information for their implementation and the effort to tune the parameters. For the nested algorithm only one gain needs to be tuned, but requires $\ddot{s}_1$ making it essential to use Levant's second-order differentiator. On the other hand, the DIA control has three more gains to tune, which implies a better precision, as shown in Figures 2 c) and 2 a). After establishing the MSE as a reference criterion, the performance of each controller is compared as to its numerical value of the control signal and its energetic implications.

The average power of the control signals ($\frac{1}{T} \int_0^T |u(t)|^2 dt$) at a total simulation time of $t = 40$ $[sec]$, can be seen in Figure 4. This Figure shows that the DIA controller has a lower energy requirement with respect to the nested controller.

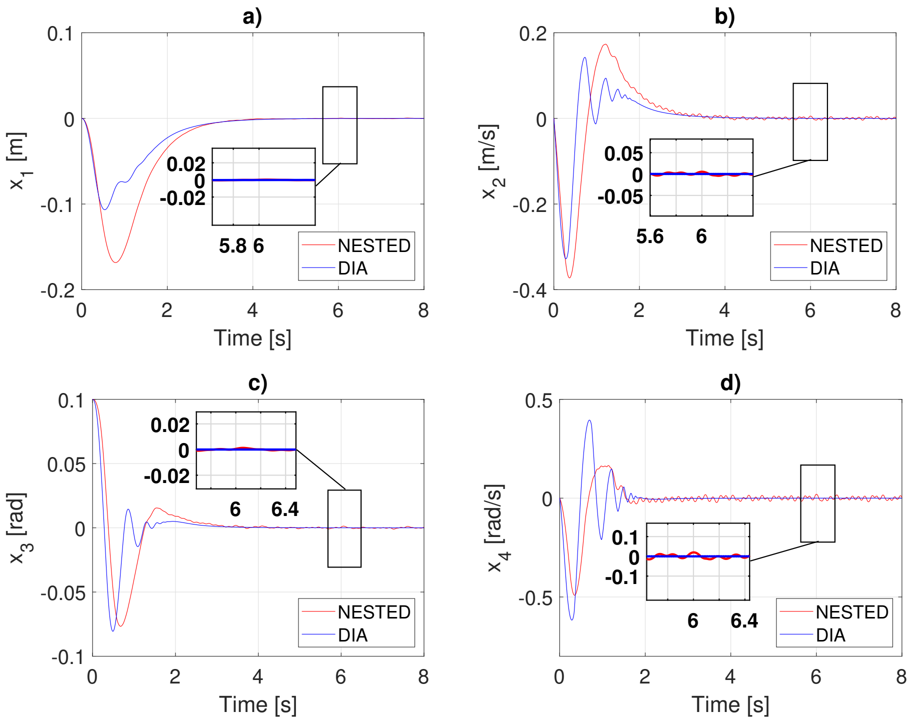


Fig. 2. Response of state variables for DIA and continuous Nested controllers: a) cart position, b) cart velocity, c) pendulum angle and d) pendulum angular velocity.

TABLE I
CPS PARAMETERS AND CONTINUOUS CONTROLLERS GAINS.

Parameters			**Controllers Gains**	
Name	***Description***	***Value***	***DIA***	***Nested***
M	Mass of cart	3.4 [Kg]	$k_1 = 2$	$h = 15$
m	Mass of pendulum	0.147 [Kg]	$k_2 = 1.5$	
l	Distance from the pendulum centre of mass to its pivot	0.1755 [m]	$k_{I1} = 2$	
g	Gravity acceleration	9.81 [m/s^2]	$k_{I2} = 3$	
f_s	Dry friction coefficient	1.1975 [N]	$\alpha_1 = 0.4$	$\alpha_1 = 0.4$
f_c	Dynamic cart friction coefficient	0.5 [N s/m]		

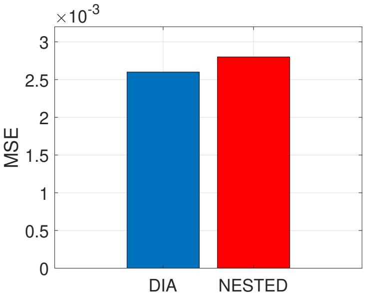


Fig. 3. MSE of the sliding surface s_1 in presence of matched perturbations.

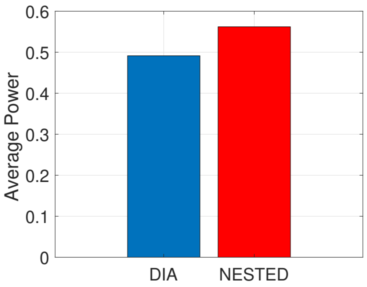


Fig. 4. The average power of the controllers.

Since the DIA controller requires lower power for the control signal, it is not necessary to make any significant compromises in terms of accuracy to reduce the control action.

Figure 6 (bottom) depicts the control input of the nested controller, where the presence of chattering is evident. One strategy to attenuate the chattering effect of the nested algorithm is to diminish the gain parameter h in (32).

VI. CONCLUSIONS

The stabilization problem of the CPS is tackled based on a scheme that uses continuous sliding mode controllers. Two controllers were considered: the classical nested with the

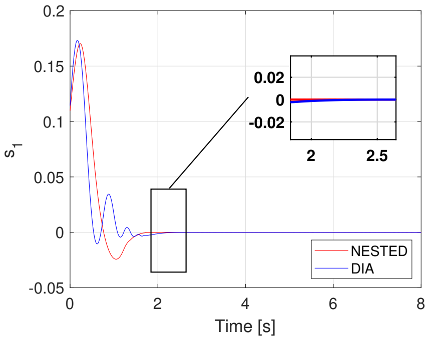


Fig. 5. Convergence to the sliding surface s_1.

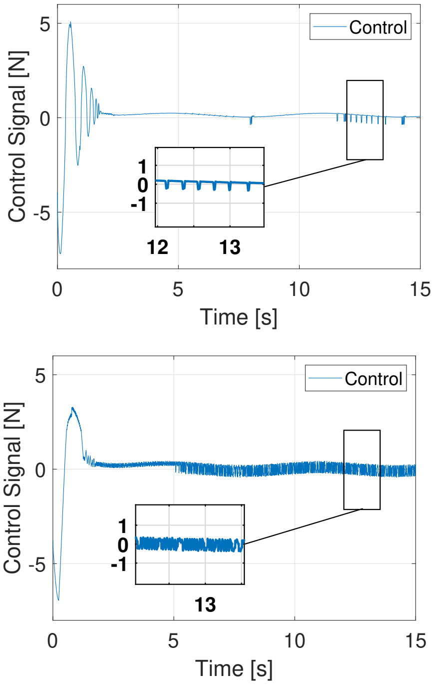


Fig. 6. Control Signal: (top) DIA controller, (bottom) Nested controller.

relative degree increased and the recently proposed DIA. Both controllers were evaluated after its gains were chosen such that the MSE of s_1 was similar. DIA and nested controllers ensures the asymptotical stabilization of the pendulum and cart position despite matched perturbations. However, simulation results show that the DIA controller has better angular precision and transient response with respect to the nested. Regarding the average power, the DIA controller has a lower energy requirement; also shows a smaller level of oscillations for the control input. In summary, the quantitative analysis verified that the DIA controller has the ability to diminish the power consumption and provides better control performance; ensuring cart and pendulum position accuracy.

REFERENCES

[1] A. I. Roose, S. Yahya, and H. Al-Rizzo, "Fuzzy-logic control of an inverted pendulum on a cart," *Computers & Electrical Engineering*, vol. 61, pp. 31–47, 2017.

[2] L. Ovalle, H. Ríos, and M. Llama, "A Robust Output-Feedback Control for the Cart-Pole System: A Super-Twisting Sliding-Mode approach," in *2018 XX Congreso Mexicano de Robótica (COMRob)*, 2018, pp. 1–6.

[3] G. P. Teja, A. Dhabale, and T. Waghmare, "Nonlinear control of the reaction wheel pendulum using passivity-based control and backstepping control," in *2020 IEEE First International Conference on Smart Technologies for Power, Energy and Control (STPEC)*, 2020, pp. 1–6.

[4] R. V. Gandhi and D. M. Adhyaru, "Takagi-sugeno fuzzy regulator design for nonlinear and unstable systems using negative absolute eigenvalue approach," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 2, pp. 482–493, 2020.

[5] Q. Khan, R. Akmeliawati, A. I. Bhatti, and M. A. Khan, "Robust stabilization of underactuated nonlinear systems: A fast terminal sliding mode approach," *ISA Transactions*, vol. 66, pp. 241–248, 2017.

[6] H. Khalil, *Nonlinear Systems*. Prentice Hall, 1996.

[7] S. Ullah, Q. Khan, A. Mehmood, and A. Bhatti, "Robust backstepping sliding mode control design for a class of underactuated electro–mechanical nonlinear systems," *Journal of Electrical Engineering & Technology*, vol. 15, 05 2020.

[8] S. Riachy, Y. Orlov, T. Floquet, R. Santiesteban, and J.-P. Richard, "Second-order sliding mode control of underactuated mechanical systems I: Local stabilization with application to an inverted pendulum," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 18, no. 4-5, pp. 529–543, 2008.

[9] D. Gutiérrez-Oribio, J. A. Mercado-Uribe, J. A. Moreno, and L. Fridman, "Robust global stabilization of a class of underactuated mechanical systems of two degrees of freedom," *International Journal of Robust and Nonlinear Control*, vol. 31, no. 9, pp. 3908–3928, 2021.

[10] J. Mendoza-Avila, I. Castillo, and R. Iriarte, "Higher Order Sliding Mode stabilization of an inverted cart-pendulum system," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 7157–7162, 2017, 20th IFAC World Congress.

[11] A. Levant, "Higher-order sliding modes, differentiation and output-feedback control," *International Journal of Control*, vol. 76, no. 9-10, pp. 924–941, 2003.

[12] J. Ángel Mercado-Uribe and J. A. Moreno, "Discontinuous integral action for arbitrary relative degree in sliding-mode control," *Automatica*, vol. 118, p. 109018, 2020.

[13] D. Gutiérrez-Oribio, Ángel Mercado-Uribe, J. A. Moreno, and L. Fridman, "Joint swing-up and stabilization of the reaction wheel pendulum using discontinuous integral algorithm," *Nonlinear Analysis: Hybrid Systems*, vol. 41, p. 101042, 2021.

[14] D. Gutiérrez-Oribio, L. Fridman, and J. Moreno, "Control de un Carro-Péndulo con algoritmos por Modos Deslizantes Continuos," in *Congreso Nacional de Control Automático 2017*, 2017.

[15] V. Utkin, J. Guldner, and J. Shi, *Sliding Modes in Electromechanical Systems*. London: Taylor and Francis, 1999.

[16] E. Cruz-Zavala and J. A. Moreno, "Levant's Arbitrary-Order Exact Differentiator: A Lyapunov Approach," *IEEE Transactions on Automatic Control*, vol. 64, no. 7, pp. 3034–3039, 2019.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

PID tuning using MOGA for joint position regulation of a pendular device

David Alonso Pimienta Berrelleza
Tecnológico Nacional de México
Instituto Tecnológico de La Paz
La Paz, B.C.S., México
M19310016@lapaz.tecnm.mx

Jesús Sandoval
Tecnológico Nacional de México
Instituto Tecnológico de La Paz
La Paz, B.C.S., México
jesus.sg@lapaz.tecnm.mx

Marco Antonio Castro Liera
Tecnológico Nacional de México
Instituto Tecnológico de La Paz
La Paz, B.C.S., México
marco.cl@lapaz.tecnm.mx

Iliana Castro Liera
Tecnológico Nacional de México
Instituto Tecnológico de La Paz
La Paz, B.C.S., México
iliana.cl@lapaz.tecnm.mx

***Abstract*—This paper presents an application of the Multi-Objective Genetic Algorithm (MOGA) for PID tuning capable of achieving good performance in the joint position regulation of a pendulum coupled to a servo-motor, and without exceed the motor nominal voltage. The performance is calculated using optimization indices as integral square error and integral time square error. Since the motor nominal voltage plays a key role in the performance of the controller, as well as the overshoot response of the joint position, the novelty of our proposal is that we have added the voltage magnitude and the overshoot response as additional performance indices. Experimental results on a commercial pendular device are presented to validate the proposed scheme, and a comparison is carried out with two other PID tuning methods to test its superiority.**

***Index Terms*—MOGA, joint position regulation, PID tuning rule, performances indices.**

I. INTRODUCTION

Mostly industrial robots manipulators are controlled through PID control, and its success lies in its simple linear structure that does not depend on the parameters of the robot [1], [2]. However, a challenging problem is to get an adequate selecting of the gains (process known as *tuning*) of PID control to achieve a good performance. For linear systems, there are several PID tuning rules on the literature help us solve this task, being the most popular methods for linear systems: Ziegler-Nichols, Cohen-Coon, Lopez et al., Sung et al., Chien-Hrones-Reswick, among others [3]. For non-linear systems, few PID tuning methods exist, and it is currently an active research area [1].

The aforementioned PID tuning methods are based on user experience and, in some cases, their application is not feasible due to time constrains or the existence of several functional restrictions (they need to accomplish different performance indices). Nevertheless, other alternatives as heuristic methods can be applied [4]. For example, it is known that given a vector of PID gains, it is not possible to modify their elements to improve a performance index without affecting another. This vector is a member of a set called *Pareto optimal*, and all the vectors in this set are called non-dominated solutions [5].

On the other hand, a Genetic Algorithm (GA) is a meta-heuristic search method inspired on the natural evolution of population and the Darwin and Mendel theories [6]. The successful application of GA to solve optimization problems has been documented in numerous articles (see e.g. [7] and references therein). Several GA versions have been applied to solve multiple-objective optimization problems, and a particular case is MOGA. As was mentioned in [5], some GA versions are able to find members of the Pareto-optimal solution set.

In the present research, MOGA is proposed as a PID tuning method, where the tuning can be seen as a multiple objective optimization problem. The Goldberg method is used to assign a fitness value to the individuals, considering two well-established performance indices: integral square error criterion (*ISE*), and integral time square error criterion (*ITSE*). In addition, two performance indices have been added for practical reasons: integral overshoot response, and integral voltage magnitude. A comparison between our proposed scheme and two PID tuning methods published in [1] and [8] is carried out to illustrate its superiority. It is worth mentioning that the tuning rule shown in [1] does not consider the voltage nor the overshoot response. Moreover, the position error integral, the voltage magnitude and torque were grouped into a single fitness function in [8], while in our proposed tuning method each criterion is evaluated independently to obtain a set of non-dominated solutions. It should be noted that the proposed scheme for PID control tuning using MOGA must meet stability conditions shown in [1]. It means that the main role of MOGA is to achieve the above performance criteria, and should be not confused as a tool for stability analysis.

The remaining of the paper is organized as follows. Section II introduces the plant model to be controlled, being a pendular device composed by a pendulum plus a DC motor with a gearbox. In Section III, we present our main result, which describes MOGA as PID tuning method for joint position regulation of the pendular device. The experimental evaluation is shown in Section IV along with a comparison with two other PID tuning methods. Finally, Section V gives the conclusions.

II. PENDULAR DEVICE

The pendular device composed by a pendulum plus a DC motor and a gearbox is shown in Fig. 1, and their parameters are listed in Tab. I. The pendulum consists of an arm plus a disk coupled to the motor through a gearbox, and the disk is attached to the end of the arm. The CD motor receives a voltage v as input and supplies a torque τ to produce an angular position q.

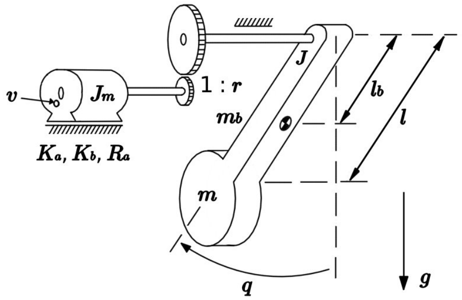


Figure 1. Pendular device with a DC motor

Table I
PHYSICAL PARAMETERS AND THEIR DESCRIPTION

Symbol	Description	Value	Unit
R_a	Armature resistance	2.6	Ω
K_a	Motor-torque constant	0.00768	Nm/A
K_b	Back emf constant	0.00768	Vs/rad
J_m	Rotor inertia of the rotor	3.9×10^{-7}	kg m^2
J	Arm inertia (without load)	0.00208	kg m^2
m_b	Arm mass (without load)	0.038	kg
m	Load mass (assumed concentrated at the center of mass)	0.04	kg
l_b	Distance from the axis of rotation to the center of mass of the arm (without load)	0.045	m
l	Distance from the axis of rotation to the load	0.1525	m
g	Gravity acceleration	9.81	m/s^2
r	Gear reduction ratio (in general, $r >> 1$)	70	-

The numerical value of the parameters shown in Tab. I are used for each tuning method to calculate the PID controller gains, which are described in Section II and Section III.

A. Dynamic model

The set of equations that describes the overall dynamic model of pendular device (pendulum plus DC motor-gearbox) can be written as [8]:

$$\frac{d}{dt}\begin{bmatrix} q \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ \frac{1}{J_m}\left[\frac{K_a}{rR_a}v - \frac{K_aK_b}{R_a}\dot{q}\right] \end{bmatrix}, \qquad (1)$$

where v is the input voltage, given by

$$v = \frac{rR_a}{K_a}\left[\left[M^{-1}J_m\right]\left[\tau - g(q)\right] + \frac{K_aK_b}{R_a}\dot{q}\right], \qquad (2)$$

with a strictly positive constant

$$M = J + ml^2, \qquad (3)$$

and the gravitational torque given by

$$g(q) = k_g \sin(q), \qquad (4)$$

being $k_g = [m_b l_b + ml]g$. We should mention that the armature inductance L_a and the viscous fiction coefficients (both, the arm f_L and the rotor f_m) have been suppressed for sake of simplicity, so, $L_a = 0$, $f_L = 0$ and $f_m = 0$.

B. PID control and closed-loop system

Since the mathematical structure of the classical PID control is expressed in terms of the joint position error $\tilde{q}$, it is convenient to introduce the error definition and its time derivative:

$$\tilde{q} = q_d - q, \qquad (5)$$
$$\dot{\tilde{q}} = -\dot{q}, \qquad (6)$$

because q_d is a constant for all $t \geq 0$, and consequently its time derivative $\dot{q}_d = 0$. Next, we present the PID control (Proportional-Integral-Derivative) as follows:

$$\tau = k_p\tilde{q} + k_v\dot{\tilde{q}} + k_i\xi, \qquad (7)$$
$$\dot{\xi} = \tilde{q}, \qquad (8)$$

where k_p, k_v and k_i are strictly positive constants named controller gains. Therefore, by substituting (7)-(8) into (1), with (2)-(4), we obtain the closed-loop system in terms of the state vector $[\xi \quad \tilde{q} \quad \dot{q}]^T$ given by

$$\frac{d}{dt}\begin{bmatrix} \xi \\ \tilde{q} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \tilde{q} \\ -\dot{q} \\ M^{-1}\left[k_p\tilde{q} - k_v\dot{q} + k_i\xi - g(q_d - \tilde{q})\right] \end{bmatrix}. \qquad (9)$$

We should mention that the model (9) has been used as an objective function to prove the tuning schemes. On the other hand, notice that the equilibrium of the closed-loop system is

$$\begin{bmatrix} \xi \\ \tilde{q} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \frac{1}{k_i}[k_g \sin(q_d)] \\ 0 \\ 0 \end{bmatrix}. \qquad (10)$$

In accordance with the stability analysis shown in [1], if we use the change of variable

$$z = \xi - \frac{1}{k_i}[k_g \sin(q_d)] \qquad (11)$$

then, the corresponding closed-loop equation may be expressed in terms of the state vector $[z \quad \tilde{q} \quad \dot{q}]^T$ as

$$\frac{d}{dt}\begin{bmatrix} z \\ \tilde{q} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \tilde{q} \\ -\dot{q} \\ M^{-1}\left[k_p\tilde{q} - k_v\dot{q} + k_i z + g(q_d) - g(q_d - \tilde{q})\right] \end{bmatrix} \qquad (12)$$

and the previous equation is autonomous and its unique equilibrium is the origin $[z \;\; \tilde{q} \;\; \dot{q}]^T = [0 \;\; 0 \;\; 0]^T$. Finally, by using the following global change of variables,

$$\begin{bmatrix} w \\ \tilde{q} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \alpha & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z \\ \tilde{q} \\ \dot{q} \end{bmatrix} \tag{13}$$

where $\alpha > 0$, the closed-loop system (12) can be rewritten as

$$\frac{d}{dt}\begin{bmatrix} w \\ \tilde{q} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \alpha\tilde{q} - \dot{q} \\ -\dot{q} \\ M^{-1}\left[\left[k_p - \frac{1}{\alpha}\right]\tilde{q} - k_v\dot{q} + \frac{1}{\alpha}k_i w + g(q_d)\right. \\ \left.-g(q_d - \tilde{q})\right] \end{bmatrix} \tag{14}$$

so, it can be proved that the origin of the state space, $[w \;\; \tilde{q} \;\; \dot{q}]^T = [0 \;\; 0 \;\; 0]^T$ is the unique equilibrium of the closed-loop system (14). If we can find conditions on the gains k_p, k_v and k_i such that the origin of the closed-loop system (14) is asymptotically stable, i.e. such that the origin is stable and at least for sufficiently small values of the initial states $z(0), \tilde{q}(0)$ and $\dot{q}(0)$, the state –particularly $\tilde{q}(t) = [q_d - q(t)]$– tend asymptotically to zero, then we are able to conclude that the position control objective [1]:

$$\lim_{t\to\infty}[q_d - q(t)] = 0 \tag{15}$$

is achieved (at least locally). In fact, it is important to remember that there is still no formal proof of global asymptotic stability of the PID controller. Moreover, due to the globality of the change of variable (13), the stability features of this equilibrium correspond to those for the equilibrium $[z \;\; \tilde{q} \;\; \dot{q}]^T = [0 \;\; 0 \;\; 0]^T$ of Equation (12).

C. PID tuning method based on Kelly & Santibáñez (2003)

As mentioned earlier, the stability analysis shown in [1] ensures asymptotic stability of the equilibrium under some conditions on the gains k_p, k_i and k_v. The following set of conditions establishes the PID tuning method:

$$k_i > 0, \tag{16}$$

$$k_p > k_s, \tag{17}$$

$$k_v > \frac{k_i M^2}{[k_p - k_s]M}, \tag{18}$$

where $k_s = m_b l_b g$.

However, the selected gains fulfilling the inequalities (16)-(18) ensure the control objective (15), but they do not ensure that the motor voltage does not exceed its nominal value. Moreover, the control gains are generally chosen after trial and error tuning for the best possible performance. In this case, to quickly reduce the joint position error $\tilde{q}$, we cannot increase the control gains too much because the voltage required by the motor grows to large values beyond its nominal value.

For convenience of notation, the PID tuning method based on Kelly & Santibáñez (2003) will be referred as “K&S.” Based on the PID tuning method of K&S, Tab. II shows two sets of gains arbitrarily chosen for two desired positions $q_d = 45$ [deg] and $q_d = 90$ [deg]. Both sets of gains k_p, k_i and k_v satisfy the inequalities (16)-(18), where $k_s = 0.01677$ and $M = 0.00301$, according to the values shown in Tab. I.

Table II
PID TUNING BASED ON KELLY & SANTIBÁÑEZ (2003)

Gains	$q_d = 45$ **[deg]**	$q_d = 90$ **[deg]**
k_p	0.75	1
k_v	0.09	0.10
k_i	0.21	0.69

III. PID TUNING METHOD BASED ON MOGA

The GA simplest form is based on three genetic operators: selection, crossover, and mutation. *Selection* consists on choosing those individuals, represented by their chromosomes, that are the fittest for their back recombination. *Crossover* operator selects pairs of parents randomly and combines their genetic material to obtain new individuals. Finally, the *Mutation* operator randomly changes the values of some genes on a small subset of randomly chosen individuals [7].

The genetic operators selected for our proposal were: the deterministic version of tournament selection, simple arithmetic crossover, and uniform mutation. A Multi-Objective version of this algorithm is able to find a set of non-dominated solutions, being suitable to solve multi objective optimization problems, where different and possibly conflicting optimization criteria are involved. This version of the algorithm is known as Multi-Objective Genetic Algorithm (MOGA). As mentioned by [9], MOGA is one alternative to handle multi-objective optimization problems by the means of heuristic search algorithms. To implement MOGA it is necessary to precisely define a set of performance indices and a way to assign a fitness value to each candidate solution within its n-th dominated subset, as will be discussed in the following subsections.

A. Performance

MOGA requires that a fitness function be created to assess the suitability of each individual to solve the optimization problem. To this end, it is necessary to establish the performance index to qualify each controller feature. The following performance indices were selected [8]:

1) Integral square error criterion (ISE) and integral time square error criterion (ITSE) are given by:

$$ISE = \int_0^\infty [q_d - q(t)]^2\, dt, \tag{19}$$

$$ITSE = \int_0^\infty t[q_d - q(t)]^2\, dt, \tag{20}$$

so that, the sum of (19) plus (20) yields

$$ITSE_+ = \int_0^\infty [t+1][q_d - q(t)]^2,\, dt \tag{21}$$

with which better results are achieved.

2) *Integral overshoot response (IOR):* this performance index increments its value for the case that the angular position $q(t)$ is greater than the desired position q_d.

The latter occurs when the joint position $q(t)$ has an overshoot over the desired constant position q_d [10]:

$$IOR = \int_0^\infty h(q(t))\, dt, \tag{22}$$

where the function $h(q(t))$ is conveniently defined as

$$h(q(t)) = \begin{cases} 0, & \text{if } |q(t)| \leq |q_d| \\ \alpha_o |q_d - q(t)|, & \text{otherwise} \end{cases} \tag{23}$$

where α_o is a weighted factor arbitrarily assigned by the user, and provides a way to evaluate the severity of the performance index. A properly selected value was $\alpha_o = 1$.

3) *Integral over-impulse voltage (IV):* this performance index increments its value when the voltage $v(t)$ exceeds the nominal voltage of the DC motor. The values for α_v and α_p are arbitrarily established by the user, where α_v is another weighted factor arbitrarily assigned by the user, and provides a way to evaluate the severity of this new index, and α_p is a proper factor to set an allowed small error between the calculated voltage magnitude $v(t)$ and the nominal voltage v_n:

$$IV = \int_0^\infty s(v(t))\, dt \tag{24}$$

where

$$s(v(t)) = \begin{cases} 0, & \text{if } \alpha_p v_n \geq v(t) \\ \alpha_v [v(t) - \alpha_p v_n]. & \text{otherwise} \end{cases} \tag{25}$$

The assigned values in the presented research were $\alpha_v = 1$ and $\alpha_p = 0.99$, which were selected as a weighted sum that considers all the performance indices, and their selection intended that the solutions found by MOGA do not reach the nominal voltage of the DC motor.

B. *Goldberg's fitness*

To assign ranks to the individuals, we use the method proposed by David Goldberg in [11]. This method ranks each individual from 1 to n depending on the number of dominant solutions found on the population. Therefore, the non-dominated solutions are ranked 1 and the remaining are ranked from 2 to n.

The fitness is determined following three steps on each iteration: first, the population is ordered accordingly to its rank; the second step is to assign a fitness values according to a monotonically descending function (usually linear); and finally as third step, each individual receives the average fitness of those with its same rank. The rank for each individual is assigned using the three performance indices $ITSE_+$, IOR and IV, defined in (21), (22) and (24), along with the definitions of inferiority, superiority and no-inferiority given by Fonseca et al. [11].

C. *PID tuning using MOGA*

Following the GA flow diagram shown in Fig. 2, an algorithm was developed to obtain the PID gains using the MOGA method, and the performance indices previously defined in (21), (22) and (24). The MOGA parameters are shown in Tab. III. These values are obtained in a empirical way, where generations and population size have an upper limit, because if their values increase, the execution time and the amount of memory increase. So, we start with low values and then increase until we get satisfactory results. The probability of selection and mutation control lies in how the best solutions are selected (or how quickly the bad ones are eliminated) and their adjustment allows to maintain the diversity of possible solutions.

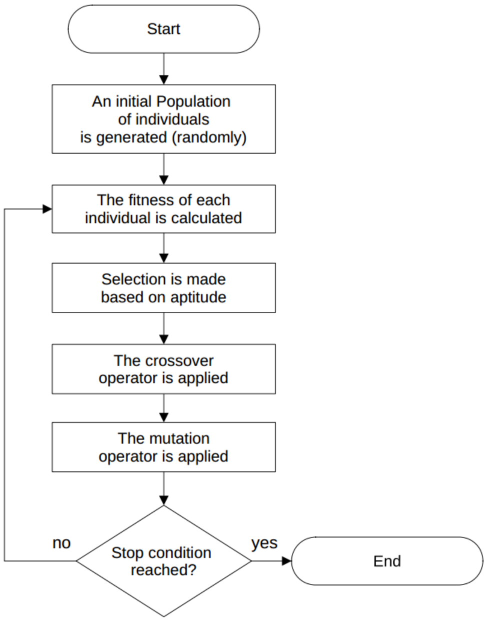


Figure 2. GA flow diagram

Table III
PID TUNING BASED ON MOGA

GA parameters	
Population size	100
Maximun generations	1000
Tournament size	2
Selection probability	0.5
Mutation probability	0.1

The execution time of the MOGA algorithm was 10 seconds to find the PID controller gains k_p, k_v and k_i, with an integration step of 0.0001 [s], and an exploration range from 0 to 0.75 for $q_d = 90$ [deg], and 0 to 1 for $q_d = 45$ [deg]. The exploration ranges were chosen in accordance to the gain k_p shown in Tab. II, being the largest value of the three gains. Two solutions from the non-dominated sets where arbitrarily selected for each desired position q_d called MOGA(a) and MOGA(b), after four executions. The gains for the PID controller (7)-(8) and the performance indices are shown in Tab. IV.

Table IV
PID TUNING BASED ON MOGA METHOD

	MOGA(a)		**MOGA(b)**	
q_d	90 [deg]	45 [deg]	90 [deg]	45 [deg]
k_p	0.456888	0.837	0.360123	0.920081
k_v	0.0593311	0.082	0.0405941	0.0756701
k_i	0.3038	0.525	0.247532	0.765307
$ISTE$	59.2699	10.9139	63.1162	9.6757
IO	0	0	2.0824	0.8861
IV	0	0	0	0

In order to ensure that the solutions obtained through MOGA meet asymptotic stability conditions given in [1], it can be verified that the set of gains obtained through MOGA, satisfy the inequalities (16), (17) and (18). As an example, from Tab. IV the PID gains selected in the second column (90 [deg]) were:

$$k_i = 0.3038, \tag{26}$$

$$k_p = 0.456888, \tag{27}$$

$$k_v = 0.0593311, \tag{28}$$

so, calculating the numerical values for $M = J + ml^2$ and $k_s = m_b l_b g$, introduced in section II-C, as well as the values presented in Tab. I, the following results are obtained:

$$k_s = 0.01677, \tag{29}$$

$$\frac{k_i M^2}{[k_p - k_s]M} = 0.00208, \tag{30}$$

such that $k_i > 0$, $k_p > k_s$ and $k_v > \frac{k_i M^2}{[k_p - k_s]M}$, fulfilling the inequalities (16), (17) and (18). By repeating the same procedure, it can be verified (16), (17) and (18) fulfill for all the sets of gains k_p, k_i and k_v listed in Tab. IV. Therefore, all solutions shown in Tab. IV satisfy the asymptotic stability conditions given in [1].

D. PID tuning method based on PSO and a weighted sum of indexes fitness

Another PID tuning heuristic method is the Particle Swarm Optimization method (PSO), which will be used as a comparison to the previous MOGA method. The confidence factors $c_1 = c_2 = 1.62$ and inertial weight $w = 0.8$ were the same values reported in [8]. By considering that a weighted sum of the optimization objectives can be used as a consolidated optimization function [5], the sum of the performance indices (21), (22) and (24) has been used to create the following fitness function:

$$A(PID) = ISTE_{+} + IOR + IV \tag{31}$$

where we have considered $\alpha_o = 100$, $\alpha_v = 100$, $\alpha_p = 0.99$ (these same values were established in [8]), 100 particles, 1000 iterations and an execution time of 10 [s], with a integration step 0.0001 [s], and an exploration range from 0 to 0.75 for $q_d = 90$ [deg], and 0 to 1 for $q_d = 45$ [deg]. Four executions of PSO were performed and the particles with best fitness were selected. Results for each desired position q_d are presented in Tab. V.

Table V
PID TUNING BASED ON PSO METHOD

Desired position q_d	90 **[deg]**	45 **[deg]**
k_p	0.4592724	0.9565021
k_v	0.0575989	0.0845647
k_i	0.2974778	0.5243776
$A(PID)$	233.7778	40.2587

E. Runtime: MOGA vs. PSO

For comparison purposes between MOGA and PSO methods, Tab. VI shows the average execution time of each heuristic method. The computer equipment used to execute the aforementioned algorithms had the following features: Linux Mint 19.3 on an AMD Ryzen 5 3400G processor with 16 GB of ram (2 modules of 8 GB of ram in dual channel configuration).

Table VI
RUNNING TIME

Heuristic tuning method	**Desired position** q_d **[deg]**	**Time [s]**
MOGA	90	290.001
	45	298.865
PSO	90	524.019
	45	520.908

It is important to clarify that the execution time of the MOGA and PSO algorithms may vary depending on its length, structure and programmation logic (coefficients, number of individuals or particles, genetic operators, etc.) as well as the computer equipment.

IV. REAL-TIME EXPERIMENTAL EVALUATION

The effectiveness of the results shown in Tab. IV with respect to the results of the Tab. II and Tab. V was verified by real-time experimentation on a physical device (commercial pendulum). Matlab R2012 and Simulink software were used along with Quanser libraries for the Quanser SRV02 model. The execution time was 95 [s] using a fourth order Runge-Kutta solver and an integration step of 0.01 [s]. The DC motor nominal voltage is 6 [V].

The initial condition for all experiments was: $[q(0) \quad \dot{q}(0)]^T = [0 \quad 0]^T$, which corresponds to the configuration shown in Fig. 3. The angular position q_m of the motor was measured with an integrated encoder [12]. It is worth mentioning that from Tab. IV and Tab. V, the smaller performance criteria values were considered the best.

From the experimental results, it can verified that for the K&S tuning method both sets of gains achieve a good performance, but exceeding the nominal voltage of the motor, that is, a maximum voltage of $v = 9.319$ and $v = 6.217$ [V], indicated in Tab. VI and Tab. VII, respectively.

Fig. 4 and Fig. 5 show the experimental results using the gains from Tables II, IV and V. The controller performance

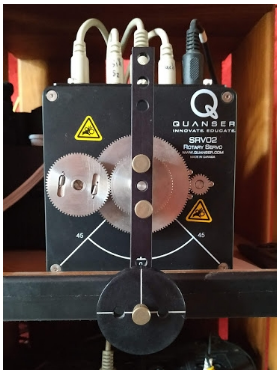


Figure 3. Commercial pendular device (SRV02 Quanser)

Table VII
EXPERIMENTAL EVALUATION WITH $q_d = 90$ [DEG]

PID controller performance criteria	**PID Tuning method**			
	K&S	**PSO**	**MOGA(a)**	**MOGA(b)**
Overshoot [deg]	0.549	0.066	**0.22**	3.164
Steady state error [deg]	0.132	0.066	**0.044**	0.308
Raising time [s]	**0.16**	0.21	0.21	0.18
Settling time [s]	0.52	0.33	0.34	**0.26**
Maximum voltage [V]	9.319	5.909	5.832	**5.459**
Maximum torque [Nm]	1.179	0.722	0.718	**0.566**

Table VIII
EXPERIMENTAL EVALUATION WITH $q_d = 45$ [DEG]

PID controller performance criteria	**PID Tuning method**			
	K&S	**PSO**	**MOGA(a)**	**MOGA(b)**
Overshoot [deg]	0.22	**0.154**	0.254	0.593
Steady state error [deg]	0.066	**0**	0.022	0.022
Raising time [s]	0.17	0.14	0.17	**0.12**
Settling time [s]	0.29	0.27	0.3	**0.19**
Maximum voltage [V]	6.217	5.945	**5.203**	5.721
Maximum torque [Nm]	0.786	0.752	**0.658**	0.724

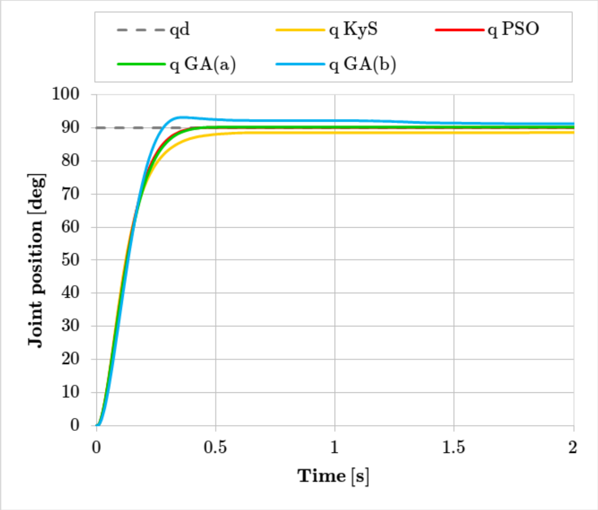


Figure 4. Time evolution of pendulum joint position q for a desired position $q_d = 90$ [deg]

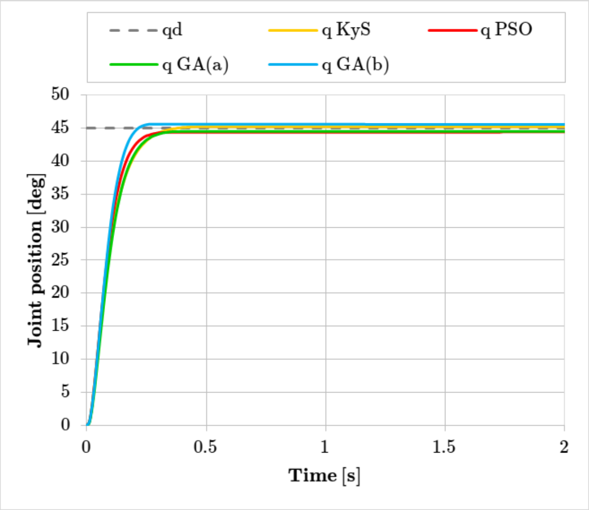


Figure 5. Time evolution of pendulum joint position q for a desired position $q_d = 45$ [deg]

criteria considered were: a settling time of 2%, a raising time from 10%, and a final value of 90% [10]. Both Figures depicted the time evolution of joint position $q(t)$. Notice that the three tuning methods have a similar response, but the gains selected with the MOGA method provide a better performance than the PSO method and the K&S method, where the nominal voltage is exceeded in the K&S approach.

Ultimately, an interesting observation is that using a PID tuning based on MOGA with the fitness calculation method presented on section III-B it was possible to find better sets of gains that produced lower maximum voltage and torque input on experimental evaluations than those obtained by the PSO method with the fitness calculation presented in section III-D.

V. CONCLUSIONS

A PID tuning based on MOGA method for the position control of a pendular device has been presented. A comparison with two other approaches showed that the proposed scheme provided better satisfactory results. Although the pendular device is a simple non-linear system, we believe that our proposal can work well in more complex mechatronic systems such as industrial robots. The latter because industrial robots can be seen as a chain of pendular devices. As future work it is intended to apply our proposal to a two degrees of freedom robot manipulator.

ACKNOWLEDGMENTS

This work was partially supported by CONACyT México grant 166636, and by project TecNM México.

REFERENCES

[1] R. Kelly, V. Santibáñez, and A. Loría, *Control of robot manipulators in joint space*. Springer Verlag, 2005.

[2] G. Lee, *Advances in Automation and Robotics, Vol. 2*, ser. Lecture Notes in Electrical Engineering, G. Lee, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, vol. 123. [Online]. Available: http://link.springer.com/10.1007/978-3-642-25646-2

[3] M. Azar, S. Martínez, E. Tarifa, S. Domínguez, and J. Gutiérrez, "Smart controller tuning using a fuzzy-PSO hybrid strategy (In Spanish)," *Workshop de Investigadores en Ciencias de la Computación*, pp. 830–836, 2016.
[4] R. Valerdi, "Heuristics for systems engineering cost estimation," *IEEE Systems Journal*, vol. 5, pp. 91–98, mar 2011. [Online]. Available: http://ieeexplore.ieee.org/document/5556044/
[5] C. Coello, "A comprehensive survey of evolutionary-based multiobjective optimization techniques," *Knowledge and Information Systems*, vol. 1, no. 3, pp. 269–308, aug 1999. [Online]. Available: http://link.springer.com/10.1007/BF03325101
[6] J. Morales, A. Rochín, M. Castro, and J. Sandoval, "AG and PSO as tuning methods of a PID for speed control of a DC motor (In Spanish)," *Nuevos avances en robótica y computación*, pp. 81–87, 2015. [Online]. Available: http://posgrado.itlp.edu.mx/CIRC2015/CIRC2015.pdf
[7] S. Haupt, "Introduction to genetic algorithms," *Artificial Intelligence Methods in the Environmental Sciences*, pp. 103–125, 2009.
[8] D. Pimienta, J. Sandoval, M. Castro, and I. Castro, "Tuning with PSO of a PID controller for the regulation of the position of a pendular motor joint (In Spanish)," *Congreso Internacional de Robótica y Computación*, pp. 1–10, 2021.
[9] C. Coello, "Constraint-handling using an evolutionary multiobjective optimization technique," *Civil Engineering and Environmental Systems*, vol. 17, pp. 319–346, 2000. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.36.3511
[10] T. Tay, I. Mareels, and J. Moore, *High Performance Control*, ser. Systems and Control: Foundations and Applications. Boston: Birkhauser, 1997.
[11] C. Fonseca and P. Fleming, "Genetic algorithms for multiobjective optimization : Formulation discussion and generalization," in *Proceedings of the 5th International Conference on Genetic Algorithms*, no. July, 1993, pp. 416–423.
[12] QUANSER, "Rotatory motion servo plant: SRV02 user manual," p. 31, 2008. [Online]. Available: http://www2.hawaii.edu/∼gurdal/EE351L/srv02.pdf

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Minimum energy nonlinear controllers based on analytic behaviors for velocity tracking on DC motors

Andrés G. Velásquez*, Eddie Clemente*, M. C. Rodríguez-Liñán‡ and Marlen Meza-Sánchez§
*TecNM - I. T. Ensenada, Email: al14760217@ite.edu.mx, eclemente@ite.edu.mx
‡CONACyT/TecNM - I. T. Ensenada, Email: mcrodriguez@conacyt.mx
§TecNM - I. T. Tijuana, Email: marlen.meza.sanchez@gmail.com

***Abstract*—This document describes the implementation of an evolutionary search for minimum energy controllers for second order DC motor model. Particularly, the recently developed methodology known as analytic-behaviors based control is utilized for the synthesis of minimum energy, nonlinear controllers. The found controllers are tested in simulation for the problem of velocity tracking.**

***Index Terms*—Analytic behaviors, Minimum energy control, Velocity tracking, DC motors.**

I. INTRODUCTION

When designing control systems, rarely, the only aim is to satisfy a given performance requirement, but to attain it in the best possible way. For example, the designer will search for a control signal that fulfills the desired closed-loop system performance with minimal energy [1]. This is referred to as optimal control. One of the advantages of optimal control methods is the ability to tailor the controller's response according to different objectives and/or conditions. Since systems are limited by physical constraints, in order to achieve said objectives, it is desirable to compute control signals that operate within the system's limits. Therefore, it is common to have a measure of the controller performance; this measure is typically called a cost function. The shape of such function depends on the desired behavior of the system [2]. In this sense, the term minimum energy control is used to refer to the admissible and optimal law that minimizes the quadratic performance index which defines the cost function [3].

Minimum energy systems are desirable in applications such as unmanned aerial vehicles (UAVs) and autonomous underwater vehicles (AUVs) [4], where the reliance on batteries determines the usability of the systems. Other examples include aerospace [5], transportation [6], and industrial applications [7], to name a few. Some relevant reasons for which energy minimization is desirable are, the effective use of batteries, as is the case for autonomous robots, the reduction of production costs in industrial applications, and the fulfillment of saturation limits related to the capacity of the actuators [8].

Popular control theory techniques for energy minimization include Model Predictive Control (MPC), Nonlinear MPC (NMPC), Linear Quadratic Regulators (LQR) and Linear Quadratic Gaussian (LQG) control. The cost function associated with each of these methods will vary slightly depending on the technique that is being used, however it is, in essence, a function that optimizes (minimizes) the energy required for the system to attain a given performance objective (or objectives, as is the case with MPC). Other classical techniques, such as PID and its variants, can be used in combination with an appropriate cost function [4].

Alternatively to traditional control, artificial intelligence methods can be used to solve the minimum energy optimization problem. Ant colony and Genetic Algorithms (GA) are used in [6] for the energy efficiency optimization in rail transport systems. Minimum energy in cyber-physical systems is considered in [9] through the application of a GA-based actuator control (GAAC). An energy management system for hybrid electrical vehicles based on reinforcement learning is presented by Lee et al. in [10]. Artificial Neural Networks are also employed in [11] for the problem of efficiently cooling a set of Computer Room Air Conditioning Units, aiming for minimal errors while optimizing the consumed energy.

This work proposes the development of a set of optimal control laws using the analytic behaviors framework [12]. These control laws address the velocity tracking problem in DC motors, modeled as second-order dynamical systems. A set of 11705 nonlinear functions are found to be used as partial controllers optimizing the total energy of the closed-loop system. This measure is denoted as a cost function of the error and the control signal. Numerical results demonstrate that the new nonlinear controllers exhibit better performance in velocity tracking with less energy consumption when compared to a classical PD controller.

The rest of this document is organized as follows. Section II introduces the DC motor model, and some preliminaries on the analytic behaviors framework, then Section III states the objective of this work. The proposed analytic behaviors-based solution is described in Section IV, while Section IV-B contains the obtained results. Finally, conclusions and future work are given in Section VI

II. PRELIMINARIES

This section defines the second order model of a DC motor in terms of velocity and acceleration; the analytic behaviors

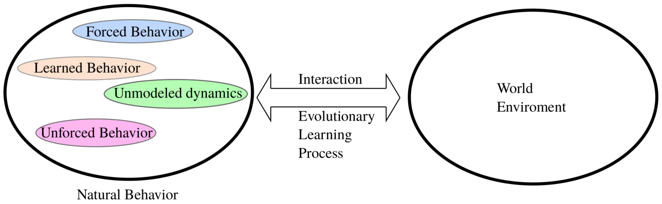


Fig. 1: Analytic Behaviors diagram.

framework is described as well.

A. Derivation of the model

The second order model of a DC motor is described as

$$J_m\ddot{\theta}(t) + b_v\dot{\theta}(t) = Ki(t) \tag{1}$$

$$L\frac{di}{dt} + Ri(t) = V(t) - K\dot{\theta}(t), \tag{2}$$

where J_m, b_v, L, R, K are parameters of the motor, being respectively, the rotor's moment of inertia, the motor's viscous friction constant, the electric inductance and resistance, and K representing both K_t, the motor torque constant, and K_e the electromotive force constant. Substituting i from equation (1) into (2), and taking $\omega(t) = \dot{\theta}(t)$, the model can be expressed as

$$J_mL\ddot{\omega}(t)+\dot{\omega}(t)(b_vL+J_mR)+\omega(t)(b_vR+K^2) = Ku(t), \tag{3}$$

where $u(t)$ is the applied voltage, $V(t)$.

B. Analytic Behaviors Framework

The analytic behaviors framework is an automated methodology devoted to the design of nonlinear controllers. This design scheme was proposed by [12], and extended in [13], [14], [15], [16]. The main idea is to emulate the natural learning process in humans. This approach is based on the acquisition of new behaviors or on the improvement of existing ones. In this case, the *natural* behavior of the system could be seen as a composition of three based behaviors. The first one is defined by the system model and their initial conditions; it is called an *unforced* behavior. The second behavior is related to the satisfaction of a given task, driven by a previously synthesized control action; it is referred to as a *forced* behavior. The third one is called *learned* behavior, and it is aimed to extend the capabilities of the system to accomplish new restrictions. The last behavior is implemented by a learning process using the Genetic Programming paradigm. Additionally, more behaviors could be considered, as part of the *natural* behavior, such as unmodeled dynamics or parametric uncertainties. Figure 1 illustrate the conceptual structure of the analytic behaviors framework.

III. PROBLEM STATEMENT

Consider the dynamical model of the DC motor given as in (3). In the following, the time dependency will be dropped for the sake of simplicity. Let the errors be defined as $\tilde{\omega} = \omega_d - \omega$, $\dot{\tilde{\omega}} = \dot{\omega}_d - \dot{\omega}$, and $\ddot{\tilde{\omega}} = \ddot{\omega}_d - \ddot{\omega}$, with ω_d being the reference angular speed, and $\dot{\omega}_d$ and $\ddot{\omega}_d$ its corresponding first and second derivatives. Then, using $a = J_mL$, $b = b_vL+J_mR$, $c = b_vR + K^2$ and $d = K$, the model can be rewritten as

$$\ddot{\tilde{\omega}} = \ddot{\omega}_d + \frac{b}{a}(\dot{\omega}_d - \dot{\tilde{\omega}}) + \frac{c}{a}(\omega_d - \tilde{\omega}) - \frac{d}{a}u_c, \tag{4}$$

where

$$u_c = \frac{a}{d}\Big[u_{inv} + u_{PD}\Big]. \tag{5}$$

Here, u_{PD} is the classical Proportional-Derivative control law and u_{inv} is an inverse dynamics feedback given as

$$u_{inv} = \ddot{\omega}_d + \frac{b}{a}\left(\dot{\omega}_d - \dot{\tilde{\omega}}\right) + \frac{c}{a}\left(\omega_d - \tilde{\omega}\right). \tag{6}$$

Then, the control objective is to apply the analytic behaviors framework to derive a set of optimal nonlinear controllers that minimize the total energy required in velocity tracking tasks for DC motors.

IV. ANALYTIC BEHAVIORS-BASED SOLUTION

In order to satisfy the defined control objective, the analytic behaviors framework is specified as follows.

- *Unforced* behavior. It is given by the dynamics of the second-order DC motor model, (3), with zero control input, and subject to some initial conditions.
- *Forced* behavior. It is described by the closed loop dynamics induced by a state-of-the-art control law. In this case, this is achieved by a traditional PD controller

  $$u_{PD} = -K_P\tilde{\omega} - K_D\dot{\tilde{\omega}}, \tag{7}$$

  where K_P, K_D are constant, positive gains. In order to provide a fair comparison with respect to the optimized controllers, the gains of the PD controller are tuned to satisfy a predefined settling time and maximum overshoot, in response to a unitary step reference signal.
- *Learned* behavior. This behavior is defined as the output response of the system to a partial nonlinear controller, denoted as u_{lb}, which is obtained by means of an evolutionary process.

Thus, the natural behavior of the system is the output response of the system to the control law (5), (6) where the optimized nonlinear term u_o is given as

$$u = u_c + u_{lb}. \tag{8}$$

The evolutionary process aims to build suitable optimized behaviors that minimize a cost function. This function is given as the time-integral of a function of the state error, and the control action. This expression has the form

$$J = \int_0^T L(x,u)dt, \tag{9}$$

in particular, the cost function is defined as [2]

$$J = \int_0^T \frac{1}{2}\left(\tilde{\omega}^2 + u^2\right)dt, \tag{10}$$

where it is assumed that only velocity tracking is sought.

The learning stage characterized by applying the Genetic Programming approach is described in the following subsection.

A. Implementation of the learning stage

The learning stage, based Genetic Programming, uses the syntactic tree representation for an expression, where the nodes are functions, and the leaves (called terminals) are independent variables, see Figure 2. The search space is defined by all the possible expressions that can be formed, in this case, rendering an optimized controller. Thus, the search space is determined by all the valid combinations of the selected functions and terminals. For this problem, the functions are described in Table I, and the terminals in Table II. Notice that the chosen functions are arithmetic trigonometric, hyperbolic, logarithmic and exponential expressions; the terminals are selected as the state variables, the error variables, and the system's constant parameters. In this case, these parameters are $J_m = 6 \times 10^{-7}\, kg \cdot cm^2$, $b_v = 0\, N \cdot m \cdot s$, $L = 8 \times 10^{-3}\, H$, $R = 10\,\Omega$, $K_t = 23.9 \times 10^{-3}\, N \cdot m/A$, and $K_e = 23.9 \times 10^{-3}\, mV/rpm$.

Initial conditions are defined as $\Omega = \{-1.5,\, 0,\, 1.5\}$ for the angular velocity, and $\dot{\Omega} = \{-1,\, 0,\, 1\}$ for the angular acceleration; then, the set of 9 initial conditions is given by the Cartesian product of $\Omega \times \dot{\Omega}$. To evaluate the efficiency of the found solutions, a tracking task for the velocity is defined next. The desired velocity is given as $\omega_d = \cos(4\pi/5 \cdot t)$; the sampling time is set to $T_s = 1 \times 10^{-4}\, s$; and the simulation is realized during $5\, s$. The PD control (7) that produces the *forced* behavior is tuned with gains $K_P = 5.33$ and $K_D = 20.34$, which have been chosen such that the close-loop system attains a unit step reference with a predefined settling time of $t_s = 1.5\, s$, and a maximum overshoot $M_p = 10\,\%$.

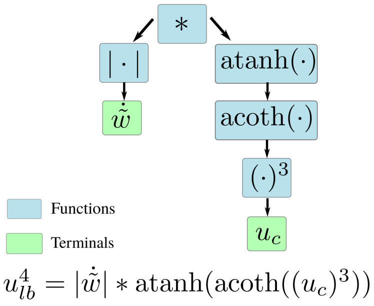


Fig. 2: Syntactic tree representation of an expression used by the Genetic Program. The expression is decoded using in-order tree traversal.

The evolutionary process used for the search of the optimized controllers is described next (see the flow diagram in Figure 3).

1) *Initialization*. A random pool of possible solutions, u_{lb}^i, is generated.
2) *Evaluation*. The candidate solutions are evaluated according to the cost function (10). Thus, the controller is defined as in (8). The evaluation is performed under a set of n initial conditions in order to build a general solution. Hence, the fitness function is defined as the average cost function over the n initial conditions, this is

$$F = \frac{1}{n}\sum_{j=1}^{n}\left[\frac{1}{2}\sum_{t=0}^{T}\left(\tilde{\omega}^2 + u^2\right)\right]. \qquad (11)$$

Notice that, in this case, the integral in (10) is written as a sum, since, in practice, the system is assumed to be discrete, with t being the time instants. The suitability of each solution is assessed by its scoring fitness value (11), where a lower fitness value implies a better solution.

TABLE I: Functions used in the evolutionary process.

Expression	Definition	Expression	Definition
$+$	Addition	$\cosh(\cdot)$	Hyperbolic cosine
$-$	Subtraction	$\tanh(\cdot)$	Hyperbolic tangent
$*$	Product	$\mathrm{sech}(\cdot)$	Hyperbolic secant
$/$	Division	$\mathrm{csch}(\cdot)$	Hyperbolic cosecant
$(\cdot)^2$	Square	$\mathrm{asinh}(\cdot)$	Inv. hyperbolic sine
$(\cdot)^3$	Cubic	$\mathrm{acosh}(\cdot)$	Inv. hyperbolic cosine
$(\cdot)^{(\cdot)}$	Exponentiation	$\mathrm{atanh}(\cdot)$	Inv. hyperbolic tangent
$\sqrt{\cdot}$	Square root	$\mathrm{asech}(\cdot)$	Inv. hyperbolic secant
$\sin(\cdot)$	Sine	$\mathrm{acsch}(\cdot)$	Inv. hyperbolic cosecant
$\cos(\cdot)$	Cosine	$\mathrm{acoth}(\cdot)$	Inv. hyperbolic cotangent
$\tan(\cdot)$	Tangent	$e^{(\cdot)}$	Exponential
$\sec(\cdot)$	Secant	$\ln(\cdot)$	Natural logarithm
$\csc(\cdot)$	Cosecant	$\mathrm{Re}(\cdot)$	Real part of argument
$\cot(\cdot)$	Cotangent	$\Vert \cdot \Vert$	Euclidean norm
$\mathrm{asin}(\cdot)$	Arcsine	$\vert \cdot \vert$	Absolute value
$\mathrm{acos}(\cdot)$	Arccosine	$\mathrm{sign}(\cdot)$	Sign
$\mathrm{atanr}(\cdot)$	Arctangent of the real part of argument	$\max(\cdot)$	Maximum
$\sinh(\cdot)$	Hyperbolic sine	$\min(\cdot)$	Minimum

TABLE II: Terminals used in the evolutionary process

Term	Description	Term	Description
ω	Velocity	u_{PD}	PD control signal
$\dot{\omega}$	Acceleration	u_c	$u_{PD} + u_{inv}$
$\tilde{\omega}$	Velocity error	a	System parameter
$\dot{\tilde{\omega}}$	Acceleration error	b	System parameter
$\vert\tilde{\omega}\vert$	Absolute value of velocity error	c	System parameter
$\vert\dot{\tilde{\omega}}\vert$	Absolute value of acceleration error	d	System parameter

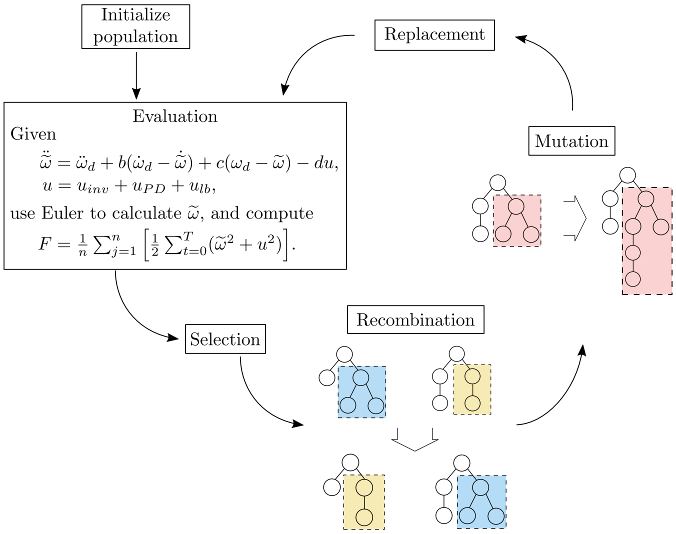


Fig. 3: Flow diagram of the *learned* behaviors search process.

3) *Selection*. The solutions are weighted according to their fitness value, (11); in consequence, the solutions with lower fitness have a higher probability of being selected. This process is performed by a roulette selection method.
4) *Recombination*. According to the recombination (crossover) and selection probabilities, two solutions are chosen from the pool. The recombination is given by interchanging two random branches of the syntactic tree. The offspring population is, thus, generated.
5) *Mutation*. Each element of the offspring pool has assigned a mutation probability. The mutation is realized by replacing a randomly selected branch of the syntactic tree by another randomly created branch, thus building a new population.
6) *Replacement*. Finally, the generation replacement is performed by substituting the previous population with the newly created one. The evolutionary cycle terminates once a set number of generations is completed, or a suitable value is found.

The following subsection discusses the best solutions obtained from the applied analytic behaviors framework, as well as the performance of a selected set of solutions.

B. The discovered optimized nonlinear controllers

The evolutionary process was executed eight times with independent runs; the chosen parameters are described in Table III. The evolution of the best solutions and their average are presented in Fig. 4. The evolutionary process built 11705 suitable *learned* behaviors with better fitness value than the u_c controller (without the optimized nonlinear term u_{lb}). From the set of found solutions, five learned behaviors u_{lb} are selected based on their fitness, their velocity tracking response, and their structural complexity. Notice that, according to equation (8), the found solutions u_{lb} complement the classical controller (5). Thus, the fitness shown in Table IV corresponds to the performance obtained by applying the integrated controller (8).

TABLE III: Parameters used by the evolutionary process.

Parameter	Value
Runs	8
Number of generations	100
Population size	400
Crossover rate	80%
Mutation rate	20%
Maximum tree depth	11
Sampling	Lexicographic
Elitism	Keep best

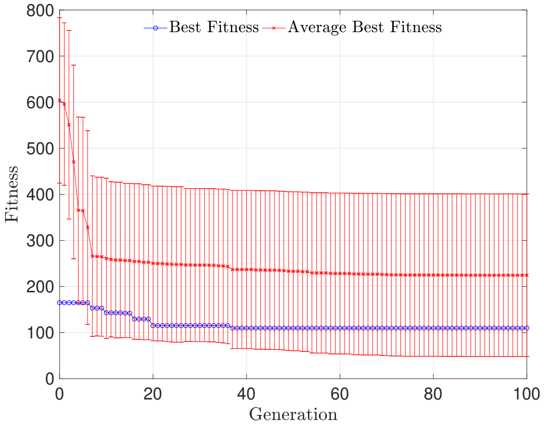
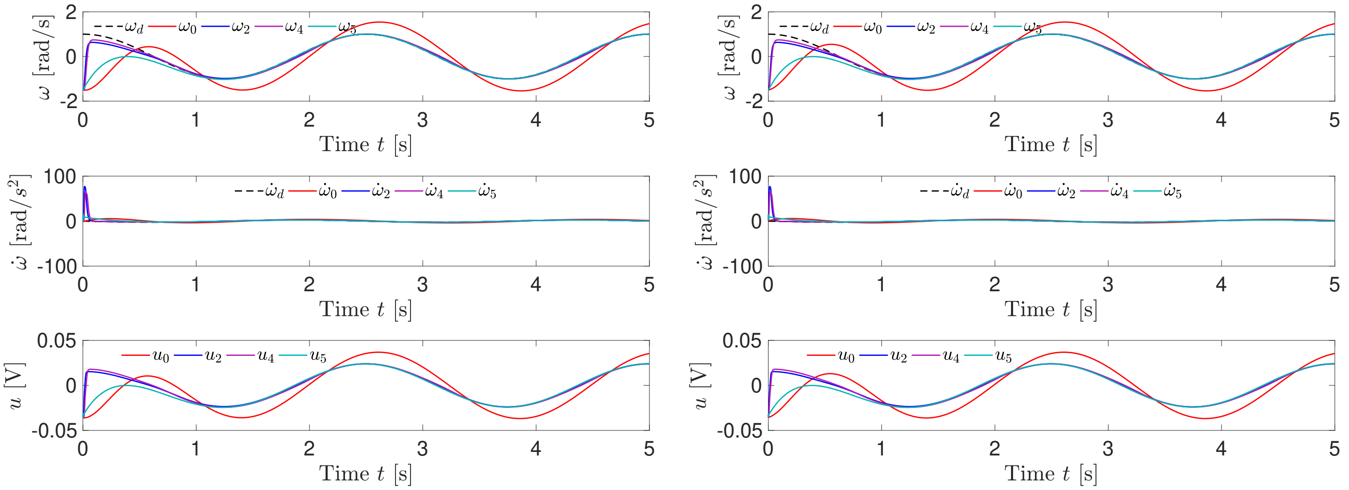


Fig. 4: The average best fitness, with their standard deviation, together with the best fitness for all runs of the evolutionary process.

The progression of the fitness, depicted in Figure 4, shows that from the first generation, the evolutionary process found solutions that improve the PD controller performance (fitness) by a factor of 3. This is, the computed fitness for the PD equals $F_0 = 690.5367$, while the fitness of the best solutions is below the 230 mark.

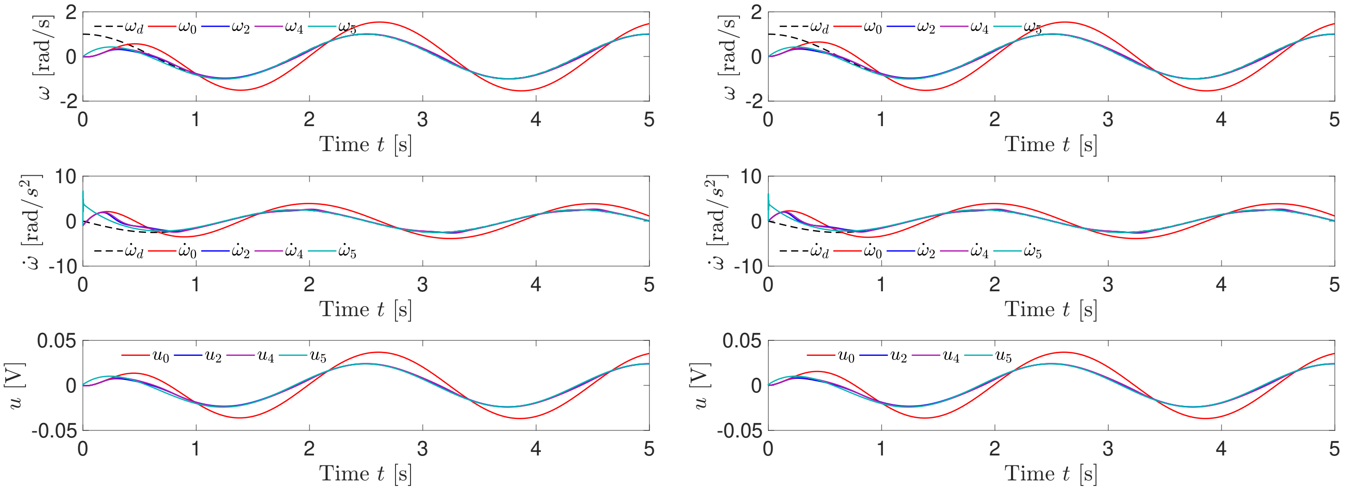
TABLE IV: Best solutions.

Sol.	Learned Behavior	Fitness
$u_i = u_c + u_{lb}^i$	u_{lb}^i	F_i
u_1	$u_{lb}^1 = \text{atanh}^4\left(\frac{6\times10^{-6}}{\text{acsch}^2\left(\frac{1.0078\times10^{-47}}{u_{PD}}\right)}\right)$	109.8
u_2	$u_{lb}^2 = exp(\text{acosh}(\text{atanh}(\text{acoth}((\min(\min(\text{atanr}^2(u_c), \text{acosh}((\lvert\dot{\tilde{w}}\rvert)^3)), \text{atanh}^3(u_c)))^3)))) \cdot \lvert\dot{\tilde{w}}\rvert$	115.2
u_3	$u_{lb}^3 = \lvert\dot{\tilde{w}}\rvert \cdot \text{acosh}(\text{atanh}(\text{acoth}((\min(u_c, \text{atanr}(\text{acoth}(u_c))))^3)))$	115.3
u_4	$u_{lb}^4 = \lvert\dot{\tilde{w}}\rvert \cdot \text{atanh}(\text{acoth}((u_c)^3))$	121.9
u_5	$u_{lb}^5 = b \cdot u_{PD}$	164.9
u_0	$u_{lb} = 0$	690.5



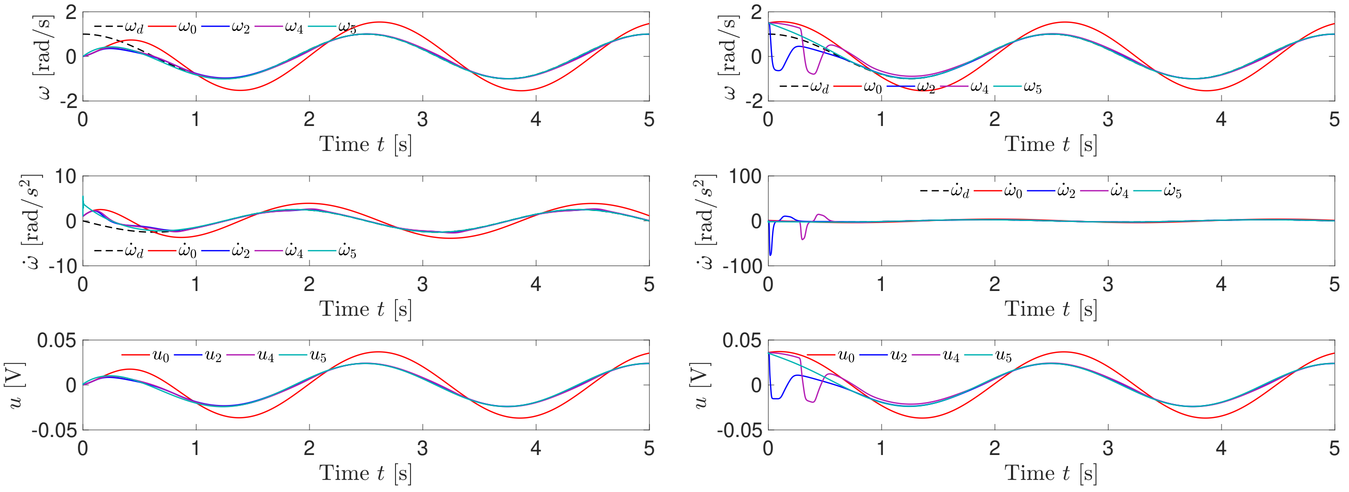
(a) $\omega(0) = -1.5$; $\dot{\omega}(0) = -1$

(b) $\omega(0) = -1.5$; $\dot{\omega}(0) = 1$



(c) $\omega(0) = 0$; $\dot{\omega}(0) = -1$

(d) $\omega(0) = 0$; $\dot{\omega}(0) = 0$



(e) $\omega(0) = 0$; $\dot{\omega}(0) = 1$

(f) $\omega(0) = 1.5$; $\dot{\omega}(0) = 1$

Fig. 5: Performance comparison of the evolved controllers u_2, u_4, and u_5 against the classical PD control law ($u_0 \triangleq u_c$). The plots show the results for six initial conditions; for each condition it is shown the tracking of velocity (top), the acceleration (middle) and the applied control signal (bottom).

V. NUMERICAL RESULTS

Simulation results are presented next to show the performance of the proposed methodology. Figure 5 shows the performance of three selected solutions: u_2, u_4 and u_5, where each u_i is the control law composed of the classical controller (5) plus the corresponding learned behavior u^i_{lb} (see equation (8)). For ease of presentation, and due to space limitations, only three significant solutions with six initial conditions, selected from Table IV, are presented. The first solution, u_2, is chosen as it showed the best overall performance; solution u_4 is selected for its low structural complexity; and solution u_5 is selected since it is a weighted version of the PD controller, u_{PD}, with a gain found by the evolutionary process.

From the plots, it can be seen that the found solutions follow the reference velocity more accurately than the PD; this holds true for all the plotted initial conditions. In terms of the acceleration response, it is observed that the evolved controllers have a similar performance in steady-state. Meanwhile, the control signal u_2 produces an acceleration overshoot during the transient for initial conditions $(\omega(0), \dot{\omega(0)}) = \{(-1.5, -1), (-1.5, 1), (1.5, 1)\}$ (see Figures 5a, 5b and 5f); and similarly with u_4 for initial conditions $(\omega(0), \dot{\omega(0)}) = \{(-1.5, -1), (-1.5, 1)\}$ (Figures 5a, 5b). Also notice that control signal u_4 shows the least smooth acceleration tracking. However, it is worth mentioning that the proposed optimizing criterion does not consider acceleration tracking, therefore, these errors are expected. Nevertheless, the two factors that are considered in the cost function: velocity tracking error, and demanded energy, are satisfied by all the found solutions. In this regard, it can be observed that for each condition, the controller u_2 demands the lesser energy when compared to the other three controllers. This can also be verified from the fitness values in Table IV.

VI. CONCLUSIONS AND FUTURE WORK

The analytic behaviors framework has been used in this paper to calculate nonlinear control laws that optimize the velocity error and the energy demanded by the controller, for the problem of velocity tracking in a DC motor. Both restrictions are optimized by a cost function that defines the fitness function of an evolutionary process. Each optimized nonlinear controller is given by the combination of a classic PD controller with dynamic feedback plus an evolved solution that optimizes the convergence of a system to a dynamic reference, while also minimizing the energy demanded by the controller. A set of 11705 solutions, with better performance than a classical Proportional-Derivative controller with dynamic feedback, have been found via the proposed methodology. The strength of the proposal is demonstrated since, from the first generation of the evolutive search, solutions with three times better fitness than the classical controller were found. From the set of suitable solutions, five are selected based on their fitness, their structural complexity, and their overall performance. Simulation results for three solutions are compared against a classical PD controller with dynamic feedback. These solutions possess better tracking properties, with less demanded energy than their classic counterpart. As future work, stability analysis can be addressed, and multi-objective optimization could be introduced to satisfy other performance requirements, such as bounded acceleration, settling time specifications, and/or saturation in the control action.

ACKNOWLEDGMENTS

Esteban González thanks for the support given by CONACYT and TecNM/I.T. Ensenada. M. C. Rodríguez-Liñán thanks CONACYT for its support through Cátedras project 923. Eddie Clemente thanks TecNM the support given through the project 11391.21-P.

REFERENCES

[1] E. D. Sontag, Mathematical Control Theory. Springer New York, 1998.
[2] P. L. F. Michael Athans, Optimal Control: An Introduction to the Theory and Its Applications. DOVER PUBN INC, Dec. 2006. [Online]. Available: https://www.ebook.de/de/product/5493358/michael_athans_peter_l_falb_optimal_control_an_introduction_to_the_theory_and_its_applications.html
[3] J. Klamka, Controllability and Minimum Energy Control. Springer International Publishing, 2019.
[4] C. Zavislak, A. Keow, Z. Chen, and F. Ghorbel, “AUV buoyancy control with hard and soft actuators,” IEEE Control Systems Letters, vol. 5, no. 6, pp. 1874–1879, dec 2021.
[5] H. Yan and K. T. Alfriend, “Approximate minimum energy control laws for low-thrust formation reconfiguration,” Journal of Guidance, Control, and Dynamics, vol. 30, no. 4, pp. 1182–1186, jul 2007.
[6] M. A. Sandidzadeh and M. R. Alai, “Optimal speed control of a multiple-mass train for minimum energy consumption using ant colony and genetic algorithms,” Proceedings of the Institution of Mechanical Engineers, Part F: Journal of Rail and Rapid Transit, vol. 231, no. 3, pp. 280–294, aug 2016.
[7] J. Maciejowski, Predictive control : with constraints. Harlow, England New York: Prentice Hall, 2002.
[8] D. S. Bernstein and A. N. Michel, “A chronological bibliography on saturating actuators,” International Journal of Robust and Nonlinear Control, vol. 5, no. 5, pp. 375–380, 1995.
[9] S.-T. Cheng, J.-S. Shih, and T.-Y. Chang, “GA-based actuator control method for minimizing power consumption in cyber physical systems,” Applied Intelligence, vol. 38, no. 1, pp. 78–87, jun 2012.
[10] W. Lee, H. Jeoung, D. Park, T. Kim, H. Lee, and N. Kim, “A real-time intelligent energy management strategy for hybrid electric vehicles using reinforcement learning,” IEEE Access, vol. 9, pp. 72 759–72 768, 2021.
[11] V. S. Simon, A. Siddarth, and D. Agonafer, “Artificial neural network based prediction of control strategies for multiple air-cooling units in a raised-floor data center,” in 2020 19th IEEE Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (ITherm). IEEE, jul 2020.
[12] E. Clemente, M. Meza-Sánchez, E. Bugarin, and A. Y. Aguilar-Bustos, “Adaptive behaviors in autonomous navigation with collision avoidance and bounded velocity of an omnidirectional mobile robot,” Journal of Intelligent & Robotic Systems, vol. 92, no. 2, pp. 359–380, Oct 2018.
[13] O. Peñaloza-Mejía, E. Clemente, M. Meza-Sánchez, C. Pérez, and F. Chavez, “GP-based motion control design for the double-integrator system subject to velocity constraint,” in GECCO ’17 Companion, Berlin, Germany, 2017.
[14] O. Peñaloza-Mejía, E. Clemente, M. Meza-Sánchez, and C. B. Pérez, “Evolving behaviors for bounded-flow tracking control of second-order dynamical systems,” Engineering Applications of Artificial Intelligence, vol. 78, pp. 12 – 27, 2019.
[15] M. Meza-Sánchez, E. Clemente, R. Villalvazo, and G. Olague, “Bounding velocity in tracking control of unicycle mobile robots with genetic programming,” in 2018 XX Congreso Mexicano de Robótica (COMRob), Sep. 2018, pp. 1–6.
[16] M. Meza-Sánchez, E. Clemente, M. C. Rodríguez-Liñán, and G. Olague, “Synthetic-analytic behavior-based control framework: Constraining velocity in tracking for nonholonomic wheeled mobile robots,” Information Sciences, vol. 501, pp. 436–459, 2019.

Capítulo 2.

Humanoids and Vision Systems

The Role of Inertial Stabilization in Walking Patterns

José Quintanilla*, Moises Perez*, Rodolfo Balderas*, Alejandro González†,
Antonio Cardenas*, Mauro Maya,* Davide Piovesan‡
**College of Engineering*, *Universidad Autonoma de San Luis Potosi*, San Luis Potosi, Mexico
Email: A{263740,263735,244124}@alumnos.uaslp.mx.
{antonio.cardenas, mauro.maya}@uaslp.mx
†*College of Engineering*, *CONACYT-Universidad Autonoma de San Luis Potosi*, San Luis Potosi, Mexico
Email: agonzalez@conacyt.mx
‡*Biomedical Engineering Program*, *Gannon University*, Eire, Pennsylvania, USA
Email: piovesan001@gannon.edu

***Abstract*—Impedance control has been often proposed as a stable control strategy for human walking. On the other hand, the continuous control of stiffness in human is energetically expensive. It has been shown that systems with unstable equilibrium points can be stabilized using inertial effects. A Kapitza's pendulum shows that it is possible to stabilize an inverted pendulum by means of a vertical oscillation at its base. This work provides a set of simulations that show how an impedance control strategy can be used for dynamic stabilization. The simulations presented are based on experimental data extracted from publicly available videos depicting the self-paced walking pattern of an unimpaired human and a that of a NAO humanoid robot. In both cases, the average lower limb stiffness can be maintained under the minimum level of stability if its modulation induces vertical oscillations on the trunk with frequencies compatible with human locomotion. The results reiterate the possibility of integrating the intermittent control of human balance under the umbrella of impedance control.**

***Index Terms*—Human Balance, Impedance Control, Kapitza's Pendulum, Human Locomotion**

I. INTRODUCTION

Human balance is important for the autonomy and well-being of individuals. Knowledge of the control mechanism used by humans to stand is paramount for the design of rehabilitation equipment and eventual intervention, as well as for the design of humanoid robots.

Various hypothesis on the control strategies employed by humans to balance have considered the regulation joint stiffness [1], and the intermittent control of their position [2]–[4]. A combination of these two strategies has recently been presented, whereupon the intermittent control was presented in the framework of stiffness control [5]. Human locomotion and balance is often modeled as an inverted pendulum with either one, or multiple degrees of freedom. This could, in part be described as a Kapitza's pendulum which shows a non-linear phenomenon where the base of an inverted pendulum is vibrated vertically, inducing an inertial effect capable of stabilizing the pendulum in an upright position [6]–[9]. This work posits that such a behavior (seen in the framework of stiffness regulation of the lower limb) is a generalized strategy used by individuals when walking, where the oscillation of the pelvis in the direction of gravity helps the torso to stay upright. Therefore, this work investigates this non-linear dynamic behavior in simulation, based on data extracted from publicly available videos of an unimpaired individual gait and a stable humanoid robot gait. The proposed analysis can be applied to the development of control algorithms for bipedal systems, which are subject to a wide range of conditions from the surrounding environment.

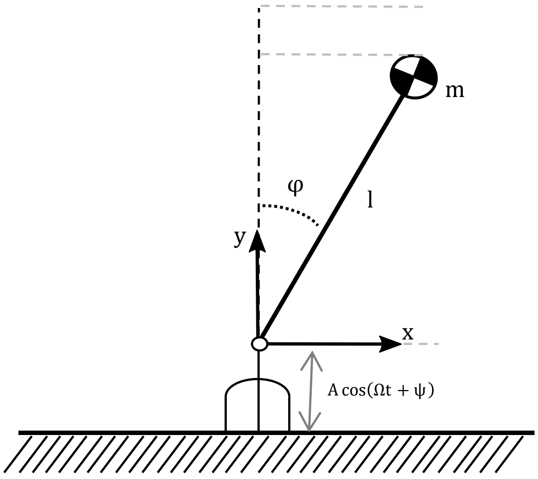


Fig. 1: Inverted Pendulum consisting of a point mass.

The following section details the dynamic model employed to study human motion and the rationale behind the introduction of a *Kapitza stiffness*, and is based on the author's previous work comparing unimpaired and impaired gaits [10]. The remaining sections will present the similarities between a human's and a humanoid's walking gaits when they are seen as Kapitza's pendulums. Finally, possible explanations for this behavior are discussed.

II. MODELING OF THE PENDULUM SYSTEM.

An inverted pendulum with a length l and a point mass m as shown in Figure 1 is considered. The angle of rotation of the pendulum is φ and the effect of vibrations is reflected in the vertical movement of the pendulum as $A\cos(\Omega t+\psi)$ where A is the amplitude of said vibration; Ω is the frequency of oscillation of the base; and ψ is the oscillation's phase. Using the Euler-Lagrange equations, the system's equations of motion can be obtained, starting from the position in x and y of the system.

$$\begin{aligned} x &= l\sin(\varphi) \qquad (1)\\ y &= l\cos(\varphi)+A\cos(\Omega t+\psi) \qquad (2) \end{aligned}$$

The time derivative of equations (1) and (2) gives the velocity components as:

$$\begin{aligned} \dot{x} &= l\cos(\varphi)\dot{\varphi} \qquad (3)\\ \dot{y} &= -l\sin(\varphi)\dot{\varphi}-A\Omega\sin(\Omega t+\psi) \qquad (4) \end{aligned}$$

Therefore, the kinetic energy of the system is computed as:

$$T=\frac{1}{2}m\left(\dot{x}^2+\dot{y}^2\right) \tag{5}$$

while the potential energy is given by:

$$V=mgy \tag{6}$$

Using the Euler-Lagrange formulation, the motion of the pendulum is described as:

$$\begin{aligned} L &= T-V \qquad (7)\\ 0 &= \frac{d}{dt}\frac{\partial L}{\partial\dot{\varphi}}-\frac{\partial L}{\partial\varphi} \qquad (8) \end{aligned}$$

from where:

$$\ddot{\varphi}=\left(\frac{g}{l}-\frac{A\Omega^2}{l}\cos(\Omega t+\psi)\right)\sin(\varphi) \tag{9}$$

Additional viscous and elastic elements may be considered by following the same procedure. That is, when the system contains a rotational spring and a damper whose coefficients are given by K and b respectively the motion of the system is described as:

$$ml^2\ddot{\varphi}+\left(mlA\Omega^2\cos(\Omega t+\psi)-mlg\right)\sin(\varphi)+K\varphi+b\dot{\varphi}=0 \tag{10}$$

A detailed description can be found in [7], [8].

We note that (10) shows a mass-spring-damper second order system where a frequency dependent component counters the effect of gravity. Additionally, the joint is subject to a rotational stiffness. For simplicity, we introduce $K_g=mlg$ as a gravity dependent coefficient, and for a single frequency oscillation:

$$K_k=mAl\Omega^2\cos(\Omega t+\psi) \tag{11}$$

However, when the vertical oscillation is composed of multiple frequencies K_k can be computed as:

$$K_k=ml\sin(\varphi)\sum_{i=1}^{n}A_i\Omega_i^2\cos(\Omega_i t+\psi_i) \tag{12}$$

In this paper, we shall refer to K as the joint/proper stiffness, K_g as the stiffness of gravity and K_k as the Kapitza's stiffness.

Using the stiffness regulation control framework, the human body movement can maintain a posture on its own without the need of a constant vertical oscillation. This happens under the hypothesis that the body's muscles act to maintain a certain stiffness that aids the body in reacting to gravity [7]. That is, the muscles act as an elastic network which can be modeled as a torsional spring (K) stabilizing the system. This work suggests that this muscle stiffness can be modulated, by an additional term resembling a Kapitza's stiffness (K_k). Said modulation would be used to reach a desired angular position with an overall decrease of the metabolic energy, due to co-contraction, necessary to maintain the proper stiffness (K).

A. Finding the suitable values of K and b

Consider a system without a vertical oscillation, that is $K_k=0$. It is possible to show that the non-linear system described in (10) will reach a steady position as a function of its joint stiffness (K). That is, K alone compensates the effect of gravity (K_g) on the pendulum. For a position where φ^* is a constant value ($\dot{\varphi}^*=\ddot{\varphi}^*=0$). It can be seen that:

$$K_g\sin\varphi^*=K\varphi^* \tag{13}$$

which may be in turn expressed as:

$$K=K_g\frac{\sin\varphi^*}{\varphi^*} \tag{14}$$

Note that the term $\sin\varphi^*/\varphi^*$ is positive within the interval $-\pi<\varphi^*<\pi$ and that $\lim_{\varphi^*\to 0}\sin\varphi^*/\varphi^*=1$. That is, in order for the pendulum to reach a constant upright position, a joint stiffness $K>K_g$ is required.

This work will study the case where the joint stiffness K is only a fraction of the gravitational stiffness K_g. This is represented in equation (15) where K is proportional to K_g by a constant $0<\alpha<1$.

$$K=\alpha K_g \tag{15}$$

That is, without the vertical oscillation the system will find its resting position at a suitable angle such that $\alpha=\frac{\sin\varphi}{\varphi}$.

When present, with the appropriate frequency (Ω) and amplitude (A) the Kapitza's stiffness K_k is capable of maintaining the vertical position of the pendulum as demonstrated by the literature [6], [7]. However, the work presented here assumes that the Kapitza effect will not be the main stabilizing effect (i.e. the average inertial stabilizing forces do not guarantee that the equilibrium position will be the upright). When combined with the joint stiffness, the addition of the vertical oscillation is seen to stabilize the system closer to the upright position than it would when no oscillation is present. Thus, this will show the superposition of the elastic and inertial stabilizing effects.

Finally, α will be determined by matching the observed behavior with a simulated pendulum from video recordings of a non-pathological gait and compared to that observed from a humanoid robot gait. We shall assume the damping coefficient of the system to be equal to a tenth part of the stiffness coefficient K_g produced by gravity:

$$b = \frac{K_g}{10} \tag{16}$$

B. The trunk as an inverted pendulum during waking

(a) Human gait[1] (b) Humanoid gait[2]

Fig. 2: Walking gait

We will analyze a system composed of a single segment inverted pendulum, which models the upper part of the body starting at the base of the pelvis. The angle of the pendulum, φ, will be measured with respect to the vertical line. Equation (11) shows that the Kapitza's stiffness is a function of frequency Ω and amplitude A of the vertical oscillation. This oscillation could be seen as an input to the dynamic system. Indeed, the vertical oscillation can be experimentally measured at the pelvis while walking. While in a classical definition of the Kapitza's pendulum the oscillation is an external input, this work will assume it to be the consequence of a neural modulation of the proper stiffness which regulates the second mode of vibration of the pendulum.

In order to create a numerical simulation of the system the values of Ω and A must be known as they are considered inputs. Here, two dynamic models are considered, they represent: i) The oscillation of the pelvis in a healthy person with no impairment in its gait. ii) The oscillation of the pelvis in a humanoid robot imitating a human gait. These models will be used to determine how much the vertical oscillation affects the output angle of the trunk and thus, the stabilization of the body.

Motion data was obtained from publicly available videos consisting of a person walking down the street[1], showing no signs of pathological gait, and a NAO humanoid robot[2] taken from [11] (see Figure 2). The software suite *TRACKER* was used to measure the displacements of the pelvis or hips; shown in Figures 3a and 4a. Specifically, *TRACKER* [12] is capable of tracking points in space on a video by analyzing each frame for said points using a similarity index.

The sagittal motion of the hip is also shown. This signal can be represented as a sum of sines at various frequencies by means of it's fast Fourier transform (FFT).

III. VERTICAL OSCILLATION IN WALKING PATTERNS

Previous studies that tried to reproduce a human like walking pattern approximated the movement of the pelvis with the solution of a second order model [11], with one single oscillatory mode. On the other hand, even a non-pathological gait produces a complex hip vertical bobbing which cannot be fully represented by a single, pure sine wave. In order to fully describe the hip vertical oscillation of the healthy human presented in Figure 3a we will make use of its frequency components determined through an FFT which amplitudes are depicted in figure 3b. The vertical oscillation for a non-pathological gait patter, is henceforth referred to as *HGP* (Human Gait Pattern).

Assuming each frequency in Figure 3b as a separate oscillator the Kapitza's stiffness will be written as in (12). Note that the maximum oscillation amplitude is below 2 cm and was found at 1.9 Hz. Also, there are no significant amplitudes after the 5.5 Hz mark. This frequency distribution may vary among individuals [13] but this example is sufficient to study the effect of the Kapitza's stiffness in the system's dynamics.

Similarly, a stable humanoid gait performed with a NAO robot was tracked (see figure 4a) and the FFT of its vertical oscillation was obtained (see figure 4b). Note that the maximum oscillation amplitude is below 5 mm and was found at 2.5 Hz. Also, there are no significant amplitudes after the 5.5 Hz mark.

The amplitude and frequency values thus obtained can be introduced into (12) to compute the Kapitza's stiffness.

IV. SIMULATION MODEL

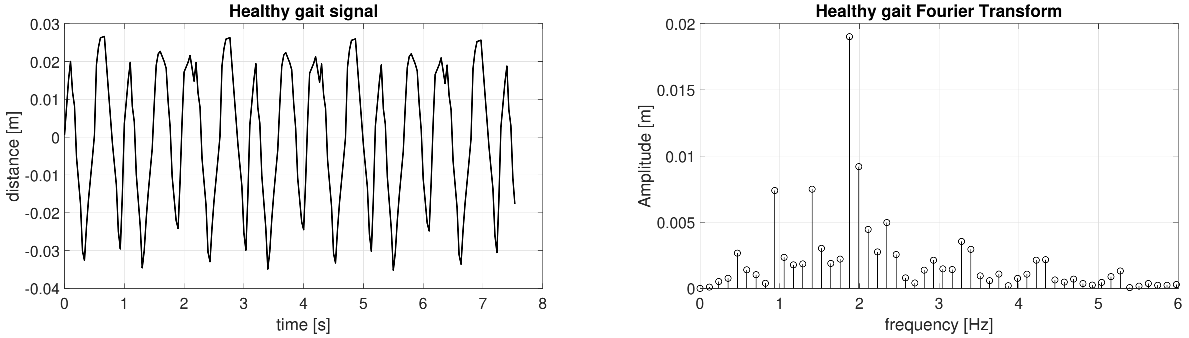
A custom model was created to solve the dynamic equations of the inverted pendulum (9). This model computes the Kapitza's stiffness K_k based on the FFT decomposition of the experimental data as detailed in (12), and allows for the use of a joint stiffness K. The output of the simulation is the angle φ. The value of K_g is computed from the system's mass.

We assumed that the human subject on the video recording has a height (H) of 1.75 m, and a mass (M) of 70 kg. The simulation considers only the upper part of the body as a single segment. Its mass distribution is shown in figure 5a which is based on the anthropometric data published by Winter [1]. For the human subject, its equivalent pendulum was assumed to have a length of ($l = 0.1802H$) and mass of ($m = 0.678M$). The NAO humanoid robot has a height (H) of 0.573 m, and a mass (M) of 5.182 kg. As is the case of the human subject simulation, the upper part of the robot's body was considered a single segment. Its mass distribution is shown in figure 5b [14]. The robot's equivalent pendulum was assumed to have a length of ($l = 0.2949H$) and mass of ($m = 0.6781M$).

---

[1] https://youtu.be/StwuCOayKBA?t=2

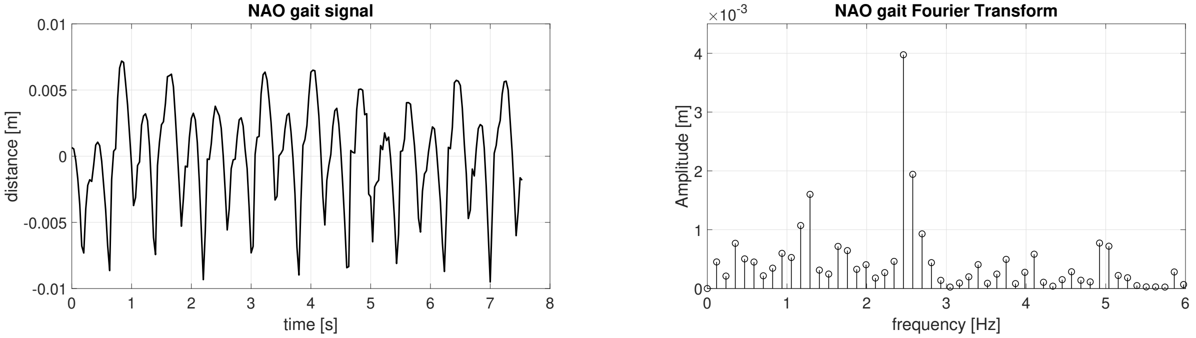
[2] https://youtu.be/OBGHU-e1kc0



(a) Vertical oscillation of the hip

(b) Fourier Spectrum

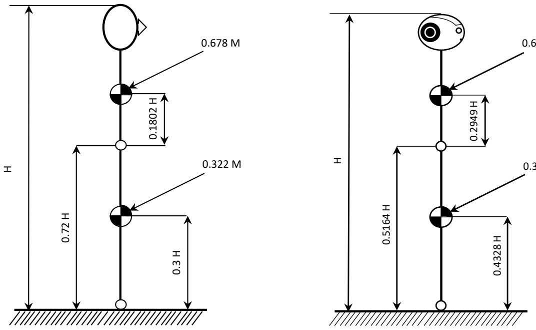
Fig. 3: Non-pathological gait pattern



(a) Vertical oscillation of the hip joint

(b) Fourier Spectrum

Fig. 4: Humanoid gait pattern



(a) Human mass distribution

(b) NAO robot mass distribution

Fig. 5: Subject mass distribution

On both instances the damping coefficient was assumed as in equation (16).

To study the Kapitza's pendulum model, we will simulate the system with and without the effect of Kapitza's stiffness, after having estimated the value of the proper stiffness for each the human and robot case.

V. RESULTS

If the proper joint stiffness is smaller than the gravity dependent stiffness $K < K_g$ the system will not stabilize around $\varphi = 0$. The angular deviation of the torso from the vertical is dependent on the ratio between the two stiffness, α. To determine the subject's (either the human or robot) proper joint stiffness a series of simulations were performed where the proper stiffness was modified until the output angle accurately reproduced the average angular deviation of the torso from the vertical measured from the videos.

For this, the Open Source *Tracker* software [12] was used to measure the vertical angle at which the subject maintained the trunk while walking. A line between the hip and sternum was created and φ was taken to be the average deviation with respect to the vertical line across all strides. While including the Kapitza's stiffness to the simulations, as computed in (11), the proper stiffness was manually tuned to match the equilibrium position in our simulation. By doing this, we approximated the ratio between the proper stiffness and the gravity induced stiffness to be $\alpha = 0.9785$ for the HGP, and $\alpha = 0.995$ for the NAO (see (15)).

Figures 6 and 7 show the effect that the Kapitza's stiffness has on the stabilization of the inverted pendulum when the aforementioned values of α are used. Both figures show, the value of φ when the Kapitza stiffness is not considered (in black); and its effect on the final output angle (in red).

In Figure 6, for a human gait, when no Kapitza stiffness is applied, the inverted pendulum stabilizes at around 20°. With the Kapitza stiffness it is able to stabilize at around 5°. A similar effect can be seen on Figure 7 were the equivalent pendulum is able to go from around 10° to 6° just by the addition of the Kapitza stiffness.

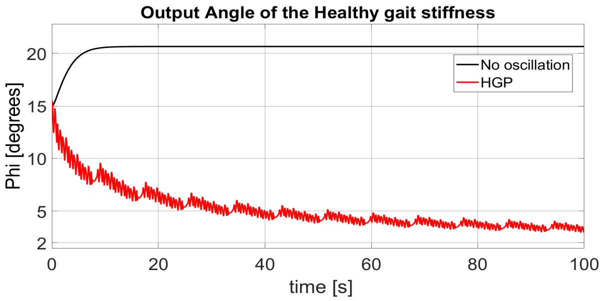


Fig. 6: Trunk angle stable position with and without the stiffness provides by the oscillation

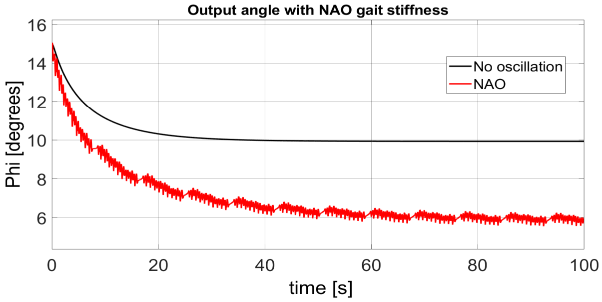


Fig. 7: Output angle stability value by the same stiffness

Interestingly, Figure 8 shows that the Kapitza's stiffness K_k is a time varying signal that is asymmetric. The signal is predominately negative. This implies that, if when adding K_k as a modulation of the proper stiffness K the average value of the sum of the two stiffness components is lower than having a proper stiffness alone. This mechanism allows for a reduction of metabolic energy in healthy humans.

VI. DISCUSSION AND CONCLUSIONS

Previous literature found that walking and running presents a variation of lower limb stiffness. The results shown here suggest that the natural oscillation of a walking pattern, has a stabilizing effect [15]–[18]. Researchers have tried to reproduce this variation of stiffness using nonlinear systems such as with Duffins or van der Pol oscillators [19]. However, nonlinear systems can be approximated by a second-order linear system with time varying stiffness [20]. A variable stiffness can be obtained with proper use of internal degrees of freedom [21]. This can be accomplished in humanoid bipeds by proper coordination of the degrees of freedom below the pelvis. There is evidence that neurological oscillators could be responsible for such time varying stiffness [22]. On the other hand, these oscillatory frequencies of the pelvis could be simply be an effect of motor noise [23]. If that were the case, a lighter system would be expected to be more prone to vibrational noise. On the other hand, it would appear that the range of frequencies characterizing the Kapitza's stiffness is not a function of the system's mass. Both systems shown here, while varying wildly on their mass do not have high frequency vibrations above 5.5 Hz.

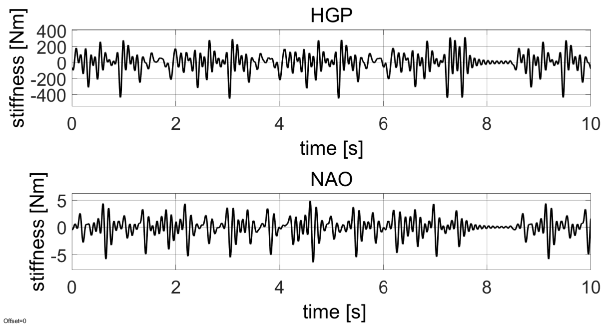


Fig. 8: Time varying characteristic of the Kapitza stiffness

While a change in 3% of the stiffness does not seem much, the equilibrium position of the torso can be changed dramatically and in the case of humans a deviation of the torso from the vertical position can exceed 20 degrees.

We have seen that a gait pattern with these characteristics can be implemented on a humanoid robot and could even be classified as *human-like*. In humanoids a compliant actuator reduces the energy cost of walking as well as reducing the effect of external disturbances [21]. Recently, new interest is being placed on the development of human-derived control strategies for robot control [24], [25]. Making use of the inertial balance provided by the Kapitza's stiffness such a could be implemented on robots to aid in the exploration of unstructured/rough terrain, such as caves or mountains.

Future work regarding the use of the Kaptiza pendulum model on human balance should extend this analysis to a double inverted pendulum to account for the dynamics of the leg segment. Additionally the segments could be considered with their full dynamic properties and not only as point masses. This will give would yield a closer representation to the human body as a inverted pendulum and allow for study of the effect that the vertical oscillation has on balance.

REFERENCES

[1] D. A. Winter, *Biomechanics and motor control of human movement*, 4th ed. Hoboken, NJ, USA: John Wiley & Sons, Inc, 2009.
[2] P. G. Morasso and V. Sanguineti, "Ankle Muscle Stiffness Alone Cannot Stabilize Balance During Quiet Standing," *Journal of Neurophysiology*, vol. 88, no. 4, pp. 2157–2162, oct 2002.
[3] I. D. Loram and M. Lakie, "Direct measurement of human ankle stiffness during quiet standing: the intrinsic mechanical stiffness is insufficient for stability," *The Journal of Physiology*, vol. 545, no. 3, pp. 1041–1053, dec 2002.

[4] I. D. Loram, M. Lakie, I. Di Giulio, and C. N. Maganaris, "The Consequences of Short-Range Stiffness and Fluctuating Muscle Activity for Proprioception of Postural Joint Rotations: The Relevance to Human Standing," *Journal of Neurophysiology*, vol. 102, no. 1, pp. 460–474, jul 2009.

[5] A. Cerda-Lugo, A. González, A. Cardenas, and D. Piovesan, "Modeling the neuro-mechanics of human balance when recovering from a fall: a continuous-time approach," *BioMed Eng Online*, vol. 67, no. 19, pp. doi.org/10.1186/s12 938–020–00 811–1, 2020.

[6] D. Acheson, "A pendulum theorem." *Proceeding of the Royal Society*, vol. 443(1997), pp. 239–245, 1993.

[7] R. Ramachandran and M. Nosonovsky, "Vibro-levitation and inverted pendulum: parametric resonance in vibrating droplets and soft materials." *Soft matter*, vol. 10(26), pp. 4633–9, 2014.

[8] E. Butikov, "An improved criterion for kapitza's pendulum stability," *Journal of Physics A: Mathematical and Theoretical*, vol. 44, p. 295202, 06 2011.

[9] T. Mullin, A. Champneys, W. Fraser, J. Galán, and D. Acheson, "The 'indian wire trick' via parametric excitation: A comparison between theory and experiment," *Proceedings of The Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 459, pp. 539–546, 03 2003.

[10] J. Quintanilla, M. Perez, R. Balderas, A. González, A. Cardenas, M. Maya, and D. Piovesan, "Inertial Stabilization of Upright Posture while walking," in *2021 10th International IEEE/EMBS Conference on Neural Engineering (NER)*. IEEE, may 2021, pp. 849–852.

[11] A. D. Ames, E. A. Cousineau, and M. J. Powell, "Dynamically stable bipedal robotic walking with NAO via human-inspired hybrid zero dynamics," in *Proceedings of the 15th ACM international conference on Hybrid Systems: Computation and Control*. ACM, 2012, pp. 135–144. [Online]. Available: https://youtu.be/OBGHU-e1kc0

[12] Open Source Physics, "TRACKER Software, Version 5.1.5," 2020.

[13] L. Bianchi, D. Angelini, and F. Lacquaniti, "Individual characteristics of human walking mechanics," *Pflügers Archiv*, vol. 436, no. 3, pp. 343–356, 1998.

[14] Aldebaran, "Nao Robot documentation Software, 1.14.5," 2017.

[15] R. Bortoletto, E. Pagello, A. D'Angelo, and D. Piovesan, "Stiffness-based control schemes for humanoids and animates," in *Workshop on Human Motion Modeling and Human-inspired Motor Control. IEEE-RAS Humanoid*, 11 2014.

[16] R. Bortoletto, E. Pagello, and D. Piovesan, "Lower limb stiffness estimation during running: The effect of using kinematic constraints in muscle force optimization algorithms," in *Simulation, Modeling, and Programming for Autonomous Robots. SIMPAR 2014*, vol. 8810, 10 2014.

[17] R. Bortoletto, E. Pagello, and D. Piovesan, "Effects of reserve actuators on optimization solutions: From muscle force to joint stiffness," in *2015 IEEE International Conference on Rehabilitation Robotics (ICORR)*, 2015, pp. 973–978.

[18] R. Bortoletto, S. Michieletto, E. Pagello, and D. Piovesan, *Human Muscle-Tendon Stiffness Estimation During Normal Gait Cycle Based on Gaussian Mixture Model*. Springer, 07 2014, vol. 302, pp. 1185–1197.

[19] P. Kumar, A. Kumar, and V. Racic, "Modeling of longitudinal human walking force using self-sustained oscillator," *International Journal of Structural Stability and Dynamics*, p. 1850080, 11 2017.

[20] M. Tomas-Rodriguez and S. Banks, *Linear, time-varying approximations to nonlinear dynamical systems. With applications in control and optimization*. Springer, 01 2010, vol. 400.

[21] D. Rodriguez-Cianca, M. Weckx, R. Jimenez-Fabian, D. Torricelli, J. Gonzalez-Vargas, M. Sanchez-Villamañan, M. Sartori, K. Berns, B. Vanderborght, J. L. Pons, and D. Lefeber, "A variable stiffness actuator module with favorable mass distribution for a bio-inspired biped robot," *Frontiers in Neurorobotics*, vol. 13, p. 20, 2019.

[22] K. Nishikawa, A. Biewener, P. Aerts, A. N. Ahn, H. Chiel, M. Daley, T. Daniel, R. Full, M. Hale, T. Hedrick, A. Lappin, T. Nichols, R. Quinn, R. Satterlie, and B. Szymik, "Neuromechanics: an integrative approach for understanding motor control." *Integrative and comparative biology*, vol. 47 1, pp. 16–54, 2007.

[23] J. Laczko, R. A. Scheidt, L. S. Simo, and D. Piovesan, "Inter-joint coordination deficits revealed in the decomposition of endpoint jerk during goal-directed arm movement after stroke," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 25, no. 7, pp. 798–810, 2017.

[24] G. Hettich, L. Assländer, A. Gollhofer, and T. Mergner, "Human hip–ankle coordination emerging from multisensory feedback control," *Human movement science*, vol. 37, pp. 123–146, 2014.

[25] A. V. Alexandrov, V. Lippi, T. Mergner, A. A. Frolov, G. Hettich, and D. Husek, "Human-inspired eigenmovement concept provides coupling-free sensorimotor control in humanoid robot," *Frontiers in Neurorobotics*, vol. 11, p. 22, 2017.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Preliminary design and experimental tests of a real-time stereoscopic foveated vision system

Eduardo Rodriguez-Orozco[1], Eusebio Bugarin[1], Juan Antonio Rojas-Quintero[2] and Ana Y. Aguilar-Bustos[1]

***Abstract*—This article describes the preliminary tests conducted on a real-time vision system for a humanoid robot with foveated stereoscopic vision. Two high performance camera models were tested to validate the proposal. The system is designed with the Debian Preempt RT as its real-time operating system. A cyclical algorithm is used to acquire statistical data from our system and to show its real time behavior. The experimental results illustrate the satisfactory performance of the vision system.**

***Index Terms*—Stereoscopic Vision System, Real-Time, Humanoid Robot, Preemp RT**

I. INTRODUCTION

During the last decades, Robotics has definitely influenced many areas of society; including the primary and industrial sectors (mining, automotive, etc.) and the service sector (education, medicine, etc.). This has led to an interesting and productive interaction between various areas of science; such as cybernetics, mechanics, automatic control, informatics, bioengineering, electronics, etc. Some recent applications of robots are the deactivation of bombs [1], terrain mapping [2], transport of materials [3], search of objects [4], rescue of people [5]; among many more. It is important to note that robotics applications can be on land, air, water, or even in space.

In the context of applications in cooperation with humans, perhaps humanoid robotics has contributed more. In the literature there are reports of success in their applications, such as: ASIMO, which is able to walk and recognize faces and voices, identify gestures, postures and moving objects; ICUB, which is a research grade humanoid robot designed to develop and test embedded artificial intelligence algorithms; TOPIO, which is a humanoid robot designed to play table tennis against a human being; just to mention a few. These examples differ in purpose but share many characteristics: actuators, processing units, sensors, power supply and software (step control, behavior control, vision system, etc.).

One of the most important sensors for the perception of the robotic system itself and its environment is the camera. Therefore, to extract information from the captured images, real-time robot vision algorithms are mandatory. To achieve this issue, humanoid robots use both monocular and stereo vision systems [6], [7]. In a monocular vision system, it is known that the system cannot provide accurate information about the depth (distances along the optical axis) of the environment. But, a stereo vision system solves this problem with its two-camera composition. However, there is still the problem of covering wide scenes with good detail in some region of interest. The above can be solved with a foveal vision system. In our case, we propose a foveal vision system composed of 4 cameras (two per eye, one in charge of the wide scene and the other one of the detail of the region of interest).

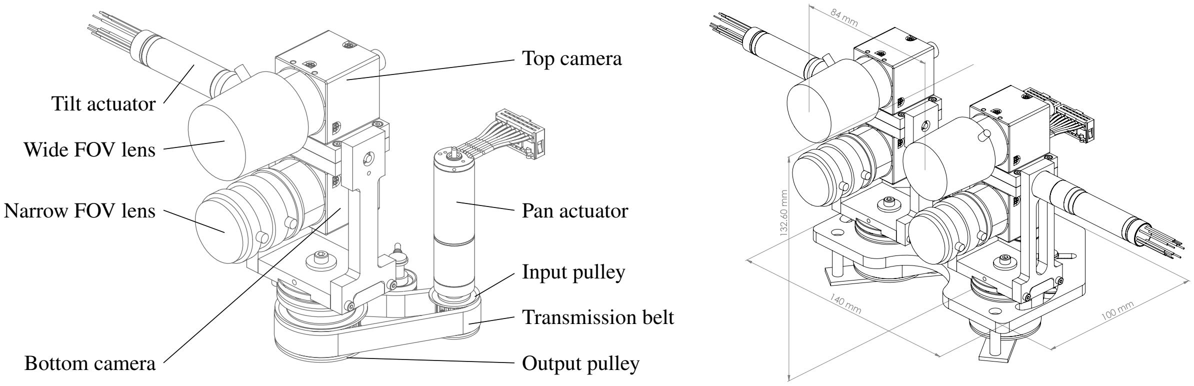
A real-time system is one whose logical correctness is based on both the correctness of the outputs and their timeliness. That is, a real-time system is deterministic and must fulfill the entrusted task in the specified time. This must be done in conjunction with other tasks in a multitasking system by taking waiting times into account while exchange of information with other devices takes place. In this sense, our proposal for the real-time stereoscopic foveated vision system must be capable of acquiring the images, processing them and extracting the necessary information at the times specified by the user (as long as the devices in the vision system are capable of reproducing these time demands).

Currently, there is a large number of computer vision systems for robots. In [8] it is presented the collaborative humanoid robot ARMAR-6, which has been developed to perform a wide variety of complex maintenance tasks in industrial environments (collaborating with human workers). Its head has the computer vision system with 2 pairs of stereo cameras (one pair corresponding with 2 cameras Point Gray Flea3) and a 3D depth sensor. It is important to note that one pair runs at 30 frames per second (FPS) and the other one at 15 FPS. In [9] it is proposed a framework that uses an overhead camera on a drone to convert arbitrary horizontal rectangular regions into interactive surfaces. For the onboard PC, an ECS LIVA (a CPU Intel Bay Trail-M SoC Celeron N2907 1.58 GHz with an O.S. Windows 8) was chosen because of its high computational performance and low weight. A Point Grey Flea3 USB3.0 camera was used as well. In [10] it is described a CV algorithm for optical flow using Pan/Tilt/Zoom cameras with theoretical and experimental measures of accuracy. The equipment used include a Dell M6600 Precision laptop with Intel R CoreTM i7-2820QM CPU 2.3 GHz with a NVIDIA Quadro 4000M graphics card and 2.0 GB dedicated graphics memory. In [11] it is presented an artificial vision process as well as its implementation through Ubuntu 16.04 and ROS

This work was financially supported by Secretaría de Educación Pública (SEP) and Consejo Nacional de Ciencia y Tecnología (CONACYT), under Grant A1-S-29824; and Tecnológico Nacional de México (TecNM) projects.

[1] Eduardo Rodriguez-Orozco, Eusebio Bugarin and Ana Y. Aguilar-Bustos are with Tecnológico Nacional de México/I.T. Ensenada, Ensenada, Baja California, Mexico. `erodriguez@ite.edu.mx`

[2] Juan Antonio Rojas-Quintero is with CONACYT/Tecnológico Nacional de México/I.T. Ensenada, Ensenada, Baja California, Mexico. `jarojas@conacyt.mx`



(a) Right "eye"   (b) Vision system

Fig. 1. CAD model of the designed foveated active vision system: (a) right eye of the vision system with description of the composing elements; (b) vision system with global dimensions.

using OpenCV-C++ and Qt libraries. An RGB camera with a resolution of 800x600 pixels at a frame rate of 30 Hz was chosen for the vision system. Some related works showing the FPS obtained in their proposals are mentioned below: [12] 30 FPS for one vision subsystem and 15 FPS for another vision subsystem, [13] with 25 FPS, [14] image with 15 FPS and depth with 30 FPS and [15] less than 30 FPS.

The rest of the paper is as follows. The description of the proposed vision system is detailed in Section II. The experimental results are provided in Section III. And, finally, the conclusions and future scope are described in Section IV.

II. VISION SYSTEM DESCRIPTION

Our vision system currently consists of three parts: the mechanical part, the real-time computer system, and two high-performance cameras per eye.

A. Mechanical desing

The mechanical design methodology behind our bio-inspired stereoscopic foveated active vision system was presented in [16]. Figure 1 shows the CAD model of our system and details the main composing elements (Figure 1(a)) along with the global dimensions of the prototype (Figure 1(b)). This system is intended to serve as the eyes of a humanoid robot head that is currently under development. Each eye is composed of two cameras, one with wide-angle lens and the other with a narrow-angle lens. This way, the vision system is able to see a scene with a wide field of view and to capture more detail in the center of the image throuhg the narrow field of view lens.

Based on the premise that a humanoid robot should hold resemblance with the human being in terms of functionality and/or appearance [6], [7], [14], our system is inspired by the human being visual apparatus in terms of kinematics, dynamics and foveal and peripheral vision capabilities. It should be noted that we chose to focus on functionality resemblance. Table I lists the targeted design parameters that were selected and that our system is capable of producing.

TABLE I
TARGETED BIO-INSPIRED DESIGN PARAMETERS

PARAMETER	VALUE
Eye angular velocity	$450\,°\,\mathrm{s}^{-1}$
Eye angular acceleration	$20\,000\,°\,\mathrm{s}^{-2}$
Eye abduction / adduction	$23°$ / $67°$
Eye elevation / depression	$51°$ / $31°$
Inter-pupillary distance	$61\,\mathrm{mm}$ $(\pm 1\,\mathrm{mm})$
Eye Field of view	$170°$ horizontal / $135°$ vertical
Focusing range	$25\,\mathrm{mm}$ – ∞ (wide lens)

For the design of each eye, we selected a traditional 2 Degrees of Freedom (DOF) panning and tilting structure (see Figure 1(a)). The actual robot is shown by Figure 2 in its current state of development.

Fig. 2. Actual foveated active vision system in its current state. Cameras are mounted on the left eye. Sensors are visible on the right eye.

B. Real-time computer system

The real-time computer system designed consists of one Intel® NUC Kit with 7th Generation Intel® Core ™ Processors

[17] with the Debian Preemp RT 10.9 [18] real-time operating system. The programming language selected was C++ together with a special package called Visual Servoing Platform (ViSP) [19].

As previously mentioned, one of the main objectives was to analyze the real-time performance of the designed platform. To meet this objective the Cyclictest test has been applied (the methodology is described in OSADL [20]); which executes a test and produces a dependent latency graph. Cyclictest accurately and repeatedly measures the difference between a thread's expected wake-up time and the time it actually wakes up to provide statistics on system latencies. It can measure latencies in real-time systems caused by hardware, firmware, and operating systems [21]. Figure 3 shows the latency generated by the threads created continuously to stress the 4 CPUs of the used architecture, a maximum latency of 35 μs was observed.

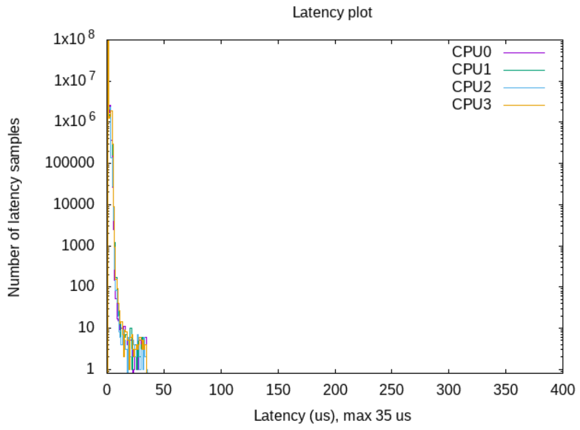


Fig. 3. Real-time operating system latency.

C. Cameras

One relevant aspect of a computer vision systems is the characterization of its captured frames; which involves, basically, its format, mode of operation, and frame rate. They are determined by the hardware and software used in the computer vision system. For purposes of this work, two Flir cameras were tested; specifically the model Flea3 USB 3.0 FL3-U3-13S2C, RoHS 1.3 MP, Color [22] and the model Grasshopper3 USB 3.0 GS3-U-23S6C, RoHS 2.3 MP, Color.

The ViSP libraries need the camera drivers to use the cameras in an optimal way. The camera drivers [23] are distributed by the same manufacturing company Flir, which are compatible with both Windows and Linux operating systems. Because we are interested in fast real-time performance, we have selected (see [24]) three modes of operation (0, 4 and 7) and two image sizes (320x240 and 640x480).

III. EXPERIMENTAL RESULTS

The algorithm for the experiments is illustrated in Figure 4. The first step is to configure and generate the threads of the application. The next two steps execute the camera threads, each one individually generates all the realated variables. The first one is for the Flea3 camera and the second one for the Grasshooper3 camera. Each camera thread is programmed with the same algorithm, which is shown in Figure 5. Finally, an image is saved at the end of the experiment.

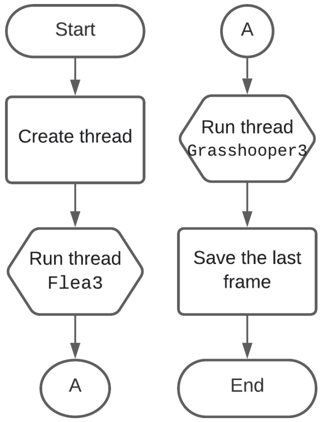


Fig. 4. Experiment algorithm flow chart.

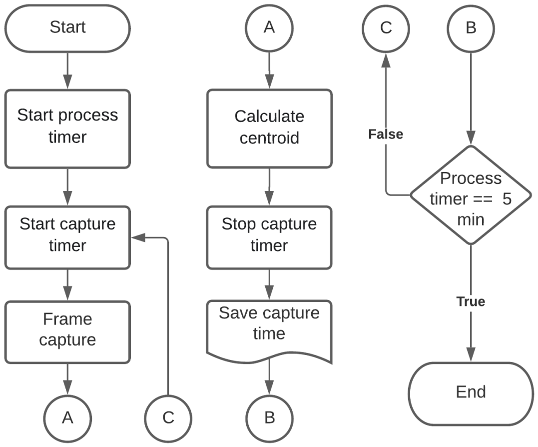


Fig. 5. Flow chart for each camera thread.

The first step in each camera thread (see Figure 5) is to start a process timer. Then a capture timer is also set to measure the time from the captured frame (then the image processing) to just before the next frame. Here, because the objective is to have a fast real-time vision system, by image processing we refer to the sequential procedure of thresholding the image (so that the original color image is black and white) and then to get the centroid of some region of interest. The centroid of the region of interest $(\bar{x}, \bar{y})$ is obtained by means of

$$\bar{x} = \frac{\sum_x \sum_y x f(x,y)}{\sum_x \sum_y f(x,y)} \tag{1}$$

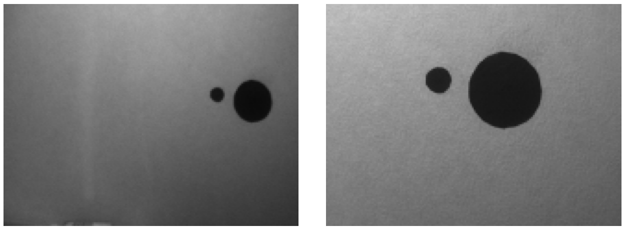
$$\bar{y} = \frac{\sum_x \sum_y y f(x,y)}{\sum_x \sum_y f(x,y)} \tag{2}$$

where

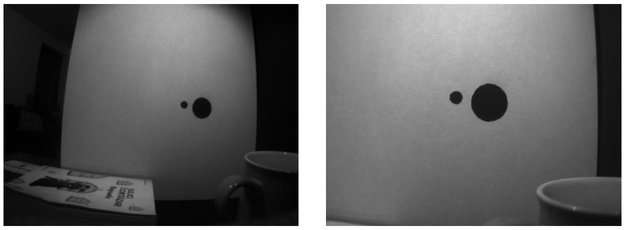
$$f(x,y) = \begin{cases} 1, & \text{if } (x,y) \text{ is black} \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

After calculating the centroid, the capture timer must be stopped and the consumed time is saved in a file. Finally, if the process timer is equal to 5 min then the thread is finished.

Now, for the experimental validation, consider the set of images that are shown in Figure 6; where the images in the left side were taken by the Flea3 camera (with 320x240 image size, Figure 6(a), and 640x480 image size, Figure 6(c)), and its counterpart were taken by the Grasshopper3 camera (with 320x240 image size, Figure 6(b), and 640x480 image size, Figure 6(d)).



(a) Image taken with 320x240 Flea3 camera (b) Image taken with 320x240 Grasshopper3 camera

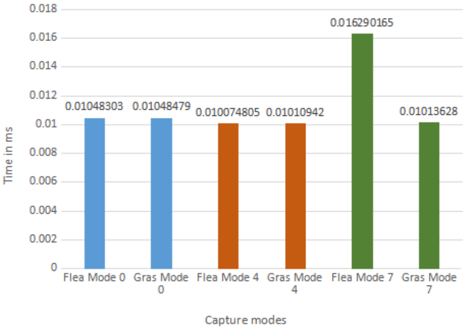


(c) Image taken with 640x480 Flea3 camera (d) Image taken with 640x480 Grasshopper3 camera

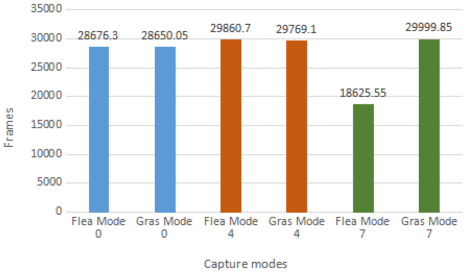
Fig. 6. Images of the experiment taken with two different cameras and formats. Figures 6(a), 6(c) with Flea3 camera and 6(b), 6(d) with Grasshopper3 camera.

A total of 20 repetitions of 6 different experiments were carried out. Figure 7 presents the results for the case with an image size of 320x240 pixels; where Figure 7(a) shows the time average for each capture cycle (mode 7 stands out as the slowest one), and Figure 7(b) illustrates the average number of photos taken during the experiments. It can be deduced, from these figures, that the worst option is mode 7; even so, other data must be considered in order to select the best one and we decided to take the most stable configuration.

Figure 8 shows the results of the experiments using an image size of 640x480 pixels. Here the data collected are as expected, fewer captures were acquired. This is a consequence of the number of pixels in the image size.



(a) Time average



(b) Average of captured frames

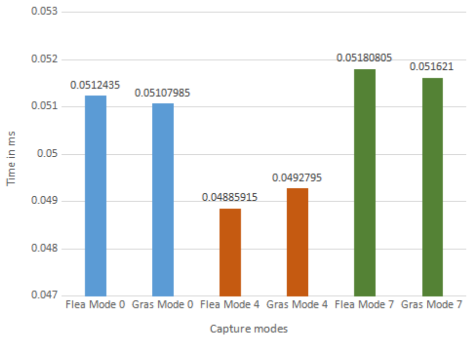
Fig. 7. Resuls for the experiments with 320x240 image size.

Table II illustrates some statistical data based on the results. It is important to note that mode 7, with 320x240 image size, was taken as the best option because it is the most stable presented configuration.

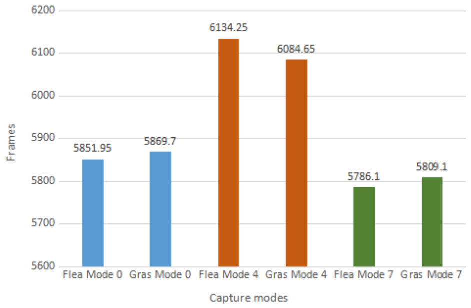
TABLE II
SOME EXPERIMENT STATISTICS.

	Average	Dev. STD	Variance
Flea mode 7 320x240	18006.55	4.6166	21.3131
Gras mode 7 320x240	30618.85	1737.5898	3019218
Flea mode 0 640x480	5851.95	173.4254	30076.37
Flea mode 4 640x480	6134.25	89.8782	8078.092
Flea mode 7 640x480	5786.1	104.1416	10845.46

Finally, Figure 9 depicts the exact capture time of a frame with mode 7 and image size of 320x240 pixels. The experiment with the Flea3 camera can be observed in blue and in red with the Grasshopper3 camera. The Flea3 camera performs better than the Grasshopper3 camera, but after 14 captures the Grasshopper3 camera acquires more frames.



(a) Average of captured frames



(b) Average of captured frames

Fig. 8. Resuls for the experiments with 640x480 image size.

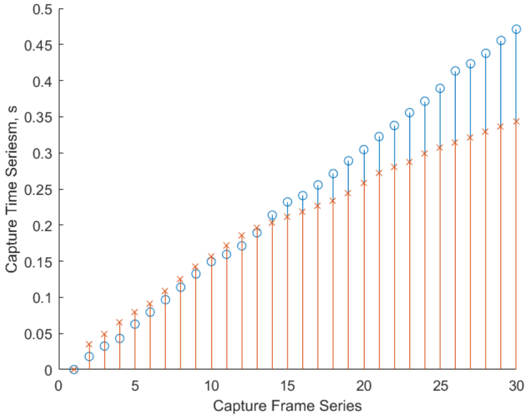


Fig. 9. Capture times, in blue the experiment with the Flea3 camera and in red with the Grasshopper3 camera (both in mode 7 with 320x240 image size).

IV. CONCLUSION AND FUTURE SCOPE

A computer system is considered a hard real-time system if, and only if, it meets strict deadlines for the completion of its actions or tasks. These deadlines must always be met, otherwise the system is cosidered to have failed. In this paper we have described a preliminary design and experimental tests of a real-time foveated stereoscopic vision system as part of a humanoid robot. The presented real-time vision system maintains a good behavior under stress. Our aim is to design a foveated vision system that operates under the regime of a hard real-time embedded system, as a future work. Also, our research project seeks to use Epos4 controllers (maxon motor) for the actuators in the mechanical part, with the EtherCAT protocol for the communication between devices and several object tracking algorithms under the proposed real-time stereoscopic foveated vision system.

REFERENCES

[1] H. Qi, W. Wang, L.-z. Jiang, and L.-q. Fan, "Automatic target positioning of bomb-disposal robot based on binocular stereo vision," 10 2007.

[2] P. Fankhauser, M. Bjelonic, C. Dario Bellicoso, T. Miki, and M. Hutter, "Robust rough-terrain locomotion with a quadrupedal robot," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 5761–5768.

[3] J. Qian, B. Zi, D. Wang, Y. Ma, and D. Zhang, "The design and development of an omni-directional mobile robot oriented to an intelligent manufacturing system," *Sensors*, vol. 17, no. 9, 2017. [Online]. Available: https://www.mdpi.com/1424-8220/17/9/2073

[4] R. Bähnemann, D. Schindler, M. Kamel, R. Siegwart, and J. Nieto, "A decentralized multi-agent unmanned aerial system to search, pick up, and relocate objects," in *2017 IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, 2017, pp. 123–128.

[5] R. Luža, J. Vyroubalová, M. Apjar, and M. Drahanský, "Control algorithms for rescue robot ruda," in *Modelling and Simulation for Autonomous Systems*, J. Mazal, Ed. Cham: Springer International Publishing, 2018, pp. 204–222.

[6] J. A. Rojas-Quintero and M. C. Rodríguez-Liñán, "A literature review of sensor heads for humanoid robots," *Robotics and Autonomous Systems*, vol. 143, p. 103834, 2021. [Online]. Available: https://doi.org/10.1016/j.robot.2021.103834

[7] S. Saeedvand, M. Jafari, H. S. Aghdasi, and J. Baltes, "A comprehensive survey on humanoid robot development," *The Knowledge Engineering Review*, vol. 34, 2019.

[8] T. Asfour, L. Kaul, M. Wächter, S. Ottenhaus, P. Weiner, S. Rader, R. Grimm, Y. Zhou, M. Grotz, F. Paus, D. Shingarey, and H. Haubert, "Armar-6: A collaborative humanoid robot for industrial environments," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, 2018, pp. 447–454.

[9] K. Tanaka, N. Tochihara, T. Sato, and H. Koike, "A real-time image processing framework with an aerial overhead camera for sports," in *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*, ser. AVI '18. New York, NY, USA: Association for Computing Machinery, 2018. [Online]. Available: https://doi.org/10.1145/3206505.3206520

[10] D. D. Doyle, A. L. Jennings, and J. T. Black, "Optical flow background estimation for real-time pan/tilt camera object tracking," *Measurement*, vol. 48, pp. 195–207, 2014. [Online]. Available: https://doi.org/10.1016/j.measurement.2013.10.025

[11] A. D. Hernández-Rojas, "Development of vision-assisted robotic manipulation system," in *XXI Congreso Mexicano de Robótica*. Asociación Mexicana de Robótica e Industria, 2019, pp. 144–150.

[12] T. Asfour, K. Welke, P. Azad, A. Ude, and R. Dillmann, "The karlsruhe humanoid head," in *Humanoids 2008-8th IEEE-RAS International Conference on Humanoid Robots*. IEEE, 2008, pp. 447–453.

[13] P. Doherty and P. Rudol, "A uav search and rescue scenario with human body detection and geolocalization," in *Australasian Joint Conference on Artificial Intelligence*. Springer, 2007, pp. 1–13.

[14] J. Englsberger, A. Werner, C. Ott, B. Henze, M. A. Roa, G. Garofalo, R. Burger, A. Beyer, O. Eiberger, K. Schmid, and A. Albu-Schäffer, "Overview of the torque-controlled humanoid robot toro," in *2014 IEEE-RAS International Conference on Humanoid Robots*, 2014, pp. 916–923.

[15] H. Duan and Q. Zhang, "Visual measurement in simulation environment for vision-based uav autonomous aerial refueling," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 9, pp. 2468–2480, 2015.

[16] J. A. Rojas-Quintero, J. A. Rojas-Estrada, E. A. Rodríguez-Sánchez, and J. A. Vizcarra-Corral, "Designing a bio-inspired foveated active vision system," in *XXII Congreso Mexicano de Robótica*. Tijuana, Mexico: Asociación Mexicana de Robótica e Industria, 2020.

[17] Intel, "Intel® nuc kit nuc7i7bnh products specifications," Available at https://ark.intel.com/content/www/us/en/ark/products/95065/intel-nuc-kit-nuc7i7bnh.html (2021/05/24).

[18] Debian, “Debian el sistema operativo universal,” Available at https://www.debian.org/ (2021/05/24).
[19] ViSP, “Visp,” Available at https://visp.inria.fr/ (2021/05/24).
[20] OSADL, “Osadl:osadl - open source automation development lab eg,” Available at https://www.osadl.org/Create-a-latency-plot-from-cyclictest-hi.bash-script-for-latency-plot.0.html?&no_cache=1&sword_list[0]=script (2021/05/24).
[21] T. L. Foundation, “realtime:documentation:howto:tools:[wiki],” Available at https://wiki.linuxfoundation.org/realtime/documentation/howto/tools/cyclictest/start (2021/05/24).
[22] T. FLIR, “Flea3 usb3 — teledyne flir.” Available at https://www.flir.com/products/flea3-usb3/ (2021/05/24).
[23] ——, “Flycapture sdk — teledyne flir.” Available at https://www.flir.com/products/flycapture-sdk/ (2021/05/24).
[24] ——, “Flea3 technical reference.” Available at https://www.ptgrey.com (2021/05/24).

Towards the development of the capability of drawing portraits with the humanoid robot DarwinOP

Pablo Sierra Costa
UPIITA-IPN
Mechatronics
psierrac1200@alumno.ipn.mx

Alexis Adrián Ortiz Olvera
Universidad Tecmilenio
Robotics Headcoach
alexis_ortiz@tecmilenio.mx

Juan Manuel Ibarra Zannatha
Cinvestav
Automatic Control Department
jibarra@cinvestav.mx

Abstract

This paper presents the development of an application that enables a small commercial humanoid robot, the DarwinOP, to draw the face of a person. The corresponding project is realized in three phases: the artificial vision system, the direct and inverse kinematics and the control system. In this paper it is presented: *i)* The design and the implementation of the vision system algorithms, and the obtained results that generate a group of lines that represent the portrait to draw; *ii)* The development of the direct and inverse kinematics of the robotic arm of the humanoid DarwinOP which has been modified adding a joint: flexion-extension of the wrist; and *iii)* The design of the working station and the results obtained previously to the final implementation in the internal computer of the DarwinOP.

1. Introduction.

The Robotics field has had a considerable growth in the last twenty years, not only for industrial robots (manipulator or process arms), but for the modern and innovative service robots. During the 2017, 381,000 Industrial robots were sold around the world; this means a growth of 30% with respect to the industrial robots sold in 2016; this represents a 16,200 millions of dollars, 21% more than in 2016 (data of the International Robotics Federation, published in its *World Industrial Robot Report* 2017).

Also, the research in robotics and the technology innovation that comes with it are focused on the service robots, that have been introduced into a vast number of applications that keep growing. Their applications vary between the entertainment, the exploration of hostile and/or hard to reach places, and the medical or home environments. Moreover, the service robots have different morphologies and versatile mechanical architectures like mobile robots, humanoids, drones or autonomous vehicles. Therefore, in the service robot applications, the robot accomplishes tasks with some economic, social, educative or productive interests.

1.1. Problem description.

With this background we propose a project where a humanoid robot, a DarwinOP, has to make an *artistic task* that consist in drawing the portrait of a person perceived by its artificial vision system. The proposed solution to solve this problem has been divided in three phases: *i)* Artificial vision: where the robot should understand the picture taken by its camera and represent it by a group of lines that constitute the portrait of the chosen person; *ii)* The transformation of this group of lines into movements of the humanoid robotic arm that follow these lines; and *iii)* The control of the servomotors in the joints of the arm so that the humanoid is able to draw the portrait on the surface of the drawing paper.

The interest of this project is given on one hand, by the artistical nature of the application rare in the robotics field, since it involves drawing a portrait of a person; On the other hand, each of the three stages contemplated in this project require a wide range of knowledge of Robotics and associated programming.

Thus, the vision phase requires to implement efficient methodologies in the two stages of any vision system: Improvement of the image and enhancement of features of interest, and image feature extraction; the objective is to represent the portrayed subject with a minimum of strokes and shadows that requires developing or using specific segmentation techniques. In particular, the transformation phase of the mentioned set of strokes into movements of the humanoid arm requires obtaining the Direct kinematics of the robot arm, as well as a methodology to solve the inverse kinematics so that it does not have problems of continuity or singularity. This phase it’s called the Kinematic phase.

Finally, the phase related to the control of the movements of the arm involves knowledge of the DarwinOP humanoid programming techniques as well as the implementation of the vision algorithm, the one corresponding to the inverse kinematics and the development of a control scheme that allows synchronization of image capture tasks, feature extraction, kinematics resolution and control of the movements of the humanoid, both from the head to look at the person to be portrayed and from the arm to draw it.

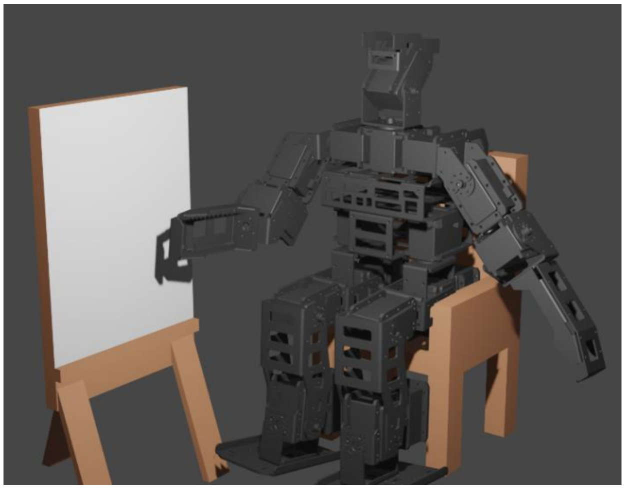


Figure 1. Workstation of the drawing humanoid robot.

1.2. Workstation

This section describes the characteristics of the workstation necessary for the humanoid robot to carry on with its drawing task. Thus, the person who will be portrayed by our robot is placed on a background of any homogeneous color without texture. Illumination will come from a source placed laterally to the model and slightly higher than the face, to highlight the shape of the face and generate the shadow areas necessary to give a three-dimensional look to the image. On the other hand, since the robot's camera has a fixed focal length, it must be placed at a distance and in such a posture that you can see the face of the person serving as the model.

The arms of the DarwinOP humanoid robot have only three degrees of freedom (DoF): shoulder flexion-extension (*pitch*), shoulder adduction-abduction (*roll*) and elbow flexion-extension (*pitch*). This greatly limits the drawing capabilities of our robot, so it was decided to add an extra DoF: wrist flexion-extension (*pitch*). In this way it has two DoF with parallel axes that allow positioning the drawing instrument, which we will call a pencil, in the plane of the paper on which the drawing will be made, although the size of the workspace covered is small. To expand this workspace, the DoF corresponding to shoulder adduction-abduction should be positioned so that the axis of the shoulder flexion-extension DoF is parallel to those of the elbow and wrist. Although with this position the robotic arm will have a greater reach and therefore a more significant workspace, the resulting kinematic chain is redundant which will complicate obtaining a unique and closed solution for the inverse kinematics, problem covered in Section 3.2 of this document.

Finally, the kinematics of the humanoid, as well as the posture determines the size of the paper in which it will be drawn, as well as the position in which it should be placed in the workplace, which is schematically illustrated in Fig. 1.

1.3. Paper organization.

This project corresponds to the Terminal Work (undergraduate thesis) developed by the first author of this article to obtain the degree of Mechatronics Engineer from the UPIITA-IPN (Engineering and Advanced Technologies Interdisciplinary Professional Unit of the National Polytechnic Institute, Mexico). To date, the Artificial Vision and Kinematics phases have already been completed, leaving the implementation in the DarwinOP humanoid.

After this first introductory chapter, Chapter 2 is dedicated to the presentation of the system of Artificial Vision developed for the humanoid robot, which includes all algorithms for image processing to eliminate lighting problems (bright zones, shadows) and other noise, as well as to emphasize the features of interest of the image in order to facilitate its extraction in the image analysis stage where the database will be obtained with the specific instructions of motion for the robot on the drawing plane. Chapter 3 is dedicated to the presentation of the kinematic models that were developed for the DarwinOP humanoid arm in its modified version with an extra joint (wrist flexion-extension), it is shown the development of our inverse kinematic model of this redundant kinematic chain that we propose to obtain the instructions in the joint space of the humanoid DarwinOP arm from the lines and points specified in the drawing plane. Chapter 4 is devoted to the presentation of the design of the workstation and the application, as well as the results obtained so far in the development project of the portrait artist application for the humanoid robot DarwinOP based on OpenCV. Finally, in Chapter 5 some comments are made on the work carried out, as well as the results obtained and the continuation of the project.

1.4. State of art.

As it was said before, the art applications for autonomous robotics are not reviewed in the state of art. Being said that, there are found some similarities between the application presented here and other projects. For example, in [5] it is described the control of a planar robot with 3 joints that is directed by a pointer held by someone and captured by cameras. Also, in [6] we can observe the similarities in the vision-controlled robot where the task is to collocate objects in a plane according to a reference image and a camera.

At last, we can find in [7] some similarities in the platform where the humanoid controlled by vision is used to distinguish a ball in a field, go towards it and kick it in a direction directed by the cameras as well.

2. Artificial Vision Phase

An Artificial Vision System (AVS) for a robot (mobile robot, manipulator arm, humanoid, drone, autonomous vehicle, etc.) has the mission of generating a description of the scene contained in an image (or series of images) that allows the robot to understand the scene and interact in a smart an efficient way with the environment captured in the image in order to execute the desired task. It should be noted that the description provided by the AVS must have two fundamental properties: it must consider only the information useful for its task, and it is based exclusively on the visual information contained in the image (although it often uses additional information about the environment and or information coming from other sensors of the robot).

In this context, the AVS of the portrait humanoid will have the mission of generating a description of the face of the person to be portrayed to serve as a movement command for the humanoid robot to make the lines contained in this description. These strokes must consist of a minimum set of lines that clearly define the face, its shaded areas, as well as the level of gray of the shadows. It is obvious that the lines that the robot can make are of a single tone (the one that the pencil or marker used has), so the shadows must be represented with a density of lines proportional to the gray level of the shadow considered. So, below are presented the different image processing methods used, as well as the algorithmic image analysis used to obtain the desired result.

2.1. Methodology of the developed AVS

The developed AVS contemplates different algorithms for the Digital Image Processing (PDI), implemented on OpenCV. The methodology developed to obtain the final image that contains only the strokes to be made by the robot consists of two stages. The first one aims to generate an image with the lines that define the face to be drawn, while the second aims to obtain the shadow areas that appear in the original image encoded in one of two bands of darkness. Now, each of these shadow levels will be assigned a diagonal line or hatched pattern, where its density depends on the grey level they should represent.

2.2. Lines Extraction Stage

This first stage of the AVS consists of four steps: Preprocessing, edge determination, edge image cleaning, and small lines removal. These four steps are briefly presented below.

Preprocessing. At this stage, the image captured by the camera is imported and, in order to improve it, a statistical filter (median) is applied to it that eliminates noise and lighting problems without deteriorating the edges that will later serve to extract the lines that determine the features of the face to be portrayed. Finally, the resolution of the image is reduced so that the characteristic features of the images are defined with a smaller amount of spatial information, which also reduces the processing time.

Edge Determination. The lines that determine the features of the face to be drawn correspond to areas with a high luminosity gradient, that means that they are lines which separate areas of different texture or gray level. One of the most efficient methods to find these zones is the Canny Algorithm, which starts by applying a Gaussian filter, then calculates the luminosity gradient image and tracks the ridges of this image with two thresholds: a high threshold to consider a gradient pixel as part of the ridge and a low one to consider that the analyzed pixel is not a ridge (hysteresis). The parameters of this algorithm are the standard deviation of the Gaussian filter and the two thresholds of hysteresis.

Edge Image Cleaning. The obtained edge image thus has spurious edges or broken lines, so a cleaning process is required. Here a morphological closing filter is applied using four structuring elements (vertical, horizontal, left diagonal and right diagonal) which have the effect of eliminating small strokes and filling in long chopped strokes. It is worth mentioning that at this stage an image is generated for each of the strokes obtained after cleaning.

Small Lines Removal. Here, all images that contain lines with a smaller number of pixels than a certain threshold are eliminated. In addition, the n images with the n detected traces, they are gathered into a single image of lines, which allows a visual control of the results of the line obtaining stage.

2.3. Shadow Obtaining Stage

This second stage of the AVS consists of four steps applied to the image in gray levels obtained in the Preprocessing step of the Lines Extraction stage: Determination of shadow areas, Generation of hatched shadows images, Cleaning of the image of hatched shadows and Elimination of shadow areas smaller than a certain size. These four steps are briefly presented below.

Determination of shadow areas. The images obtained usually have shades of different gray values. In order to simplify the portrait, only two levels of shadow areas will be considered: one that goes from black to the threshold U_1 and a second zone that is between the thresholds U_2 and U_3, with $U_1 < U_2 < U_3$. In this way, two images are generated, one with blobs representing dark shadows. $[0, U_1]$ and another with blobs that represent subtle less dark shadows $[U_2, U_3]$.

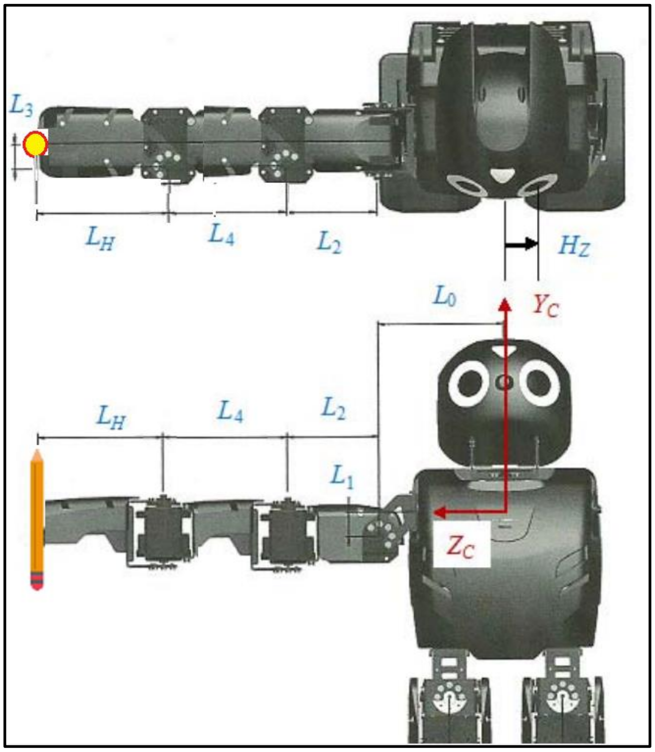


Figure 2. Kinematic of the modified DarwinOP arm.

Generation of hatched shadows images. To represent these two types of shadows it was decided to generate two types of hatching, which can be made of two different densities or the same density, but with directions at 90° to each other. Two images of the same dimension as the image of the person are then generated, one for each of the selected hatching types, then the intersection between the shadow image (dark and light) with the corresponding hatching image in each case is made.

Cleaning of the image of hatched shadows. In the same way as in the previous stage, and for the same reasons, the images of hatched blobs are cleaned using the same morphological filters used previously with equivalent results. As in the previous stage, here also an image is generated for each detected blob.

Elimination of shadow areas smaller than a certain size. Here all images containing blobs with a smaller number of pixels than a certain threshold are eliminated. In addition, the m images with the m blobs detected are gathered into a single blob, which allows a visual control of the results of the stage: obtaining shadow areas.

At this point, the three obtained images are brought together: one with lines/borders, one with dark shadows and one with light shadows to have a visual control of the results. To generate the robot movement instructions, it seems to be more efficient to use the n images of lines with the m images of blobs.

3. Kinematics Phase

The Artificial Vision phase delivers the information related to the lines to be drawn, where each line is a sequence of pixels, that is, of coordinates (x,y) in the drawing paper plane. Thus, for the robot to be able to make these strokes, it is necessary to transform that sequence of Cartesian coordinates (x,y) expressed in the referential of the paper into a sequence of joint coordinates $(\theta_1, \theta_2, \theta_3, \theta_4)$, i.e. the four joint variables of the robot arm mentioned in Section 1.2. This transformation is carried out by means of the Inverse Kinematics of the robot arm, to obtain it, it is first necessary to have the equations corresponding to its Direct Kinematics.

Table 1. Arm link dimensions for modified DarwinOP.

Length	Value (mm)
L_0	
L_1	16.0
L_2	60.0
L_3	16.0
L_4	60.0
L_H	100.0

3.1. Robotic Arm Direct Kinematics

As commented in Section 1.2, an extra DoF was added to the arm of the DarwinOP with which the portrait will be drawn: wrist flexion-extension (pitch), as shown in Figure 2. In the Table 1 are shown the dimensions of the links of the corresponding kinematic chain and, in Table 2, the articular limits of this kinematic chain are shown.

Table 2. DarwinOP Joint Limits

Joint i	Joint name	Axis	$\theta_{i\text{MIN}}$	$\theta_{i\text{MAX}}$
1	Shoulder pitch	Z_1	–250	250
2	Shoulder roll	Z_2	–100	100
3	Elbow	Z_3	0	160
4	Wrist pitch	Z_4	0	160

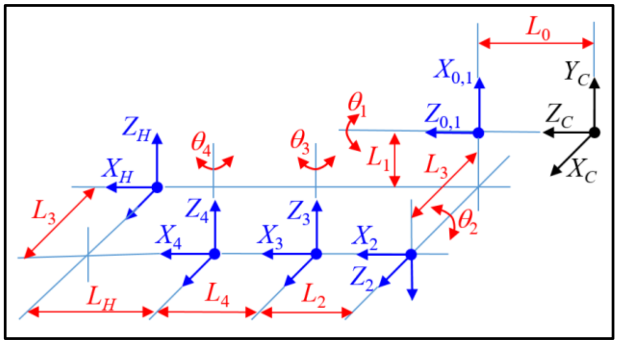


Figure 3. Denavit-Hartenberg diagram of the DarwinOP arm.

Table 3. D-H parameters of modified DarwinOP arm

i	α_{i-1}	a_{i-1}	d_i	θ_i
1	0°	0	0	θ_1
2	–90°	L_1	L_3	θ_2–90°
3	90°	L_2	0	θ_3
4	0°	L_4	0	θ_4

To obtain the Direct Kinematics of the arm, the classic D-H Methodology (Denavit-Hartenberg) [1] will be used. The steps are: *i)* Assign reference frames to each joint, as shown in Figure 3; *ii)* Make a table with the four parameters D-H (length and twist of the link plus the distance and angle between successive links) of each joint, shown in Table 3; *iii)* Obtain the four $^{i-1}{}_{i}T$ matrices corresponding to the four rows of Table 3; and multiply them to obtain the homogeneous transformation $^{4}{}_{0}T$ that represents the Direct Kinematics of the DarwinOP arm: ${}_{0}^{4}T = \prod_{i=1}^{i=4} {}_{i}^{i-1}T$, where $^{i-1}{}_{i}T$ matrices are done by:

$$\left[{}_{i}^{i-1}T\right] = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & -d_i s\alpha_{i-1} \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & d_i c\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (1)$$

The classical form of the generalized step matrix $^{i-1}{}_{i}T$, given by (eq. 1), is applied to the four rows of Table 3, obtaining the following four matrices:

$$[{}_{1}^{0}T] = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad [{}_{2}^{1}T] = \begin{bmatrix} c_2 & -s_2 & 0 & L_1 \\ 0 & 0 & 1 & L_3 \\ -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$[{}_{3}^{2}T] = \begin{bmatrix} c_3 & -s_3 & 0 & L_2 \\ 0 & 0 & -1 & 0 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad [{}_{4}^{3}T] = \begin{bmatrix} c_4 & -s_4 & 0 & L_4 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Multiplying these four matrices the Homogeneous Transformation $^{4}{}_{0}T$ is obtained, which represents the Direct Kinematics of the arm:

$${}_{0}^{4}T = \prod_{i=1}^{i=4} {}_{i}^{i-1}T\,(\theta_i) \qquad (2)$$

This matrix is shown in a complete form in Addendum 1. Finally, using the transformation between the reference frame associated with the robot's chest and the frame of the arm base $^{C}{}_{0}T$, as well as the missing transformation of the drawing tool $^{4}{}_{H}T$, as showed by figure 4, the direct kinematics of the drawing tool expressed in the reference frame of the humanoid robot located on his chest will be:

$$[{}_{H}^{C}T] = [{}_{0}^{C}T][{}_{4}^{0}T(\theta_1, \theta_2, \theta_3, \theta_4)][{}_{H}^{4}T] \qquad (3)$$

Where:

$$\left[{}_{0}^{C}T\right] = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & L_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \left[{}_{H}^{3}T\right] = \begin{bmatrix} 1 & 0 & 0 & L_H \\ 0 & 1 & 0 & -L_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The direct kinematics of the DarwinOP arm with four joints is given by the matrix shown in Addendum 2.

3.2. Robotic Arm Inverse Kinematics

The arm with four joints has a configuration (θ_2=0, or rather θ_2=90) so that the three remaining joints ($\theta_1, \theta_3, \theta_4$) have parallel axes that represents a redundant manipulator whose inverse kinematics has an infinite number of solutions. This is because there are only two equations and three unknown variables. There are a large number of methods in the literature to solve this problem in a unique way; one of them consists in proposing a third equation, for example, in [2] a desired orientation θ_H is imposed on the final effector (third link). In our case it would be:

$$\theta_H = \theta_1 + \theta_3 + \theta_4$$

Another way to generate this third equation is to impose constraints on the joints. In this work we propose to obtain this equation by imposing that the line that joins the origin of this redundant arm with the drawing tool must always be above its first link, this restriction is known as arm with elbow down. Let us remember that the destination position of the drawing tool is known in Cartesian coordinates (X, Y); or in their equivalent polar coordinates: $\rho = \sqrt{X^2 + Y^2}$ and $\theta_\rho = \tan^{-1}(Y/X)$. The third coordinate is not considered because it is a constant value: Z=100 mm. As it is shown in the kinematic diagram corresponding to this redundant arm in figure 5. Thus, the restriction "*elbow down*" is materialized by imposing that $\theta_1 < \theta_\rho$. This is:

$$\theta_1 = \theta_\rho - \theta$$

where: $\theta = -\frac{\theta_{min}}{(L_1+L_2)+L_4+L_H}\rho + \theta_{min}$ and: $\theta_{min} = \pi$ (4)

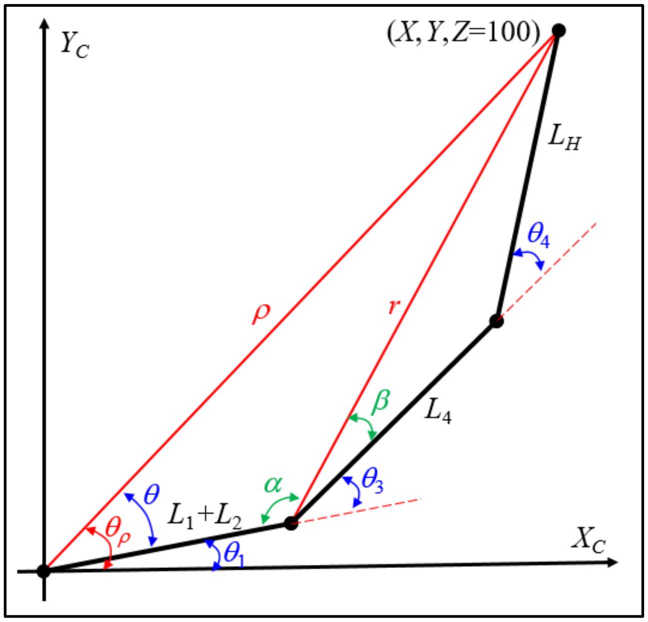


Figure 4. Scheme of the redundant arm to solve its Inverse Kinematics.

With this linear model of the angular offset θ between ρ and (L_1+L_2), it is obtained so that $\rho=L_1+L_2+L_4+L_H$ when the offset angle is zero, meanwhile, when ρ becomes very small, the angle θ approaches the value of θ_{min} that we have set at 180°. To calculate θ_3 we proceed to solve for the angles α and β shown in the diagram of figure 5 using the Law of Cosines, obtaining:

$$\cos\alpha = \frac{(L_1+L_2)^2+r^2-\rho^2}{2(L_1+L_2)r} \quad \text{and} \quad \cos\beta = \frac{(L_1+L_2)^2+r^2-L_4{}^2}{2l_4 r}$$

where: $r = \sqrt{(X-(L_1+L_2)\cos\theta_1)^2 + (Y-(L_1+L_2)\sin\theta_1)^2}$

With which the second joint variable θ_3 can now be calculated by:

$$\theta_3 = \pi - \alpha^+ - \beta^+ \tag{5}$$

Finally, the third joint variable will be calculated as follows:

$$\theta_4 = \tan^{-1}\left(\frac{Y-(L_1+L_2)\sin\theta_1 - L_4\sin(\theta_1+\theta_3)}{X-(L_1+L_2)\cos\theta_1 - L_4\cos(\theta_1+\theta_3)}\right) - \theta_1 - \theta_3 \tag{6}$$

The Equations 4, 5 and 6 constitute the Inverse Kinematics of the DarwinOP humanoid arm with four joints, of which only three of them will be used for the drawing task described in this article.

4. Results

Once the direct kinematics (Addendum 1) as well as the inverse kinematics (eq. 4, 5 and 6) have been calculated, it can be found the value of the joint coordinates (θ_1, θ_2=90, θ_3, θ_4) with which the drawing tool can be positioned in a Cartesian coordinates point (X, Y, Z) expressed in the humanoid frame (R_C) located on his chest. However, the coordinates of the lines provided by the AVS (x, y, z=0) are referenced to the paper frame (R_P) located in its upper left corner as indicated in figure 5. Therefore, it is necessary to find a transformation of the Cartesian coordinates on the paper frame to Cartesian coordinates in the humanoid frame.

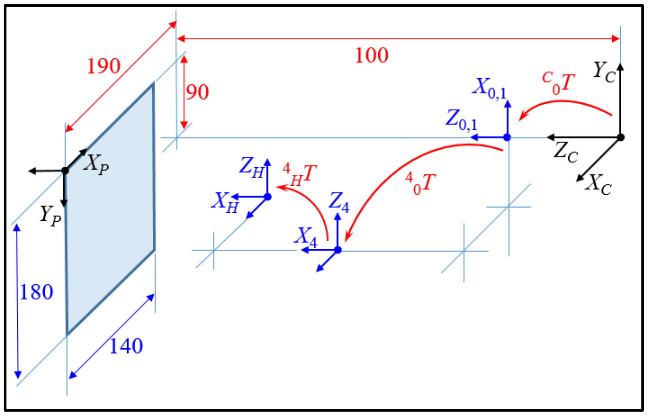


Figure 5. Placement of the drawing paper.

As mentioned in Section 1.2 and taking into account the modified architecture of the DarwinOP arm, the largest paper you can draw on is 140 mm wide by 180 mm height (almost half a letter size paper), as long as the paper is parallel to the sagittal plane of the robot and at a distance of L_P=100 mm to the right of such plane, the robot will be able to draw on it with a pencil mounted in its right hand. In this way, the origin of the paper's framework (R_P) is at the $(190, 90, 100)$ mm with respect to the robot's chest frame (R_C), while its axes have been rotated 180° around the Z_C axis (figure 4).

The homogeneous transformation that allows obtaining the Cartesian coordinates (X, Y, Z) in the robot's chest frame (R_C) from the coordinates $(x, y, 0)$ in the paper frame(R_P) is then, the following:

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 190 \\ 0 & -1 & 0 & 90 \\ 0 & 0 & 1 & 100 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 0 \\ 1 \end{bmatrix} \tag{9}$$

$$\begin{aligned} X &= 190 - x \\ Y &= 90 - y \\ Z &= 100 \end{aligned} \tag{10}$$

Let us recall that the coordinates $(x, y, 0)$ in the paper frame (R_P) have been obtained in the Artificial Vision phase, while the Cartesian coordinates (X, Y, Z) in the robot frame (R_C) must be transformed to joint coordinates of the humanoid arm through the Inverse Kinematics given in equations. 4, 5 and 6.

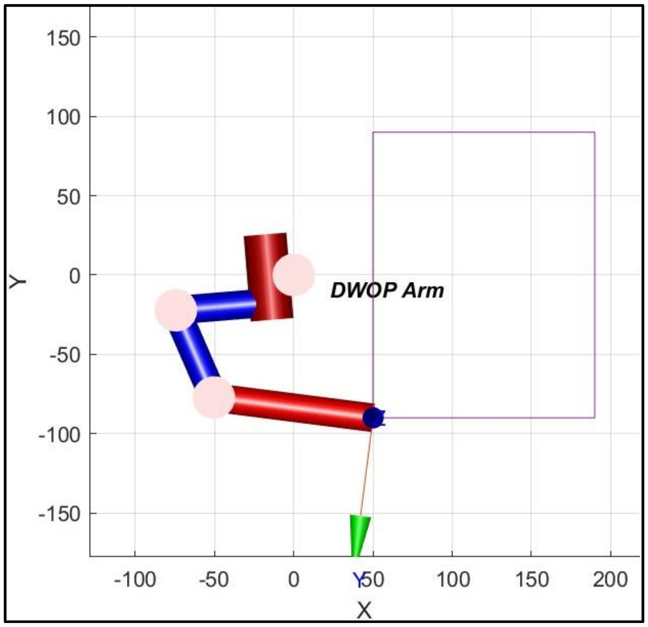


Figure 6. Simulation results: drawing a 140 × 180 mm rectangle in the paper surface.

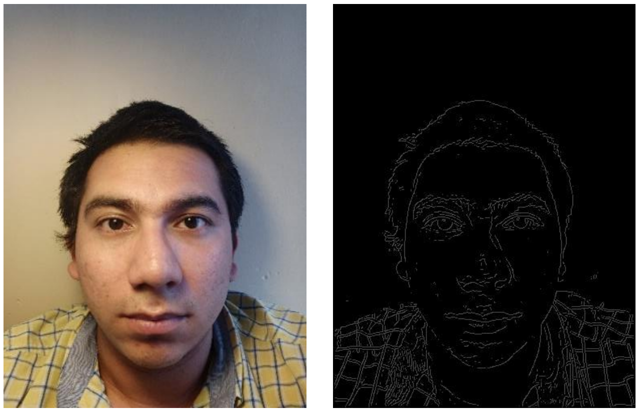


Figure 7. *a)* Original RGB image; *b)* Canny edges image.


Figure 8. *a)* Dark shadows, *b)* Light shadows


Figure 9. Cleaned images: *a)* Canny, *b)* Hatched shadows.

4.1. Workstation and calibration

As it has been mentioned before, the DarwinOP humanoid robot will be seated in a chair in which there are no slides in any direction so that the robot frame remains fixed within the workstation. The drawing plane is located at L_P=100 mm to the right of the sagittal plane of the robot and parallel to this plane. Then, the origin of the drawing paper frame (R_P) is located at $(190, 90, 100)$ mm in the robot frame (R_C), while the center of the drawing plane will be at the same height as the robot frame origin.

To verify that the drawing plane is in the correct location in terms of position and orientation, a stage of what we could call calibration is carried out. This stage consists of bringing the drawing tool to the positions corresponding to the four corners of the drawing paper, i.e. $(50, 90, 100)$mm, $(50, -90, 100)$ mm, $(190, 90, 100)$ mm, $(190, -90, 100)$ mm, and verify that they match perfectly with the paper, otherwise make the appropriate adjustments in the frame of the paper to guarantee a good performance of the drawing robot.

4.2. Artificial Vision System

Figures 7 to 10 show the results obtained through the application of the PDI algorithms mentioned above (Sections 2.2 and 2.3), using OpenCV [4], until reaching a reduced resolution image with the lines to be drawn, specified as sequences of Cartesian coordinates $(x, y, 0)$ in the paper frame (R_P).

4.3. Kinematics

In a first approach, both the direct and inverse kinematics were programmed in Matlab using the Peter Corke's Robotics Toolbox [3]. Thus, a redundant arm was built with 4 joints, three of them with parallel axes (only when θ_2=90), which obeys the direct and inverse kinematic models obtained in Sections 3.1 and 3.2. To test this model different movements, corresponding to possible lines of a portrait, were made verifying that all of them lay in the drawing plane. In particular, it was simulated the draw of a rectangle of 140×180 mm in the paper surface, as shown in figure 6.

5. Final comments

This article shows the results obtained in the first two of the three phases of the development project of a portrait robot using the DarwinOP humanoid, namely: The Artificial Vision phase that obtains the set of minimum lines that define the face and its main shadows and the Kinematic phase, which is in charge of translating these defined lines in the image plane to movements of the arm on the drawing paper plane. The simulation results of what will be the third and last phase of this project are also shown: its implementation in the humanoid DarwinOP.

As regards the Artificial Vision phase, a sequence of algorithms for image processing and feature extraction was developed using OpenCV that allowed obtaining the set of minimum lines at an appropriate resolution that represent the main features of the model's face to portray. For its part, a simple methodology was proposed to solve the problem of the inverse kinematics of a redundant manipulator with three joints with parallel axes, having validated these models through Matlab simulations using Peter Corke's Robotics Toolbox. It is important to point out that we don't know applications of this kind for small size commercial humanoids like DarwinOP.

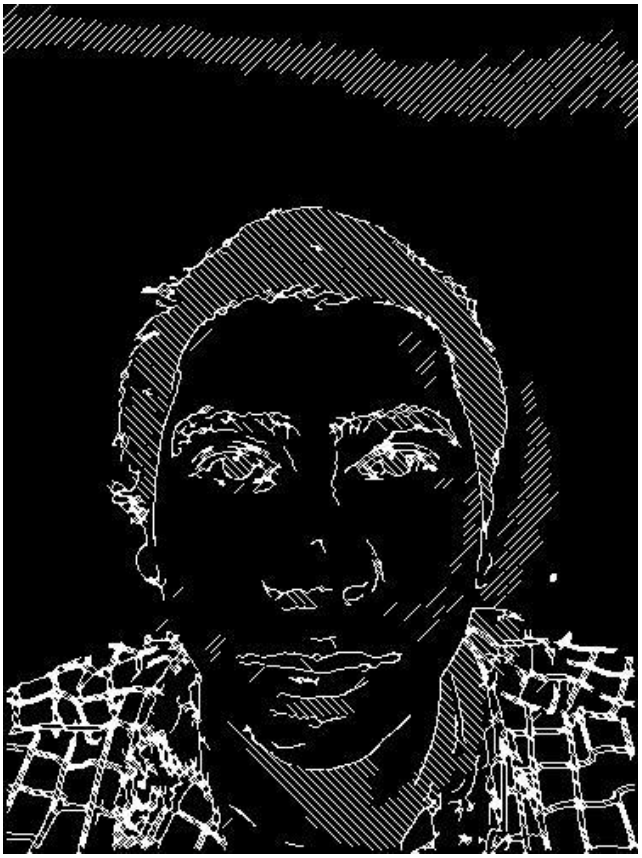


Figure 10. Final image: lines to be draw.

5.1. Future Work

In order to finish the project, two important tasks must be carried out: *i)* Build the workstation formed by the DarwinOP chair and the drawing "easel", which should have adjustments in all three axes and, perhaps a tilt adjustment; *ii)* Develop the terminal organ of the arm, including the wrist flexion-extension joint (wrist pitch), as well as the holding system of the drawing tool *iii)* Transfer all the system programming to the on-board computer of the humanoid of Artificial Vision using OpenCV (like in the actual prototype), as well as the kinematic programming system; and, finally, *iv)* Program the arm movement control system so that it can make the traces on the paper.

References

[1] J. Denavit and R.S. Hartenberg, 1955, A Kinematic Notation for Lower-Pair Mechanisms Based on Matrices, Journal of Applied Mechanics: 215-221.

[2] Shahnaz Habibkhah, René V. Mayorga. The Computation of the Inverse Kinematics of a 3dof Redundant Manipulator via an ANN Approach and a Virtual Function. ICINCO 2020. 17th Int. Conf. On Informatics and Control, Automation and Robotics. pp 471-477.

[3] P.I. Corke, "Robotics, Vision & Control", Springer 2017, ISBN 978-3-319-54413-7.

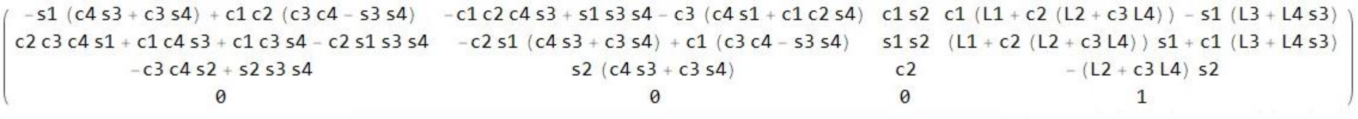
[4] Gary Bradski, Adrian Kaehler. Learning Opencv: Computer Vision with the Opencv Library. O'Reilly 2008.

[5] E. J. Galindo Ramírez. Sistema mecatrónico de planeación de trayectorias para robot manipuladores por medio de visión artificial. UPIITA-IPN, 2007.

[6] R. Flores Osante, R. Calderón Duana. Prototipo de investigación en generación de trayectorias y orientación de objetos en 2D por visión artificial. UPIITA-IPN, 2009.

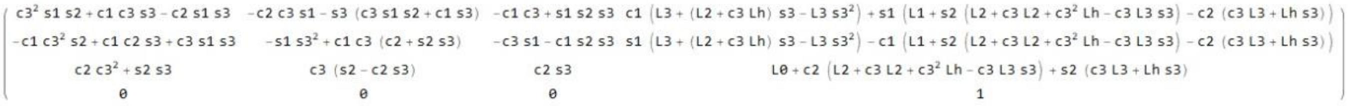
[7] J. Gonzáles Godoy. Control mediante visión de un robot humanoide. CIC-IPN, 2013.

Addendum 1. $[{}^{4}_{0}T] = \prod_{i=1}^{i=4} {}^{i-1}_{i}T\ (\theta_i)$ in a complete form:



$$\begin{pmatrix} -s1\ (c4\ s3 + c3\ s4) + c1\ c2\ (c3\ c4 - s3\ s4) & -c1\ c2\ c4\ s3 + s1\ s3\ s4 - c3\ (c4\ s1 + c1\ c2\ s4) & c1\ s2 & c1\ (L1 + c2\ (L2 + c3\ L4)) - s1\ (L3 + L4\ s3) \\ c2\ c3\ c4\ s1 + c1\ c4\ s3 + c1\ c3\ s4 - c2\ s1\ s3\ s4 & -c2\ s1\ (c4\ s3 + c3\ s4) + c1\ (c3\ c4 - s3\ s4) & s1\ s2 & (L1 + c2\ (L2 + c3\ L4))\ s1 + c1\ (L3 + L4\ s3) \\ -c3\ c4\ s2 + s2\ s3\ s4 & s2\ (c4\ s3 + c3\ s4) & c2 & -(L2 + c3\ L4)\ s2 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Addendum 2. $[{}^{C}_{H}T] = [{}^{C}_{0}T][{}^{0}_{4}T(\theta_1, \theta_2, \theta_3, \theta_4)][{}^{4}_{H}T]$ in a complete form:



$$\begin{pmatrix} c3^2\ s1\ s2 + c1\ c3\ s3 - c2\ s1\ s3 & -c2\ c3\ s1 - s3\ (c3\ s1\ s2 + c1\ s3) & -c1\ c3 + s1\ s2\ s3 & c1\ \left(L3 + (L2 + c3\ Lh)\ s3 - L3\ s3^2\right) + s1\ \left(L1 + s2\ \left(L2 + c3\ L2 + c3^2\ Lh - c3\ L3\ s3\right) - c2\ (c3\ L3 + Lh\ s3)\right) \\ -c1\ c3^2\ s2 + c1\ c2\ s3 + c3\ s1\ s3 & -s1\ s3^2 + c1\ c3\ (c2 + s2\ s3) & -c3\ s1 - c1\ s2\ s3 & s1\ \left(L3 + (L2 + c3\ Lh)\ s3 - L3\ s3^2\right) - c1\ \left(L1 + s2\ \left(L2 + c3\ L2 + c3^2\ Lh - c3\ L3\ s3\right) - c2\ (c3\ L3 + Lh\ s3)\right) \\ c2\ c3^2 + s2\ s3 & c3\ (s2 - c2\ s3) & c2\ s3 & L0 + c2\ \left(L2 + c3\ L2 + c3^2\ Lh - c3\ L3\ s3\right) + s2\ (c3\ L3 + Lh\ s3) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Capítulo 3.

Robot Manipulators

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Comparing cost functions for the optimal control of robotic manipulators using Pontryagin's Maximum Principle

Hedy César Ramírez-de-Ávila[1], Juan Antonio Rojas-Quintero[2], Sergio Morales-López[1] and Eusebio Bugarin[1]

***Abstract*—In the framework of Pontryagin's Maximum Principle, the choice of the Hamiltonian has an impact on the resulting controls. In this framework, the cost function composes the Hamiltonian which leads to a desired control and motion behavior when appropriately chosen. In order to optimally control motion of robotic manipulators we compare the numerical impact of two cost functions on the resulting controls and positions of the optimal trajectory. The selected cost functions focus on the controls so that minimum effort is achieved during motion. We compared a cost function that is typically found in the literature with one that involves the robot mass tensor components. We applied these to the optimal control simulation of two robotic manipulators. Numerical trials showed that the mass tensor acts as a stabilizing factor that leads to lower motion amplitudes, increased numerical stability and reduced CPU computing times. Our proposed cost function may therefore be beneficial for the optimal path planning and control of robotic manipulators.**

***Index Terms*—Optimal control; Cost function; Trajectory optimization; Manipulators; Robot motion.**

I. INTRODUCTION

One of the traditional optimal control approaches involves the optimization in the neighborhood of the optimal trajectory which leads to the application of Pontryagin's Maximum Principle (PMP). This requires solving a coupled Hamiltonian system of ordinary differential equations (ODE) governing state and conjugate variables [1]. We are interested in developing an optimal control methodology based in PMP, where the goal is to minimize an integral functional which is regarded as a performance index (PI) (also commonly called objective function) [2], [3]. The PI functional is the integral of an appropriately chosen cost function which then composes a Hamiltonian function that describes the system. The application of the PMP to this optimal Hamiltonian leads to a system of ODE for the state and co-state variables. Therefore, for the control of articulated robots, the choice of the Hamiltonian is a critical part of this process.

In this paper, we study the numerical impact of two torque criterion cost functions on the ODE solution that characterize the optimal control of robotic serial manipulators using PMP. We take a cost function that is commonly used in the literature (as in [4]) and compare it with its invariant form that we propose as a better alternative. We apply these cost functions to the case of a 1 degree of freedom (DOF) robotic manipulator in order to illustrate how our proposed cost function affects the structure of optimality conditions. Numerical simulations are then also carried out on a 2 DOF robotic manipulator to demonstrate how our proposed cost function improves the numerical stability of the methodology, and additionally reduces the required computing time. Note that the proposed cost function has been presented in [5]–[7] but has not been directly compared against other cost functions. Therefore, this research contributes in establishing the benefits of using the proposed invariant cost function for the optimal control of robots, by comparing its impact on an ODE solver against a similar, non-invariant cost function.

II. ROBOT DYNAMICS

To derive the dynamic model, we follow standard procedures that can be found in [8]. The robotic manipulators that we study are of the serial type in which one actuator is placed between two consecutive links, and induces revolute motion. The robotic manipulator is governed by second order nonlinear ordinary differential equations of motion of the following form:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{V}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \mathbf{u}, \tag{1}$$

in which $\mathbf{M}(\mathbf{q})$ is the system mass tensor which is a positive-definite symmetric bilinear form dependent on configurations $\mathbf{q}$. For the 1 DOF manipulator, it is a single scalar component, but for the 2 DOF manipulator, it is of rank 2. $\mathbf{V}(\mathbf{q}, \dot{\mathbf{q}})$ is the vector containing centrifugal and Coriolis effects. $\mathbf{G}(\mathbf{q})$ represents Earth's gravity action on the system and $\mathbf{u}$ is the torque vector necessary to cause the desired motion.

A. Dynamics of a 1 DOF manipulator robot

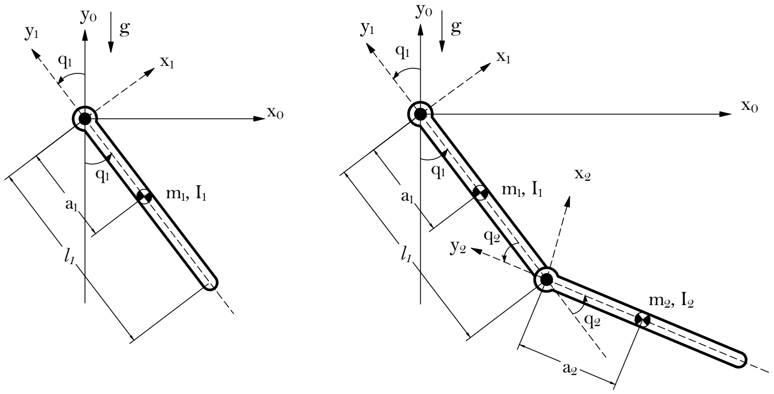
A kinematics scheme of this 1-DOF manipulator robot is shown in Figure 1(a) where a nomenclature convention is indicated in the caption. For this manipulator, we set the following values: $m_1 = 2.4\,\mathrm{kg}$; $a_1 = 0.4\,\mathrm{m}$; $I_1 = 1\,\mathrm{kg\,m^2}$. For this case, $M = m_1{a_1}^2 + I_1 = 1.384\,\mathrm{kg\,m^2}$ is the sole mass tensor component (which is constant). For this 1 DOF robotic manipulator, motion equation (1) simplifies in

$$u_1 = M\ddot{q}_1 + m_1 a_1 g \sin(q_1).$$

This work was financially supported by Secretaría de Educación Pública (SEP) and Consejo Nacional de Ciencia y Tecnología (CONACYT), under Grant A1-S-29824; and Tecnológico Nacional de México (TecNM) projects.

[1] Hedy César Ramírez-de-Ávila, Sergio Morales-López and Eusebio Bugarin are with Tecnológico Nacional de México/I.T. Ensenada, Ensenada, Baja California, Mexico. `hramirez@ite.edu.mx`

[2] Juan Antonio Rojas-Quintero is with CONACYT/Tecnológico Nacional de México/I.T. Ensenada, Ensenada, Baja California, Mexico. `jarojas@conacyt.mx`



(a) 1 DOF robotic manipulator (b) 2 DOF robotic manipulator

Fig. 1. Kinematics scheme of the used robotic manipulators: (a) 1-DOF robotic manipulator; (b) 2-DOF robotic manipulator. Parameters $(m_i, I_i, a_i, l_i, q_i)$ respectively denote the i^{th}: link mass; link moment of inertia; distance from center of twist to center of mass; link length; configuration parameter.

B. Dynamics of a 2 DOF manipulator robot

A kinematics scheme of the targeted 2 DOF robotic manipulator is shown in Figure 1(b). The parameters for this robot are reported in [9]. For this manipulator, (1) leads to two equations of motion, u_1 and u_2, that describe each joint. The composing terms of each of these motion equations are given by

$$\begin{aligned}
M_{11} &= I_1 + I_2 + m_1 {a_1}^2 + m_2({a_2}^2 + {l_1}^2 + 2 l_1 a_2 \cos(q_2));\\
M_{12} &= I_2 + m_2 {a_2}^2 + m_2 l_1 a_2 \cos(q_2);\\
M_{21} &= I_2 + m_2 {a_2}^2 + m_2 l_1 a_2 \cos(q_2);\\
M_{22} &= I_2 + m_2 {a_2}^2;\\
V_1 &= -m_2 l_1 a_2 \sin(q_2)(2\dot{q_1}\dot{q_2} + \dot{q_2}^2);\\
V_2 &= m_2 l_1 a_2 \sin(q_2)\dot{q_1}^2;\\
G_1 &= g(m_1 a_1 + m_2 l_1)\sin(q_1) + m_2 g a_2 \sin(q_1 + q_2);\\
G_2 &= m_2 g a_2 \sin(q_1 + q_2).
\end{aligned}$$

III. OPTIMAL CONTROL METHODOLOGY

A. Performance index and cost function

We wish to bring the robotic articulated system from a state of initial positions and velocities $(q(0), \dot{q}(0))$ to a final state $(q(T), \dot{q}(T))$ in a prescribed time T. The required torque intensities to accomplish this, can be constrained by minimizing an integral functional of the type

$$J(\mathbf{u}) = \int_0^T \gamma(\mathbf{u})\mathrm{d}t, \tag{2}$$

where $\mathbf{u}$ is the robot torques vector that we regard as the controls of our system. Integral functional (2) is regarded as the PI, where γ is the selected cost function. This cost function needs to be selected in such a way that (2) can be effectively minimized. Therefore, it is usually chosen to be convex. In our case, it will depend on the robot torques $\mathbf{u}$.

We have noticed that when torque intensities minimization is sought, γ is usually chosen as:

$$\gamma_A = \frac{1}{2}\mathbf{u^T A u}, \tag{3}$$

where $\mathbf{A}$ is a diagonal, constant coefficient weighting matrix [10]–[12], often taken as the identity [4], [13]–[16]. We will be conducting simulations using this cost function (3). Without loss of generality, we will focus in the case where $\mathbf{A}$ is the identity.

Instead, as proposed in [5]–[7], we encourage the use of the following cost function

$$\gamma_M = \frac{1}{2}\mathbf{u^T M^{-1} u}. \tag{4}$$

It renders $J(\mathbf{u})$ invariant under a change of coordinates [5]–[7]. We will be carrying out simulations with this cost function in order to quantify the performance improvement over the case where γ_A is used.

B. Pontryagin's Maximum Principle

One of the traditional approaches to determine the numerical solution of optimal control problems is an indirect approach based on PMP in which optimality conditions, in the form of adjoint equations, are continuous [3]. We will determine the solutions to the optimal control problem of the robotic manipulators in this framework.

Let us define the optimal Hamiltonian function as:

$$H(\mathbf{u}, \mathbf{x}, \lambda) = \lambda^T \dot{\mathbf{x}} - \gamma(\mathbf{u}), \tag{5}$$

where $\mathbf{x}$ is the vector of states; λ is the vector of co-states (also called adjoint states or conjugate variables); and γ corresponds to the selected cost function.

The states are continuous quantities defined as

$$\mathbf{x} = (\mathbf{q} \;\; \zeta)^T = (\mathbf{q} \;\; \dot{\mathbf{q}})^T,$$

which can be used to establish motion equation (1) in state space form as

$$\dot{\mathbf{x}} = \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\zeta} \end{bmatrix} = \begin{bmatrix} \zeta \\ \mathbf{M}(\mathbf{q})^{-1}[\mathbf{u} - \mathbf{G}(\mathbf{q}) - \mathbf{V}(\mathbf{q}, \zeta)] \end{bmatrix}.$$

The co-states or adjoint variables are continuous quantities with continuous derivatives defined as

$$\lambda = (\mathbf{p} \;\; \xi)^T.$$

PMP provides us with the necessary conditions to obtain an optimal control law in the form of first order ODE that are nonlinear for the case of the control of robotic manipulators. Optimal controls are such that u_i maximize the Hamiltonian (5) as

$$\frac{\partial H}{\partial u_i} = 0, \quad \text{for } i = 1, .., n,$$

where n denotes the system DOF.

In state space, the robot motion equations can be expressed in symplectic form as first order nonlinear ODE conditions

$$\dot{q}_i(t) = \frac{\partial H}{\partial p_i}, \quad \dot{\zeta}_i(t) = \frac{\partial H}{\partial \xi_i}, \quad \text{for } i = 1, .., n. \tag{6}$$

The co-states (or adjoint equations) are in turn given also in symplectic form by

$$\dot{p}_i(t) = -\frac{\partial H}{\partial q_i}, \quad \dot{\xi}_i(t) = -\frac{\partial H}{\partial \zeta_i}, \quad \text{for } i = 1, .., n. \tag{7}$$

Optimality conditions under the selected criterion are established by (7). In order to obtain the required optimal controls and states that minimize (2), the set of $4n$ nonlinear ODE (6) and (7), requires a solution.

C. Cost function impact on optimality conditions

Let us focus on the case of a 1 DOF robotic manipulator in order to illustrate the impact of our cost function on optimality conditions. In this case, the mass tensor has a single constant component. Therefore, γ_M leads to exactly the same system of 4 ODE that describe the optimal controls and states as γ_A.

However, when γ_A is taken, then Pontryagin's optimality condition, i.e., the Hamiltonian is maximized when

$$u_1 = \frac{\xi_1}{M}. \tag{8}$$

Conversely, when γ_M is taken, then Pontryagin's optimality condition is met when

$$u_1 = \xi_1.$$

This means that when the inverse mass tensor components are involved in the cost function of (2), the second co-state (or adjoint parameter), directly describes the system torques, as advocated in [5]. This is significant because when γ_M is used, there is no need to reconstruct the control variable after computation.

It is important to remark that if γ_A (3) is used for the optimal control of a n DOF robotic manipulator, the expression of torques inevitably become much more complicated than (8). On the contrary, whenever γ_M (4) is used for the optimal control of a n DOF robotic manipulator,

$$u_i = \xi_i, \quad \text{for } i = 1, ..., n \tag{9}$$

(see Proposition 4 and corresponding proof in [5]).

IV. SIMULATION RESULTS

A. Evaluation methodology

In order to evaluate the impact of each cost function (4) and (3) on the optimal control of our 1 DOF and 2 DOF robotic manipulators, we have followed a specific methodology that articulates in the steps below:

1) Solve the system of $4n$ ODE (6) and (7) for growing values of T starting at 0.5 s with increments of 0.1 s, taking the same boundary values to determine each solution.
2) Compute the RMS torque for each trajectory as

$$u_{\text{RMS}} = \sqrt{\frac{\int_0^T \sum_{i=1}^n {u_i}^2}{T}}.$$

3) Compute the RMS power for each trajectory as

$$P_{\text{RMS}} = \sqrt{\frac{\int_0^T \sum_{i=1}^n (u_i \dot{q}_i)^2}{T}}.$$

4) Determine the CPU computing time for each trajectory.

Let us now mention some specifics for the above steps.

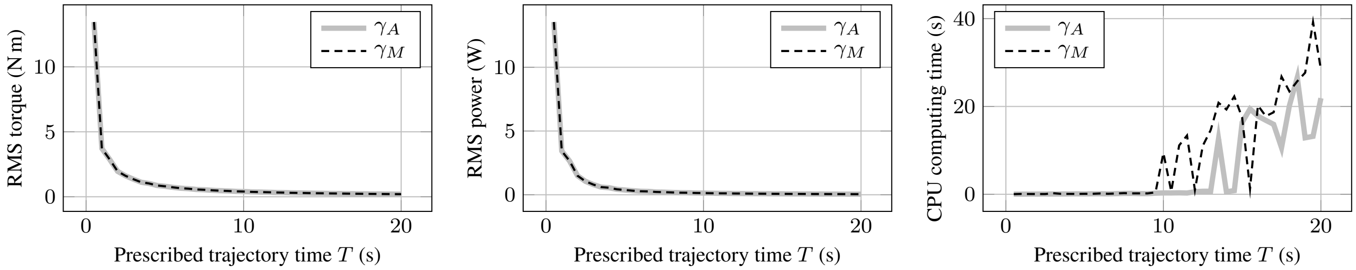
- Solutions were obtained with Mathematica's NDSolve ODE solver. This particular solver can be set up to operate with a wide range of solution methods. We have obtained best results by setting it up to operate with a Runge-Kutta time integration method (see [7]). Therefore, every solution was provided with this method.
- CPU computing time denotes the overall time it took our computer to obtain a solution. This was determined with Mathematica's AbsoluteTiming embedded function.
- All described solutions were computed with a 4-cores Intel® Core™ i7-8550U processor (1.8 GHz, 8 MB of cache memory).

B. Optimal control of a 1 DOF robotic manipulator

Figure 2 shows three sets of curves. Following step 1) of the above methodology, we obtained solutions to the set of 4 ODE composed of (7) and (6) with growing values of prescribed trajectory time up to $T = 20$ s. Solutions are obtained by fixing the following boundary values

$$\begin{cases} \mathbf{q}(0) = q(0) = 0\,\text{rad} \\ \dot{\mathbf{q}}(0) = \zeta(0) = 0\,\text{rad}\,\text{s}^{-1} \\ \mathbf{q}(T) = q(T) = 0.8\,\text{rad} \\ \dot{\mathbf{q}}(T) = \zeta(T) = 0\,\text{rad}\,\text{s}^{-1}. \end{cases} \tag{10}$$

Note that the solver stability was not affected by the change in cost function for this one-dimensional case. Trajectories RMS torque values were obtained by following steps 1) and 2) of the above methodology. These are shown by Figure 2(a) for both cost functions γ_A (3) and γ_M (4). For this one-dimensional case, there is no difference in the RMS torque values because the mass tensor has a unique constant component. Trajectories RMS power values were obtained by following steps 1) and 3) of the above methodology. These are shown by Figure 2(b) for both cost functions γ_A

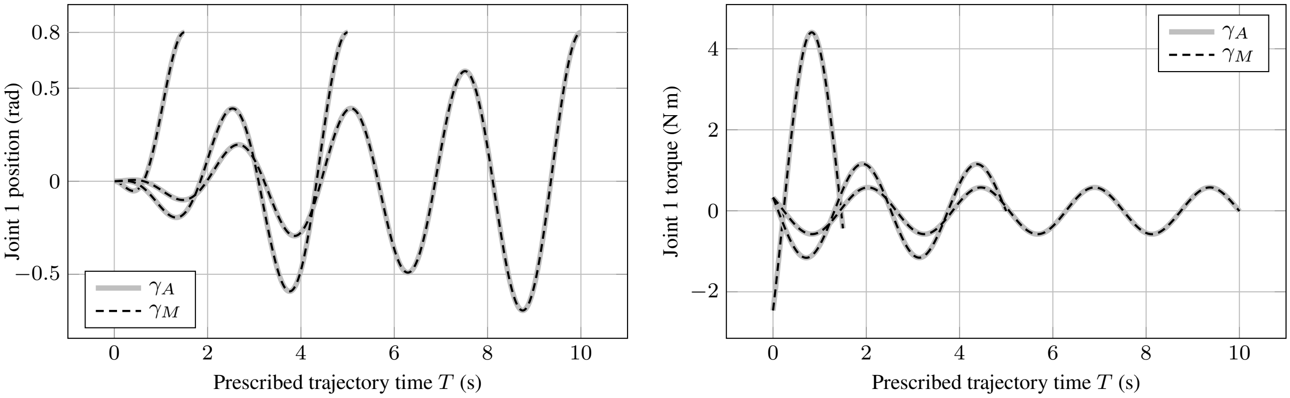


(a) RMS torque

(b) RMS power

(c) CPU computing time

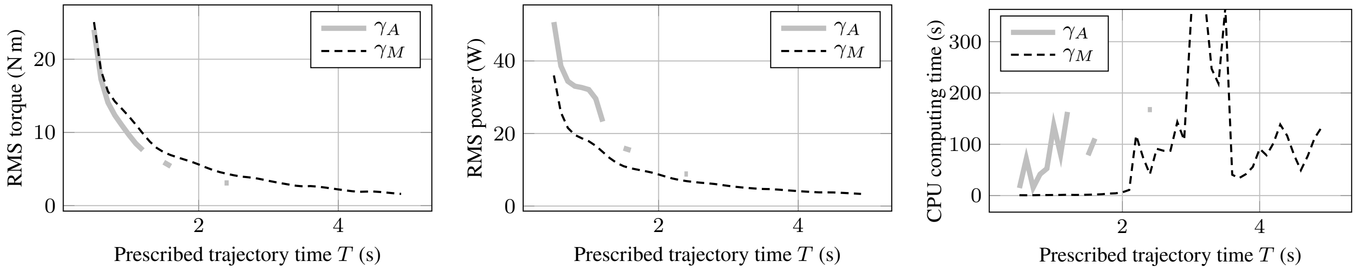
Fig. 2. Optimal control of a 1 DOF robotic manipulator. Evolution of RMS torque, RMS power and CPU computing time required to obtain a solution for growing values of prescribed time T. RMS results are equal with both cost functions. CPU computing time becomes larger after $T = 9\,\mathrm{s}$ with γ_M (4).



(a) Joint positions

(b) Joint torques

Fig. 3. Optimal control of a 1 DOF robotic manipulator. Evolution of joint positions and torques for three different values of prescribed time, $T = 1.5\,\mathrm{s}$, $T = 5\,\mathrm{s}$ and $T = 10\,\mathrm{s}$. For this manipulator, obtained joint positions and torques are equal with both cost functions γ_A (3) and γ_M (4).



(a) RMS torque

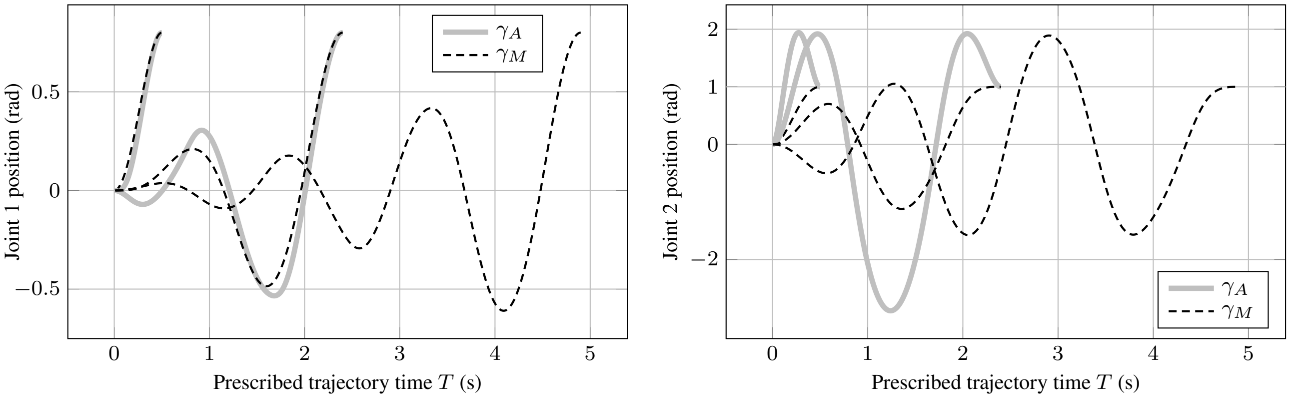
(b) RMS power

(c) CPU computing time

Fig. 4. Optimal control of a 2 DOF robotic manipulator. Evolution of RMS torque, RMS power and CPU computing time required to obtain a solution for growing values of prescribed time T. RMS results are lower with γ_A (3). CPU computing time is lower with γ_M (4). Note that gaps in the curves show that solutions cannot always be found when using γ_A (3). Solutions are obtained uninterruptedly when using γ_M (4) up to $T = 4.9\,\mathrm{s}$.

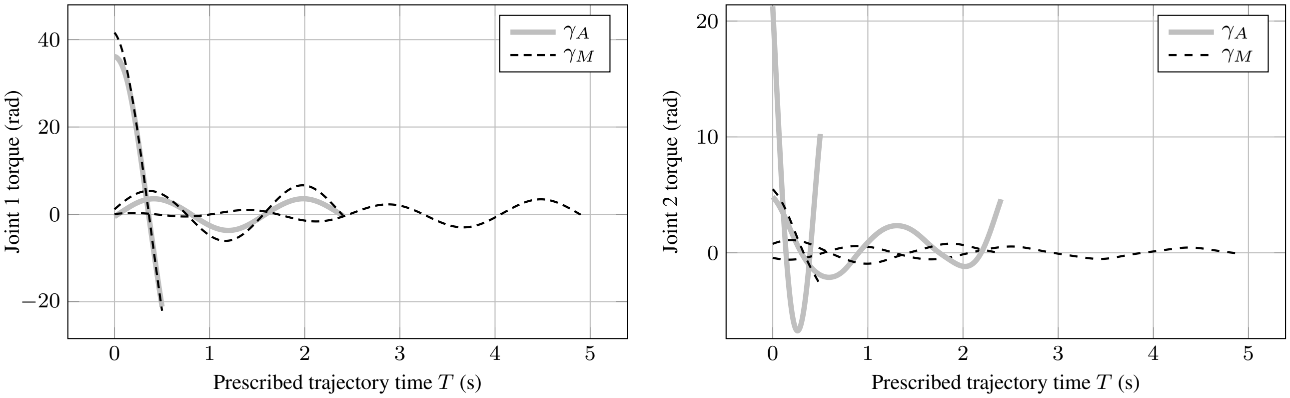
(3) and γ_M (4). As with RMS torques, obtained values are equal with both cost functions. CPU computing time for each calculated trajectory was obtained by following steps 1) and 4) of the above methodology. Figure 2(c) shows these times for growing values of prescribed trajectory time T. Note that values obtained with both cost functions are very similar up to $T = 9\,\mathrm{s}$. Afterwards, CPU times become larger when using γ_M (4). Therefore, γ_M (4) does not increase performance for this 1 DOF robotic manipulator because the sole mass tensor component is constant.

Next, consider Figure 3 which shows position and torque curves for the 1 DOF robotic manipulator during its optimal motion. Curves 3(a) show motions for $T = 1.5\,\mathrm{s}$, $T = 5\,\mathrm{s}$ and $T = 10\,\mathrm{s}$. We can appreciate the tendency as T increases: the robot will oscillate around its initial position of zero, approaching its final goal position with each swing. Note that for this one-dimensional case, a change in the cost function does not result in a different motion. This is in accordance with optimal control fundamentals which imply that a multiplying constant in the cost function does not change the minimizer [2]. Therefore, controls are also equal for both cost function cases as shown by Figure 3(b).



(a) Joint 1 positions

(b) Joint 2 positions



(c) Joint 1 torques

(d) Joint 2 torques

Fig. 5. Optimal control of a 2 DOF robotic manipulator. Evolution of joint positions and torques for three different values of prescribed time, $T = 0.5\,\mathrm{s}$, $T = 1.5\,\mathrm{s}$ and $T = 4.9\,\mathrm{s}$. For this manipulator, obtained joint positions and torques depend on the choice of the cost function.

C. Optimal control of a 2 DOF robotic manipulator

Let us now consider the case of a 2 DOF robotic manipulator. The results of the previous subsection confirmed that γ_M (4) does not result in a performance increase because the mass tensor has a unique constant component. However, this is not the case for any n DOF robot for which $n > 1$. In such cases, the mass tensor $\mathbf{M} = \mathbf{M}(\mathbf{q})$ is configuration dependent. Naturally, the mass tensor will not be simply a scaling factor for $n > 1$ DOF robots and therefore, results can be expected to change when using γ_M (4) as opposed to γ_A (3). For the present case, we took the following boundary values:

$$\begin{cases} \mathbf{q}(0) = \ (q_1(0), q_2(0)) \ = (0,0) \text{ rad} \\ \dot{\mathbf{q}}(0) = \ (\zeta_1(0), \zeta_2(0)) \ = (0,0) \text{ rad s}^{-1} \\ \mathbf{q}(T) = (q_1(T), q_2(T)) = (0.8, 1.0) \text{ rad} \\ \dot{\mathbf{q}}(T) = (\zeta_1(T), \zeta_2(T)) = (0,0) \text{ rad s}^{-1}. \end{cases} \tag{11}$$

Figure 4 shows three sets of curves. Following step 1) of the methodology in section IV-A, we obtained solutions to the set of 8 ODE composed of (6) and (7) with growing values of prescribed trajectory time T up to $T = 2.4\,\mathrm{s}$ when using γ_A (3), and up to $T = 4.9\,\mathrm{s}$ when using γ_M (4). Surprisingly, solutions that meet boundary values (11) could not be obtained for $T > 2.4\,\mathrm{s}$ in the case of γ_A (3). In fact, there are gaps in the corresponding curves of Figure 4 for the case of γ_A: solutions that effectively meet boundary values (11) could only be obtained inside the following intervals of T: [0.5 s,1.2 s], [1.5 s,1.6 s] and 2.4 s. This is caused by numerical issues such as stiffness, that commonly arise in optimal control [17]. On the contrary, solutions that meet the required boundary values could easily be obtained up to $T = 4.9\,\mathrm{s}$ when using γ_M (4), as shown by the curves of Figure 4. Therefore, for this 2 DOF robotic manipulator, solver stability was indeed affected by the change in cost function. Our proposed cost function γ_M (4) brings stability with respect to an increase in prescribed trajectory time T.

Trajectories RMS torque and RMS power values are respectively shown by Figures 4(a) and 4(b) for both cost functions. We can see that γ_A leads to lower torque consumption. However, γ_M leads to lower power consumption instead. Additionally, CPU computing time is significantly decreased by using γ_M for a fixed value of T, as shown by Figure 4(c). Table I displays how CPU computing time, required by the ODE solver to obtain a solution, is reduced with γ_M. This table indicates the ratio between the time t_{γ_A} it takes the solver to obtain a solution with γ_A, and the time t_{γ_M} taken with γ_M. Here, we have additionally obtained CPU computing times

TABLE I
DECREASE FACTOR IN CPU COMPUTING TIME REQUIRED BY NDSOLVE TO OBTAIN A SOLUTION: RATIO BETWEEN COMPARABLE COST FUNCTIONS USING TWO DIFFERENT PROCESSORS.

Ratio	$\frac{t_{\gamma_A}}{t_{\gamma_M}}$ i7-8550U	$\frac{t_{\gamma_A}}{t_{\gamma_M}}$ i9-9880H
Base value	4	5
Average	51	73

with a 2.3 GHz 8-Core Intel® Core™ i9-9880H processor in order to obtain alternative times. Table I values were rounded up to the closest integer and we can see how CPU computing times were reduced: with γ_M instead of γ_A, the solver finds a solution at least 4 times faster and 51 times faster in average with the i7-8550U processor (73 times faster in average with the i9-9880H processor).

Let us now evaluate the impact of cost functions γ_A and γ_M on the tendency of optimal positions and torques as prescribed trajectory time T increases. Figures 5(a) and 5(b) respectively show this tendency for the first and second joints of the robot, at fixed values of T: 0.5 s, 2.4 s and 4.9 s. As previously mentioned, a trajectory could not be obtained for $T = 4.9\,\mathrm{s}$ with γ_A (3). We can also see that for joint 2 (Figure 5(b)) motion amplitude is considerably reduced when using γ_M. In fact, we can see that with γ_M the tendency of both joint positions as, T increases, become very similar. Therefore, the mass tensor in γ_M acts as a stabilizing factor that ensures that both joints behave in a similar manner with motion amplitudes being reduced when compared to motions resulting from the use of γ_A. The same applies to robot torques as shown by Figures 5(c) and 5(d).

V. CONCLUSION

In the framework of Pontryagin's Maximum Principle, we have evaluated the numerical impact of two cost functions on an ODE solver. The first cost function γ_A (3) is commonly encountered in the literature. It has the benefit of causing lower torque consumption during optimal motion of robotic manipulators. However, when γ_A is used, governing optimal control equations exhibit stiffness to the point that solutions can only be found for a limited set of prescribed time T intervals. Additionally, motion amplitudes tend to be higher with this particular cost function. It should be possible to *weight* each joint performance with a similar cost function to γ_A by increasing or reducing the weight matrix values. However, determining these values is not systematic.

Conversely, our proposed cost function γ_M (4) involves the mass tensor components that act as a stabilizing factor. It provides a systematic way to obtain the following benefits:

- lower joint power consumption (see Figure 4(b));
- increased stability with respect to growing values of prescribed time T (see Figures 4 and 5);
- CPU computing times are significantly reduced (see Figure 4(c) and Table I);
- motion amplitudes are considerably diminished for every joint (see Figure 5).

These advantages apply for the case of n DOF robotic manipulators where $n > 1$. Further numerical trials will be held with other invariant cost functions in future research.

REFERENCES

[1] L. S. Pontryagin, V. G. Boltyanskii, R. V. Gamkrelidze, and E. F. Mishchenko, *The Mathematical Theory of Optimal Processes*, L. W. Neustadt, Ed. New York: Interscience Publishers (division of John Wiley & Sons, Inc.), 1962. ISBN 9780470693810

[2] M. Mesterton-Gibbons, *A primer on the calculus of variations and optimal control theory*. Providence, Rhode Island: American Mathematical Society, 2009, vol. 50. ISBN 9780821847725

[3] D. Liberzon, *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton, New Jersey: Princeton University Press, 2012. ISBN 9780691151878

[4] A. Eriksson, "Temporal finite elements for target control dynamics of mechanisms," *Computers & Structures*, vol. 85, no. 17, pp. 1399–1408, 2007. doi: 10.1016/j.compstruc.2006.08.080 Computational Structures Technology.

[5] F. Dubois, D. Fortuné, J. A. Rojas Quintero, and C. Vallée, "Pontryagin calculus in riemannian geometry," in *Geometric Science of Information*, F. Nielsen and F. Barbaresco, Eds. Cham: Springer International Publishing, 2015. ISBN 9783319250403 pp. 541–549.

[6] J. A. Rojas-Quintero, J. A. Rojas-Estrada, J. Villalobos-Chin, V. Santibañez, and E. Bugarin, "Optimal controller applied to robotic systems using covariant control equations," *International Journal of Control*, vol. 0, no. 0, pp. 1–14, 2021. doi: 10.1080/00207179.2020.1865570

[7] J. A. Rojas-Quintero, J. Villalobos-Chin, and V. Santibanez, "Optimal control of robotic systems using finite elements for time integration of covariant control equations," *IEEE Access*, vol. 9, pp. 104 980–105 001, 2021. doi: 10.1109/ACCESS.2021.3099131

[8] R. Kelly, V. Santibáñez Dávila, and J. A. Loría Peréz, *Control of Robot Manipulators in Joint Space*, ser. Advanced Textbooks in Control and Signal Processing. Springer-Verlag London, 2005. ISBN 9781852339944

[9] F. Reyes Cortés and R. Kelly, "Experimental evaluation of model-based controllers on a direct-drive robot arm," *Mechatronics*, vol. 11, pp. 267–282, 04 2001. doi: 10.1016/S0957-4158(00)00008-8

[10] M. Kaphle and A. Eriksson, "Optimality in forward dynamics simulations." *Journal of biomechanics*, vol. 41, no. 6, pp. 1213–21, 2008. doi: 10.1016/j.jbiomech.2008.01.021

[11] A. Eriksson and A. Nordmark, "Temporal finite element formulation of optimal control in mechanisms," *Computer Methods in Applied Mechanics and Engineering*, vol. 199, no. 25, pp. 1783–1792, 2010. doi: 10.1016/j.cma.2010.02.003

[12] M. Ghasemi, N. Kashiri, and M. Dardel, "Time-optimal trajectory planning of robot manipulators in point-to-point motion using an indirect method," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 226, pp. 473–484, 02 2012. doi: 10.1177/0954406211415321

[13] A. Nikoobin and M. Moradi, "Optimal balancing of robot manipulators in point-to-point motion," *Robotica*, vol. 29, pp. 233–244, 03 2011. doi: 10.1017/S0263574710000093

[14] C. Mirz, F. Scholer, J. P. Barreto, and B. Corves, "Optimal control based path planning for parallel kinematic manipulators utilising natural motion," in *2018 IEEE 14th International Conference on Automation Science and Engineering (CASE)*. IEEE, aug 2018. doi: 10.1109/coase.2018.8560547 pp. 223–228.

[15] M. Asgari and A. Nikoobin, "Analysis of optimal dynamic manipulation for robotic manipulator based on pontryagin's minimum principle," *Arabian Journal for Science and Engineering*, vol. 45, pp. 9159–9169, 2020. doi: 10.1007/s13369-020-04663-8

[16] M. R. Vezvari, A. Nikoobin, and A. Ghoddosian, "Zero-power balancing a two-link robot manipulator for a predefined point-to-point task," *Journal of Mechanical Science and Technology*, vol. 34, no. 6, pp. 2585–2595, 2020. doi: 10.1007/s12206-020-0533-5

[17] A. Grancharova and T. A. Johansen, *Survey of Explicit Approaches to Constrained Optimal Control*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 47–97. ISBN 9783540305606

Regulación de posición usando control PID con compensación de gravedad adaptable para robots manipuladores*

M. Moran-Armenta
Instituto Politécnico Nacional-CITEDI
Tijuana, Baja California, Mexico
mzextry@gmail.com

J. Montoya-Cháirez
Instituto Politécnico Nacional-CITEDI
Tijuana, Baja California, Mexico
jorge.montoya.ch@gmail.com.

J. Moreno-Valenzuela
Instituto Politécnico Nacional-CITEDI
Tijuana, Baja California, Mexico
moreno@citedi.mx.

R. Pérez-Alcocer
CONACYT-Instituto Politécnico Nacional-CITEDI
Tijuana, Baja California, Mexico
rrperez@citedi.mx

***Resumen*—En este documento, se propone un controlador PID con compensación de gravedad adaptable para regulación global de robots manipuladores con *n* gados de libertad. Por medio del método directo de Lyapunov, se concluye estabilidad asintótica del sistema de lazo cerrado de manera global. Para validar la teoría propuesta, se presentan simulaciones numéricas de un robot manipulador de 3 grados de libertad. Además, con propositos de demostrar el desempeño del controlador propuesto, se realiza una comparación con un controlador PID lineal previamente reportado en la literatura.**

***Index Terms*—Control PID, regulación, robot manipulador.**

I. INTRODUCCIÓN

Para poder diseñar un controlador, es necesario analizar el sistema al que se le desea aplicar. El controlador, es una señal que se calcula mediante la información del modelo que describe al sistema. El desempeño puede verse afectado por incertidumbres paramétricas o por perturbaciones externas. Debido a esto, en los últimos años se han desarrollado controladores robustos capaces de lidiar con dichas perturbaciones e incertidumbres [1]. Algunos ejemplos de sistemas a controlar son: vehículos espaciales, sistemas robóticos, procesos modernos de fabricación que requieran el control de temperatura, presión, humedad, flujo, etc. [2].

Los robots manipuladores tienen dinámicas no lineales complejas que pueden dificultar el control preciso y robusto [3]. Afortunadamente, los robots manipuladores pertenecen a la clase de sistemas dinámicos lagrangianos, los cuales tienen varias propiedades físicas agradables que hacen que su control sea sencillo [4].

El controlador proporcional, integral y derivativo (PID), es uno de los controladores más sencillos para implementar en robots manipuladores y uno de los más usados en la industria. Sin embargo, no quiere decir que sea el controlador más eficiente [3]. Este controlador ha sido ampliamente estudiado a lo largo de los años. A continuación, mencionaremos algunos trabajos relacionados. Un procedimiento de sintonización para controladores PID de robots rígidos es propuesto en [5]. En [6], se diseñó un controlador PID con el método de sintonización de ganancias por Ziegler-Nichols y se implementó a un sistema de control de temperatura. En [7], se propuso un algoritmo de optimización, basado en un algoritmo mejorado inspirado en un riñón, para la sintonización de un controlador PID aplicado a un regulador de voltaje automático. En [8], se estudiaron los enfoques modernos y clásicos utilizados para el ajuste de PID y sus aplicaciones en varios dominios.

Por otra parte, el control adaptable es aquel que tratará de realizar una estimación en línea del proceso desconocido. Luego, producirá una entrada de control para anticipar, sobrellevar o minimizar la desviación no deseada del comportamiento de lazo cerrado del sistema a controlar [9]. A continuación, se mencionan algunos trabajos relacionados con control adaptable. En [10], se demostró la conexión entre el control adaptable y los métodos de optimización comúnmente usados en aprendizaje automático. En [11], se propuso un esquema de control adaptable para sistemas no lineales inciertos conmutados usando el método del tiempo medio de permanencia modo dependiente. En [12], se diseñó un controlador adaptable basado en la técnica de backstepping y barreras de Lyapunov para una clase de sistemas conmutados inciertos con realimentación estricta.

En este documento, se presenta un controlador de regulación para robots manipuladores de n grados de libertad. Debido a que en la práctica se tienen incertidumbres paramétricas, se propone un controlador con compensación de gravedad adaptable. Además, se le suma un controlador tipo PID para robustecer el método propuesto. Se presenta un análisis de estabilidad, donde se concluye estabilidad asintótica de manera global. Se presentan simulaciones numéricas para validar la teoría

---

*Trabajo apoyado por el Consejo Nacional de Ciencia y Tecnología, CONACYT-Fondo Sectorial de Investigación para la Educación con el proyecto A1-S-24762, SIP-IPN, y por la Secretaría de Investigación y Posgrado-Instituto Politécnico Nacional, México. Proyecto apoyado por el Fondo Sectorial de Investigación para la Educación.

presentada y se hizo una comparación con un controlador PID lineal previamente reportado en la literatura, demostrando que el controlador propuesto tiene mejor rendimiento.

El contenido de este documento se organiza de la siguiente manera. La sección 2 presenta el modelo dinámico general del manipulador, así como los preliminares y propiedades matemáticas a utilizar en este documento. La sección 3 presenta el controlador propuesto con su prueba de estabilidad. La sección 4 presenta la descripción del robot manipulador a utilizar y los resultados de la simulación. Finalmente, la sección 5 concluye los resultados obtenidos.

II. MODELO DINÁMICO DE ROBOTS MANIPULADORES, PRELIMINARES MATEMÁTICOS Y OBJETIVO DE CONTROL

A. Modelo dinámico de robots manipuladores [3]

Los robots manipuladores son sistemas compuestos por articulaciones conectados por pares cinemáticos. El modelo dinámico de un robot manipulador rígido con n grados de libertad se expresa de la siguiente manera:

$$M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) + F_v\dot{\boldsymbol{q}} = \boldsymbol{\tau}, \tag{1}$$

donde los vectores $\boldsymbol{q},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}} \in \mathbb{R}^n$, describen la posición, velocidad y aceleración de cada articulación, respectivamente, $M(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ es la matriz de inercias, $C(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$ es la matriz de fuerzas centrífugas y de Coriolis, $\boldsymbol{g}(\boldsymbol{q}) \in \mathbb{R}^n$ es el vector de pares gravitacionales, $F_v \in \mathbb{R}^{n\times n}$ es la matriz de fricción viscosa y $\boldsymbol{\tau} \in \mathbb{R}^n$ es el vector de pares de cada actuador. [3]

B. Preliminares matemáticos [3]

Propiedad 1. La matriz de inercia $M(\boldsymbol{q})$ es una matriz simétrica definida positiva.

Propiedad 2. Para el caso de robots provisto únicamente de articulaciones rotacionales, existe una constante $k_{C1} > 0$ para toda $\boldsymbol{q},\boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^n$, tal que:

$$\|C(\boldsymbol{q},\boldsymbol{x})\boldsymbol{y}\| \le k_{C1}\|\boldsymbol{x}\|\|\boldsymbol{y}\|.$$

Propiedad 3. La matriz de fuerzas centrifugas y de Coriolis $C(\boldsymbol{q},\dot{\boldsymbol{q}})$ esta relacionada con la matriz de inercia $M(\boldsymbol{q})$, por la expresión:

$$\dot{\boldsymbol{q}}^T[\frac{1}{2}\dot{M}(\boldsymbol{q}) - C(\boldsymbol{q},\dot{\boldsymbol{q}})]\dot{\boldsymbol{q}} = 0, \quad \forall \boldsymbol{q},\dot{\boldsymbol{q}} \in \mathbb{R}^n.$$

Además, resulta cierto que:

$$\dot{M}(\boldsymbol{q}) = C(\boldsymbol{q},\dot{\boldsymbol{q}}) + C(\boldsymbol{q},\dot{\boldsymbol{q}})^T.$$

Propiedad 4. Sea A una matriz de dimensión $n\times n$, con a_{ij} representando los elementos de la fila i y columna j. La matriz A puede ser llamada estrictamente matrix diagonal dominante, si [5]:

$$|a_{ii}| > \sum_{j=1,j\neq i}^{n} |a_{ij}|, \quad i = 1,\dots,n.$$

Propiedad 5. Linealidad en términos de los parámetros, El modelo dinámico expresado en (1), se puede representar de la siguiente forma:

$$\begin{aligned} M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{g}(\boldsymbol{q}) + F_v\dot{\boldsymbol{q}} &= \\ M(\boldsymbol{q})\ddot{\boldsymbol{q}} + C(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} &+ \Phi_g(\boldsymbol{q})\boldsymbol{\theta} + F_v\dot{\boldsymbol{q}}, \end{aligned}$$

donde $\Phi_g(\boldsymbol{q}) \in \mathbb{R}^{n\times p}$ es la matriz de regresión del vector de gravedad, $\boldsymbol{\theta} \in \mathbb{R}^p$ es el vector de parámetros agrupados y p es el número de parámetros.

1) Propiedades de la tangente hiperbólica: Se define la función vectorial de tangentes hiperbólicas de la siguiente manera:

$$\mathbf{tanh}(\boldsymbol{x}) = \begin{bmatrix} \tanh(x_i) \\ \vdots \\ \tanh(x_n) \end{bmatrix}, \quad \boldsymbol{x} \in \mathbb{R}^n.$$

La función vectorial de la tangente hiperbólica satisface las siguientes propiedades:

(1.a). La derivada de la función vectorial de la tangente hiperbólica se expresa como

$$\frac{d}{dx}\mathbf{tanh}(\boldsymbol{x}) = -Sech^2(\boldsymbol{x})\dot{\boldsymbol{x}},$$

donde $Sech^2(\boldsymbol{x}) := diag\{sech^2(x_i)\}$ con $i = 1, 2, \dots, n$, es una matriz diagonal cuyos elementos, $sech^2(x_i)$, son positivos y menores que 1.

(1.b). La norma euclidiana de la función vectorial de la tangente hiperbólica satisface que:

$$\begin{aligned} \|\mathbf{tanh}(\boldsymbol{x})\| &\le \sqrt{n}, \\ \|\mathbf{tanh}(\boldsymbol{x})\| &\le \|\boldsymbol{x}\|. \end{aligned}$$

(1.c). Se cumple la siguiente propiedad [13]:

$$\sqrt{\sum_{i=1}^{n} \ln(\cosh(x_i))} \ge \frac{1}{\sqrt{2}}\|\mathbf{tanh}(\boldsymbol{x})\|, \quad \forall \boldsymbol{x} \in \mathbb{R}^n.$$

C. Objetivo de control

El objetivo de control consiste en diseñar una entrada de control $\boldsymbol{\tau}$ tal que la señal $\boldsymbol{q}(t)$ satisfaga lo siguiente:

$$\lim_{t\to\infty} \boldsymbol{q}(t) = \boldsymbol{q}_d,$$

donde $\boldsymbol{q}(t)$ es la posición actual del sistema y $\boldsymbol{q}_d \in \mathbb{R}^n$ es un vector constante.

III. CONTROLADOR PID CON COMPENSACIÓN DE GRAVEDAD ADAPTABLE

Se propone la siguiente ley de control con estructura PID con compensación de gravedad adaptable

$$\boldsymbol{\tau} = K_p\tilde{\boldsymbol{q}} - K_v\dot{\boldsymbol{q}} + K_I\boldsymbol{\xi} + \Phi_g\hat{\boldsymbol{\theta}}, \tag{2}$$

donde K_p, K_v y $K_I \in \mathbb{R}^{n\times n}$ son matrices diagonales y definidas positivas, $\Phi_g \in \mathbb{R}^{n\times p}$ es la matriz de regresión correspondiente al vector de gravedad, $\hat{\boldsymbol{\theta}} \in \mathbb{R}^p$ es el vector de parámetros estimados del vector de gravedad y $\boldsymbol{\xi} \in \mathbb{R}^n$ es una variable de estado, cuya derivada temporal está definida por

$$\dot{\boldsymbol{\xi}} = \alpha \mathbf{tanh}(\tilde{\boldsymbol{q}}) - \dot{\boldsymbol{q}}, \tag{3}$$

donde α es un escalar postivo y $\tilde{\boldsymbol{q}} \in \mathbb{R}^n$ es el error de posición definido como $\tilde{\boldsymbol{q}} = \boldsymbol{q}_d - \boldsymbol{q}$.

La ley de adaptación usada para estimar los valores de $\hat{\boldsymbol{\theta}}$ en (2), es

$$\dot{\hat{\boldsymbol{\theta}}} = \Gamma \Phi_g^T (\alpha \mathbf{tanh}(\tilde{\boldsymbol{q}}) - \dot{\boldsymbol{q}}), \tag{4}$$

donde $\Gamma \in \mathbb{R}^{n\times n}$ es una matriz diagonal definida positiva.

Al sustituir (2) en el modelo dinámico (1), se obtiene

$$\begin{aligned} M(\boldsymbol{q})\ddot{\boldsymbol{q}} &= K_p\tilde{\boldsymbol{q}} - K_v\dot{\boldsymbol{q}} + K_I\boldsymbol{\xi} + \Phi_g\hat{\boldsymbol{\theta}} - C(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} \\ &\quad -\boldsymbol{g}(\boldsymbol{q}) - F_v\dot{\boldsymbol{q}}. \end{aligned} \tag{5}$$

Tomando en cuenta la propiedad 5, se tiene que

$$\boldsymbol{g}(\boldsymbol{q}) = \Phi_g \boldsymbol{\theta}, \tag{6}$$

donde $\boldsymbol{\theta} \in \mathbb{R}^m$ son los parámetros reales que pertenecen al vector de gravedad.

Finalmente, se obtiene el sistema de lazo cerrado como

$$\begin{aligned} \frac{d}{dt}\tilde{\boldsymbol{q}} &= -\dot{\boldsymbol{q}}, && (7)\\ \frac{d}{dt}\dot{\boldsymbol{q}} &= M(\boldsymbol{q})^{-1}(K_p\tilde{\boldsymbol{q}} - K_v\dot{\boldsymbol{q}} + K_I\boldsymbol{\xi} + \Phi_g\tilde{\boldsymbol{\theta}} \\ &\quad -C(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - F_v\dot{\boldsymbol{q}}), && (8)\\ \frac{d}{dt}\boldsymbol{\xi} &= \alpha\mathbf{tanh}(\tilde{\boldsymbol{q}}) - \dot{\boldsymbol{q}}, && (9)\\ \frac{d}{dt}\tilde{\boldsymbol{\theta}} &= \Gamma\Phi_g^T(\alpha\mathbf{tanh}(\tilde{\boldsymbol{q}}) - \dot{\boldsymbol{q}}). && (10) \end{aligned}$$

donde el origen es un punto de equilibrio y $\tilde{\boldsymbol{\theta}}$ es el error de los parámetros definido como $\tilde{\boldsymbol{\theta}} = \hat{\boldsymbol{\theta}} - \boldsymbol{\theta}$.

Note que debido a que $\boldsymbol{\theta}$ es constante, $\dot{\tilde{\boldsymbol{\theta}}}$ es igual a $\dot{\hat{\boldsymbol{\theta}}}$. En la figura 1 se muestra el diagrama de bloques del controlador (2).

A. *Prueba de estabilidad*

Para analizar la estabilidad del sistema de lazo cerrado (7)-(10), se propone la siguiente función candidata

$$\begin{aligned} V &= \frac{1}{2}[-\dot{\boldsymbol{q}} + \alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})]^T M(\boldsymbol{q})[-\dot{\boldsymbol{q}} + \alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})] \\ &\quad -\frac{1}{2}\alpha^2\mathbf{tanh}(\tilde{\boldsymbol{q}})^T M(\boldsymbol{q})\mathbf{tanh}(\tilde{\boldsymbol{q}}) \\ &\quad +\alpha\sum_{i=1}^{n}(K_{vi} + F_{vi})\ln(\cosh(\tilde{q}_i)) \\ &\quad +\frac{1}{2}\tilde{\boldsymbol{q}}^T K_p\tilde{\boldsymbol{q}} + \frac{1}{2}\boldsymbol{\xi}^T K_I\boldsymbol{\xi} + \frac{1}{2}\tilde{\boldsymbol{\theta}}^T\Gamma^{-1}\tilde{\boldsymbol{\theta}}. \end{aligned} \tag{11}$$

Usando las propiedades de la tangente hiperbólica, la ecuación (11) se puede acotar inferiormente, obteniendo

$$\begin{aligned} V &\geq \frac{1}{2}[-\dot{\boldsymbol{q}} + \alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})]^T M(\boldsymbol{q})[-\dot{\boldsymbol{q}} + \alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})] \\ &\quad +\frac{1}{2}\mathbf{tanh}(\tilde{\boldsymbol{q}})^T A_1\mathbf{tanh}(\tilde{\boldsymbol{q}}) + \frac{1}{2}\tilde{\boldsymbol{\theta}}^T\Gamma^{-1}\tilde{\boldsymbol{\theta}} + \frac{1}{2}\boldsymbol{\xi}^T K_I\boldsymbol{\xi} \\ &\quad +\frac{1}{2}\tilde{\boldsymbol{q}}^T K_p\tilde{\boldsymbol{q}} + \frac{1}{2}\alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})^T F_v\mathbf{tanh}(\tilde{\boldsymbol{q}}). \end{aligned} \tag{12}$$

donde $A_1 = [\alpha K_v - \alpha^2 M(\boldsymbol{q})]$. Por medio del teorema de Sylvester se garantiza que V sea definida positiva [3]. Usando la propiedad 4, se garantiza que A_1 sea definida positiva por medio de la siguiente condición:

$$\frac{K_{vi}}{\sum_{i=1}^{n}\max|M_{ij}|} > \alpha, \quad i = 1, \ldots, n. \tag{13}$$

Diferenciando (11) con respecto al tiempo a lo largo de las trayectorias del sistema de lazo cerrado (7)-(10), se obtiene

$$\begin{aligned} \dot{V} &= \dot{\boldsymbol{q}}^T(K_p\tilde{\boldsymbol{q}} - K_v\dot{\boldsymbol{q}} + K_I\boldsymbol{\xi} + \Phi_g\tilde{\boldsymbol{\theta}} - C(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - F_v\dot{\boldsymbol{q}}) \\ &\quad +\frac{1}{2}\dot{\boldsymbol{q}}^T\dot{M}(\boldsymbol{q})\dot{\boldsymbol{q}} - \alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})^T\dot{M}(\boldsymbol{q})\dot{\boldsymbol{q}} \\ &\quad -\alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})^T(K_p\tilde{\boldsymbol{q}} - K_v\dot{\boldsymbol{q}} + K_I\boldsymbol{\xi} + \Phi_g\tilde{\boldsymbol{\theta}} \\ &\quad -C(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - F_v\dot{\boldsymbol{q}}) + \alpha\dot{\boldsymbol{q}}^T Sech^2(\tilde{\boldsymbol{q}})M(\boldsymbol{q})\dot{\boldsymbol{q}} \\ &\quad -\alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})^T(K_v + F_v)\dot{\boldsymbol{q}} - \tilde{\boldsymbol{q}}^T K_p\dot{\boldsymbol{q}} \\ &\quad +\tilde{\boldsymbol{\theta}}^T\Gamma^{-1}\dot{\tilde{\boldsymbol{\theta}}} + \boldsymbol{\xi}^T K_I\dot{\boldsymbol{\xi}}. \end{aligned} \tag{14}$$

Nótese que K_v puede ser representada por la siguiente sumatoria $K_v = K_{v\gamma} + K_{v\sigma}$, y usando las propiedades 2, 3 y las propiedades de la tangente hiperbólica, se pueden establecer las siguientes cotas:

$$\begin{aligned} \alpha\dot{\boldsymbol{q}}^T Sech^2(\tilde{\boldsymbol{q}})M(\boldsymbol{q})\dot{\boldsymbol{q}} &\leq \alpha\dot{\boldsymbol{q}}^T M^*(\boldsymbol{q})\dot{\boldsymbol{q}}, \\ -\alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})^T K_p\tilde{\boldsymbol{q}} &\leq -\alpha\lambda_{min}\{K_p\}\|\mathbf{tanh}(\tilde{\boldsymbol{q}})\|^2, \\ \|\alpha\mathbf{tanh}(\tilde{\boldsymbol{q}})^T C(\boldsymbol{q},\dot{\boldsymbol{q}})^T\dot{\boldsymbol{q}}\| &\leq \alpha\sqrt{n}k_{C1}\|\dot{\boldsymbol{q}}\|^2, \\ -\dot{\boldsymbol{q}}^T K_{v\sigma}\dot{\boldsymbol{q}} &\leq -\lambda_{min}\{K_{v\sigma}\}\|\dot{\boldsymbol{q}}\|^2, \\ -\dot{\boldsymbol{q}}^T F_v\dot{\boldsymbol{q}} &\leq -\lambda_{min}\{F_v\}\|\dot{\boldsymbol{q}}\|^2. \end{aligned}$$

donde $M^*(\boldsymbol{q}) = [\frac{1}{2}Sech^2(\tilde{\boldsymbol{q}})M(\boldsymbol{q}) + \frac{1}{2}M(\boldsymbol{q})Sech^2(\tilde{\boldsymbol{q}})]$. Por lo tanto, (14) se puede acotar superiormente por

$$\begin{aligned} \dot{V} &\leq -\dot{\boldsymbol{q}}^T A_2\dot{\boldsymbol{q}} \\ &\quad -\alpha\begin{bmatrix}\|\mathbf{tanh}(\tilde{\boldsymbol{q}})\| & \|\dot{\boldsymbol{q}}\|\end{bmatrix} Q \begin{bmatrix}\|\mathbf{tanh}(\tilde{\boldsymbol{q}})\| \\ \|\dot{\boldsymbol{q}}\|\end{bmatrix}, \end{aligned} \tag{15}$$

donde

$$\begin{aligned} A_2 &= (K_{v\gamma} - \alpha M^*(\boldsymbol{q})), \\ Q &= \begin{bmatrix}\lambda_{min}\{K_p\} & 0 \\ 0 & Q_{22}\end{bmatrix}. \end{aligned}$$

con

$$Q_{22} = \frac{1}{\alpha}\lambda_{min}\{K_{v\sigma}\} - \sqrt{n}K_{c1} + \frac{1}{\alpha}\lambda_{min}\{F_v\}.$$

Para garantizar que $\dot{V}$ sea negativa, la cota superior (15) debe cumplir que las matrices Q y A_2 sean definidas positivas. Por lo cual, aplicando la propiedad 1 y 4, se obtiene las condiciones:

$$\begin{aligned} \frac{K_{v\gamma i}}{\sum_{i=1}^{n}\max|M^*_{ij}|(\boldsymbol{q})} &> \alpha, \quad i = 1, \ldots, n, && (16)\\ \lambda_{min}\{K_p\} &> 0, && (17)\\ \frac{\lambda_{min}\{K_{v\sigma}\} + \lambda_{min}\{F_v\}}{\sqrt{n}k_{C1}} &> \alpha. && (18) \end{aligned}$$

Nótese que si se cumple la condición (16) se garantiza (13). Con $\dot{V}(t) \leq 0$ se concluye que la función $V(t)$ es una función

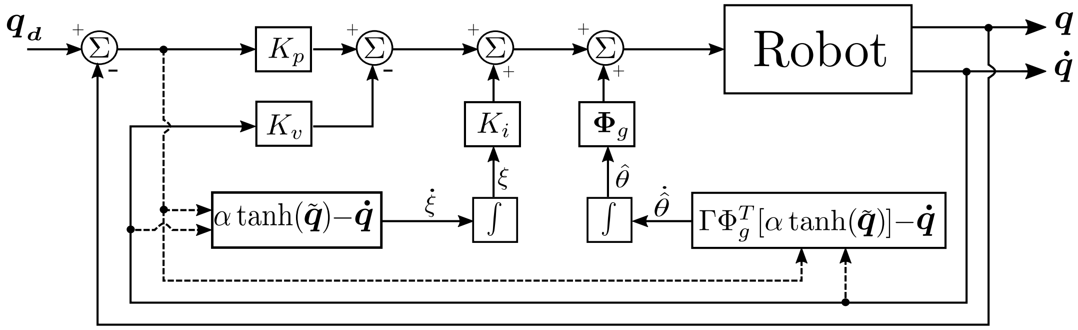


Fig. 1. Diagrama de control PID con compensación de gravedad adaptable.

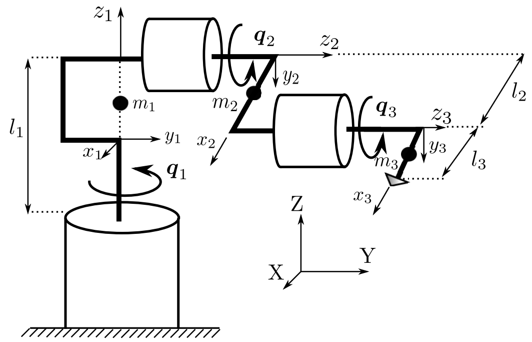


Fig. 2. Representación del robot manipulador de 3 grados de libertad.

de Lyapunov, por lo que el origen es estable de manera global y las soluciones de $\tilde{\boldsymbol{q}}(t)$, $\dot{\boldsymbol{q}}(t)$, $\boldsymbol{\xi}(t)$ y $\tilde{\boldsymbol{\theta}}(t)$ están acotadas. Con esto tenemos las condiciones necesarias para aplicar el lema de Barbalat [14], y concluir que

$$\lim_{t\to\infty} \tilde{\boldsymbol{q}}(t) = 0.$$

IV. RESULTADOS

A. Descripcion del robot manipulador

Considerando un robot de 3 grados de libertad, como el mostrado en la figura 2, cuyos elementos del modelo dinámico (1) se encuentran expresados por

$$M(\boldsymbol{q}) = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix}, \tag{19}$$

donde

$$\begin{aligned}
M_{11} &= \theta_1 + \theta_3 \cos^2(q_2) - \theta_2 \sin^2(q_2+q_3) \\
&\quad + \theta_4 \left(\cos(2q_2+q_3) + \cos(q_3) \right), \\
M_{12} &= \theta_7 \sin(q_2+q_3) + \theta_6 \sin(q_2), \\
M_{13} &= \theta_7 \sin(q_2+q_3), \\
M_{21} &= \theta_7 \sin(q_2+q_3) + \theta_6 \sin(q_2), \\
M_{22} &= \theta_5 + \theta_8 + 2\theta_4 \cos(q_3), \\
M_{23} &= \theta_8 + \theta_4 \cos(q3), \\
M_{31} &= \theta_7 \sin(q_2+q_3), \\
M_{32} &= \theta_8 + \theta_4 \cos(q3), \\
M_{33} &= \theta_8.
\end{aligned}$$

Los elementos de la matriz de fuerzas centrifugas y de Coriolis son:

$$C(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}, \tag{20}$$

donde

$$\begin{aligned}
C_{11} &= -\dot{q_3}(\theta_4(\sin(2q_2+q_3) + \sin(q_3)) \\
&\quad + \theta_2 \sin(2q_2+2q_3))/2 \\
&\quad - \dot{q_2}((\theta_2 \sin(2q_2+2q_3))/2 + \theta_4 \sin(2q_2+q_3)) \\
&\quad - \dot{q_2}\theta_3 \cos(q_2)\sin(q_2), \\
C_{12} &= -\dot{q_1}((\theta_2 \sin(2q_2+2q_3))/2 + \theta_4 \sin(2q_2+q_3)) \\
&\quad - \dot{q_1}\theta_3 \cos(q_2)\sin(q_2) + \dot{q_3}\theta_7 \cos(q_2+q_3), \\
C_{13} &= -\dot{q_1}(\theta_4(\sin(2q_2+q_3) + \sin(q_3)) \\
&\quad + \theta_2 \sin(2q_2+2q_3))/2 \\
&\quad + \dot{q_2}\theta_7 \cos(q_2+q_3) + \dot{q_3}\theta_7 \cos(q_2+q_3), \\
C_{21} &= \dot{q_1}(\theta_4 \sin(2q_2+q_3) + \theta_3 \cos(q_2)\sin(q_2)) \\
&\quad + \dot{q_1}(\theta_2 \sin(2q_2+2q_3))/2, \\
C_{22} &= -\theta_4 \dot{q_3} \sin(q_3), \\
C_{23} &= -\theta_4 \sin(q_3)(\dot{q_2} + \dot{q_3}), \\
C_{31} &= \theta_2 \dot{q_1} \sin(q_2+q_3)\cos(q_2+q_3) \\
&\quad + \theta_4 \dot{q_1}(\sin(2q_2+q_3) + \sin(q_3))/2, \\
C_{32} &= \theta_4 \dot{q_2} \sin(q_3), \\
C_{33} &= 0.
\end{aligned}$$

El vector de gravedad está dado por

$$\boldsymbol{g}(\boldsymbol{q}) = \begin{bmatrix} 0 \\ -g\theta_9 \cos(q_2) - g\theta_{10}\cos(q_2+q_3) \\ -g\theta_{10}\cos(q_2+q_3) \end{bmatrix}, \tag{21}$$

donde la aceleración gravitacional es $g = 9.81 m/s^2$.

TABLA I
PARÁMETROS DEL ROBOT DE 3 GRADOS DE LIBERTAD

Parámetros	**Valor**	**Unidades**
θ_1	2.8717	Kg m^2
θ_2	0.2297	Kg m^2
θ_3	0.6352	Kg m^2
θ_4	0.0755	Kg m^2
θ_5	0.1917	Kg m^2
θ_6	0.0447	Kg m^2
θ_7	0.1183	Kg m^2
θ_8	0.0984	Kg m^2
θ_9	1.0655	Kg m
θ_{10}	0.2286	Kg m
θ_{11}	5.7135	Kg m^2/s
θ_{12}	0.5968	Kg m^2/s
θ_{13}	0.9986	Kg m^2/s
θ_{14}	5.8697	Kg m^2/s^2
θ_{15}	2.6923	Kg m^2/s^2
θ_{16}	4.7547	Kg m^2/s^2

La matriz de fricción viscosa está dada por

$$F_v = \begin{bmatrix} \theta_{11} & 0 & 0 \\ 0 & \theta_{12} & 0 \\ 0 & 0 & \theta_{13} \end{bmatrix}. \tag{22}$$

Se ha considerado una aproximación continua del modelo de fricción de Coulomb como sigue

$$\boldsymbol{f}_c(\dot{\boldsymbol{q}}) = \begin{bmatrix} \theta_{14}\tanh(100\dot{q}_1) \\ \theta_{15}\tanh(100\dot{q}_2) \\ \theta_{16}\tanh(100\dot{q}_3) \end{bmatrix}. \tag{23}$$

La tabla I muestra los valores de los parámetros a utilizar en (19) - (23), los cuales se obtuvieron en [15].

Nótese que el vector de gravedad (21), se puede expresar como en (6), como

$$\Phi_g = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -g\cos(q_2) & -g\cos(q_2+q_3) \\ 0 & 0 & -g\cos(q_2+q_3) \end{bmatrix}, \boldsymbol{\theta} = \begin{bmatrix} 0 \\ \theta_9 \\ \theta_{10} \end{bmatrix}.$$

B. Controlador a comparar

Para evaluar el desempeño de la propuesta de control PID con compensación de gravedad adaptable (2), se realizaron simulaciones con un controlador conocido en la literatura.

El controlador usado para comparar, propuesto en [5], es un control PID lineal, expresado por:

$$\boldsymbol{\tau} = K_p\tilde{\boldsymbol{q}} - K_v\dot{\boldsymbol{q}} + K_I\boldsymbol{z}, \tag{24}$$

donde $\boldsymbol{z}$ es una variable de estado, cuya derivada temporal se define como

$$\dot{\boldsymbol{z}} = \alpha\tilde{\boldsymbol{q}} - \dot{\boldsymbol{q}},$$

Es importante aclarar que las condiciones en la selección de ganancias que aseguran la estabilidad de (24) pueden ser consultadas en [5].

TABLA II
GANANCIAS DE CONTROL

Ganancias	**Valor**
K_p	diag{50,40,40}
K_v	diag{10,5,5}
K_I	diag{10,10,10}
α	0.52
Γ	diag{1,1,1}

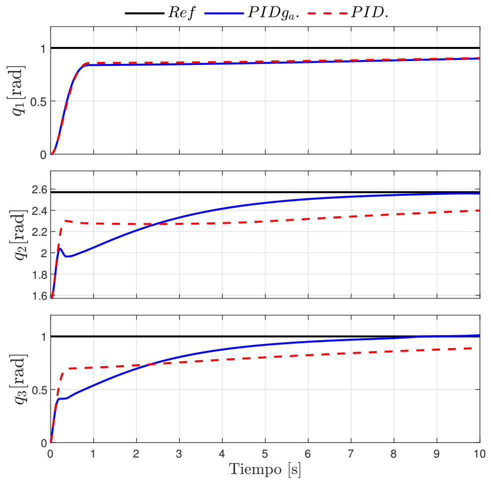


Fig. 3. Evolución temporal de las posiciones $\boldsymbol{q}(t)$.

C. Resultados de simulación

En esta sección, se presentan los resultados de simulación de los controladores mencionados anteriormente. Las ganancias seleccionadas para el controlador propuesto (2)-(4) y el controlador PID lineal (24) se encuentran en la tabla II.

Nótese que las ganancias presentadas en la tabla II cumplen las condiciones de estabilidad presentadas en [5] y las ecuaciones (16)-(18). El objetivo de seleccionar las mismas ganancias para los dos controladores es demostrar que tanto la acción integral del controlador propuesto, como la acción adaptable, juegan un papel fundamental en el desempeño del controlador.

Las condiciones iniciales seleccionadas para todas las simulaciones son de $\boldsymbol{q}(0) = [0 \quad \pi/2 \quad 0]^T$, las cuales representan el estado de reposo. La posición deseada seleccionada es de $\boldsymbol{q}_d = [1 \quad (2+\pi)/2 \quad 1]^T$ y un tiempo de simulación de 10 segundos.

Las figuras 3-5 muestran los resultados en simulación para cada unos de los controladores, donde la etiqueta $PIDg_a$ denota el controlador propuesto en (2) y (4) y la etiqueta PID es para el controlador (24).

La figura 3 muestra las posiciones $\boldsymbol{q}(t)$ de cada articulación del manipulador. Los resultados toman en cuenta la fricción de Coulomb y se observa que la posición del manipulador está tendiendo al de referencia.

Los pares aplicados $\boldsymbol{\tau}(t)$ se muestran en la figura 4. Noté que las vibraciones que se pueden observar son debido a la

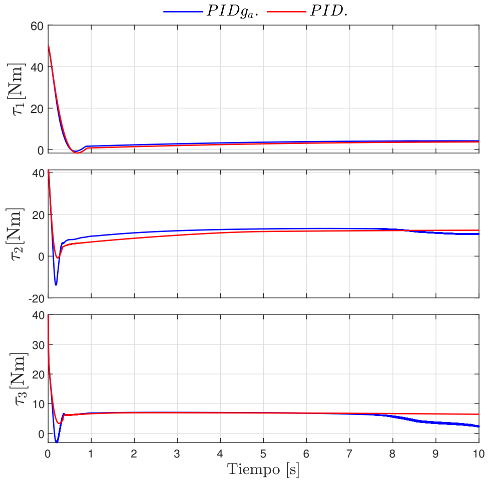


Fig. 4. Torque ($\boldsymbol{\tau}$) para cada articulación.

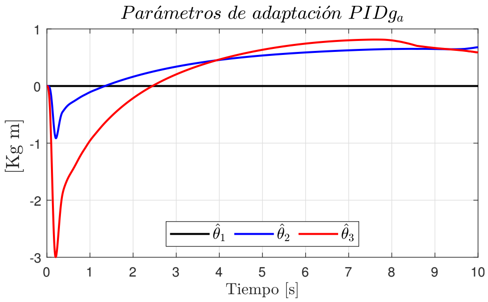


Fig. 5. Evolución temporal de los parámetros de adaptación $\hat{\boldsymbol{\theta}}(t)$.

TABLA III

VALOR RMS DEL ERROR DE POSICIÓN EN $5 \leq t \leq 10$ SEGUNDOS

$\tilde{\boldsymbol{q}}$	PID	$PIDg_a$
$\tilde{q}_1$	**0.11280**	0.12170
$\tilde{q}_2$	0.22290	**0.04790**
$\tilde{q}_3$	0.15230	**0.03610**

fricción de Couloumb. Los parámetros de adaptación $\hat{\boldsymbol{\theta}}(t)$ del vector de gravedad, se muestran en la figura 5, con una condición inicial $\hat{\boldsymbol{\theta}}(0) = [0 \quad 0 \quad 0]^T$ y un valor constante de $\hat{\theta}_1 = 0$ dado que el primer eslabón no es afectado por la fuerza gravitacional.

La tabla III muestra el valor RMS del error de posición $\tilde{\boldsymbol{q}}(t)$ de cada uno de los controladores, en el intervalo de 5 a 10 segundos. Se puede observar que el desempeño del controlador propuesto (2) y (4) es mejor que el controlador (24) para las articulaciones 2 y 3, donde se obtiene un valor RMS menor. Sin embargo, la articulación 1 tiene un aparente mejor rendimiento para el caso del controlador (24). No obstante, la diferencia en el valor RMS de $\tilde{\boldsymbol{q}}_1(t)$ entre ambos controladores no es significativa.

V. CONCLUSIONES

En este trabajo, se presentó un control PID con compensación de gravedad adaptable para regulación de posición de robots manipuladores. Además, se presentó su prueba de estabilidad, la cual concluye estabilidad asintótica de manera global. Se realizaron simulaciones numéricas y se hicieron comparaciones con un controlador conocido en la literatura, un controlador PID lineal. Los resultados muestran las ventajas de contemplar tanto la acción integral como la compensación de gravedad adaptable. Los resultados son avalados por medio del valor RMS del error de posición en estado estacionario.

REFERENCIAS

[1] M. S. Fadali y A. Visolini, *Digital Control Engineering Analysis and design*, 2^{nd} ed., Waltham: Academic Press, 2013.
[2] K. Ogata, *Ingenieria de Control Moderna*, 5^{th} ed., Madrid: Pearson education, 2010.
[3] R. Kelly, V. Santibañez, and A. Loría. *Control of Robot Manipulators in Joint Space*, London: Springer-Verlag, 2006.
[4] F. L. Lewis, A. Yesildirek, and S. Jagannathan. *Neural Network Control of Robot Manipulators and Nonlinear Systems*, London: Taylor and Francis, 1999.
[5] V. M. Hernández-Guzmán, V. Santibáñez y R. Silva-Ortigoza, “A new tuning procedure for PID control of rigid robots”, *Advanced Robotics*, vol. 22, no. 9, pp. 1007-1023, 2008.
[6] T. K. Palaniyappan, V. Yadav, Ruchira, V. K. Tayal, and P. Choudekar, “PID control design for a temperature control system”, *in proc. of the International Conference on Power Energy, Environment and Intelligent Control*, 2018, pp. 632-637.
[7] S. Ekinci and B. Hekimoğlu, “Improved kidney-inspired algorithm approach for tuning of PID controller in AVR system”, *IEEE Access*, vol. 7, pp. 39935-39947, 2019.
[8] R. P. Borase, D. K. Maghade, S. Y. Sondkar, and S. N. Pawar. “A review of PID control, tuning methods and applications”, *International Journal of Dynamics and Control*, vol. 9, pp. 818–827, 2021.
[9] E. Lavretsky and K.A. Wise, *Robust Adaptiive Control with Aerospace Applications*, London: Springer-Verlag, 2013.
[10] J. E. Gaudio, T. E. Gibson, A.M. Annaswamy, M. A. Bolender, and E. Lavretsky, “Connections between adaptive control and optimization in machine learning”, *in proc. of the IEEE 58th Conference on Decision and Control*, 2019, pp. 4563-4568.
[11] N. Ben, Z. Ping, L. Ji-Dong, M. Hong-Jun, and L. Yan-Jun, “Global adaptive control of switched uncertain nonlinear systems: An improved MDADT method”, *Automatica*, vol 115, pp. 1088-1098, 2020.
[12] L. Lei, L. Yan-Jun, C. Aiqing, T. Shaocheng, and C. Philip, “Integral barrier Lyapunov function-based adaptive control for switched nonlinear systems”, *Science China Information Sciences*, vol 63, no. 3, pp.1869-1919, 2020.
[13] F. Zhang, D. M. Dawson, M. S. de Queiroz, and W. E. Dixon, “Global adaptive output feedback tracking control of robot manipulators”, *IEEE Transactions on Automatic Control*, vol. 45, no. 6, pp. 1203-1208, 2000.
[14] H. K. Khalil. *Nonlinear Systems*, Upper Saddle River: Prentice Hall, 2002.
[15] J. Montoya-Cháirez, J. Quijada, O. García-Alarcón, R. Pérez-Alcocer, and J. Moreno-Valenzuela, “Estudio de identificación paramétrica en un robot de tres grados de libertad”, *in proc. of the XXI Congreso Mexicano de Robótica (COMRob) 2019*, Manzanillo, Colima, México pp. 256-258, 2019.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Study of performances of manipulators using Jacobian matrices homogenized by two approaches

Daniel Roberto Soto Delgado[1], José Alfonso Pámanes García[2]
[1 2]*Division of Postgraduate Studies and Research*
Tecnologico Nacional de Mexico-Instituto Tecnologico de la Laguna
Blvd. Revolución y Calz. Cuauhtémoc, Torreón, Coahuila, México, C.P. 27000
[1]idanielsotod@gmail.com, [2]japamanesg@correo.itlalaguna.edu.mx

***Abstract*—A simulation study is accomplished in this paper to assess the effectiveness of two indices of performance to measure the ability of postures of industrial robots for converting joint-rates into motion of the end-effector. Both classes of motion of the end effector, translational and rotational, are simultaneously considered for evaluation. Consequently, the whole Jacobian matrix must be homogenized such that all its entries be coherently mixed in any index of performance. Specifically, two approaches for homogenization of the Jacobian matrix are compared. The first one, based on the so-called *characteristic length* of the manipulator, and the second one that uses the inertial parameters of the end-effector. The results show that the second approach allows to measure the performances of the manipulator with a clear physical meanning. To the knowledge of the authors, such a meaning cannot be obtained with any other index of performance based on a different homogenized Jacobian matrix.**

***Keywords*—Jacobian matrix, homogenization, characteristic length, performance, optimization**

I. INTRODUCTION

The kinematic performance of robotic manipulators has been largely studied in the scientific literature over the last few decades. Most of the applied indices of performance in such studies are naturally based on the use of the Jacobian matrix of the manipulators. This matrix determines the relationship between the joint-rates and the twist (or Cartesian state of velocity) of the end-effector (EE) of a manipulator. Examples of approaches to describe the kinematic performance of manipulators are presented in [1]- [9] and applications of such indices, in designing and motion planning of manipulators, can be found in [10]- [16]. Nevertheless, the lack of homogeneity of the whole Jacobian matrix makes it difficult to simultaneously measure the efficacy to produce both, translational and rotational, motions of the EE. Indeed, the ability of postures of a manipulator for completing such a conversion could be effectively measured only if the entries of the matrix are homogeneous and the applied performance index yields values with a clear physical meaning.

The scientific community has proposed different approaches to avoid or mitigate the issue of the non-homogeneity of the Jacobian matrix. Gosselin [13] applied a Jacobian matrix that characterizes the relationship between the joint-rates of the manipulator and the linear velocity of two specific points in the EE, instead of considering linear and rotational velocities. On the other hand, Angeles and Lopez-Cajun [4] proposed the use of the characteristic length of the manipulator for dividing the entries associated to linear velocity, in such a way that all the entries of the Jacobian matrix are dimensionless. The *characteristic length* will be defined in section 3.

Yoshikawa [2] proposed partitions of the Jacobian matrix, corresponding to the translational and rotational parts of the twist of the EE, in order to compute the translational and rotational manipulabilities, respectively, of the manipulator. Clearly, this approach provides one first aproximation for simultaneously assessing the ease to produce translational and rotational motions of the EE. However, a strict global evaluation of performance is not obtained because the whole Jacobian matrix is not considered.

In contrast to the above-mentioned approaches for homogenization of the Jacobian matrix, the procedure proposed in [17] yields performance indices with a clear physical meaning. In such a reference the homogenized Jacobian matrix, denoted by $\boldsymbol{J^*}$, is obtained based on the use of the inertial parameters of the EE. Such a matrix allows to compute a 6-dimensional precursor vector of the kinetic energy of the EE. Consequently, $\boldsymbol{J^*}$ was employed to calculate performance indices that measure the ability of a 6 degrees of freedom (DOF) serial manipulator for converting joint-rates into kinetic energy of the EE. This approach was extended for application to a parallel 3-RRR robot [18].

A study is presented in this paper for comparing the effectiveness of two indices of performance to assess the ability to convert joint-rates of a robotic manipulator into kinetic energy of its EE. The applied indices of performance are based on the homogenized Jacobian matrix of the manipulator, and the study takes into account two approaches existing in the literature for homogenizing it. The former, that applies the notion of the *characteristic length* of the manipulator [4], and the second one that uses the inertial parameters of an ideal EE [17]. The manipulator considered is the *Mitsubishi* model PA 10, mounted at the *Mechatronics and Control Laboratory* (LMyC) of the *Tecnológico Nacional de Mexico - Instituto Tecnológico de la Laguna (TecNM-ITLag)*. The robot is shown in Fig. 1. In this paper , one joint of the PA 10 is fixed to consider the manipulator as a non-redundant one. Then, its indices of performance are analyzed for a specific motion of the EE.

Fig. 1. Robotic manipulator Mitsubishi PA10 of the TecNM-ITLag.

In the next Section a theoretical background is presented. Two approaches for homogenizing the Jacobian matrix are briefly recalled. The third Section describes the indices of performance considered in this paper. The architecture of the manipulator studied and the desired motion of this one are defined in the fourth Section. The fifth Section formulates the problem of the optimal placement for the path of the EE, and the proposed case studies are developed. The final comments and conclusion of the paper are presented in sixth Section.

II. BACKGROUND

The relationship between the joint-rates, $\dot{\boldsymbol{q}}$, of a manipulator and the twist, $\boldsymbol{t}$, of the EE is determined by the Jacobian matrix, as shown in the following equation:

$$\boldsymbol{t} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{1}$$

where $\boldsymbol{J}(\boldsymbol{q})$ is the basic Jacobian matrix of the manipulator. For simplicity, this matrix will be written as $\boldsymbol{J}$. The twist of the EE is defined as:

$$\boldsymbol{t} \equiv \begin{bmatrix} \boldsymbol{v} \\ \boldsymbol{\omega} \end{bmatrix} \tag{2}$$

Vectors $\boldsymbol{v}$ [m/s] and $\boldsymbol{\omega}$ [rad/s] in this Equation are, respectively, the velocity of the tool-center point (TCP) and the angular velocity of the EE (s denotes seconds). $\boldsymbol{t}$ is a 6-dimensional array and $\dot{\boldsymbol{q}}$ is a n-dimensional array ($n \geq 6$), with n being the number of DOF of the manipulator.

The Jacobian matrix $\boldsymbol{J}$ is a $6 \times n$ array whose units of the first 3 lines are meters, and the last 3 lines are dimensionless. Thus, to homogenize this matrix by applying the approach in [12], the three first lines should be divided by the characteristic length of the robot. So, the entries of the obtained homogenized matrix are dimensionless. Note that the homogenized Jacobian matrix, denoted by $\bar{\boldsymbol{J}}$, can no longer be applied in (1) because doesn't have a physical meaning.

To apply the approach proposed in [17] it is considered that the EE of the robot is a sphere whose mass is m_s and the radius is r_s. When the robot moves, the kinetic energy of the EE can be computed by

$$k = \parallel \boldsymbol{k_{eT}} \parallel^2 + \parallel \boldsymbol{k_{eR}} \parallel^2 \tag{3}$$

where $\boldsymbol{k_{eT}}$ and $\boldsymbol{k_{eR}}$ are precursors vectors of kinetic energy associated to the translational and rotational motion of the end-effector, respectively, that are obtained from

$$\begin{bmatrix} \boldsymbol{k_{eT}} \\ \boldsymbol{k_{eR}} \end{bmatrix} = \boldsymbol{J}^* \dot{\boldsymbol{q}} \tag{4}$$

where $\boldsymbol{J}^*$ is a matrix defined as

$$\boldsymbol{J}^* \equiv \begin{bmatrix} \lambda_T \boldsymbol{J}_{TA} & \lambda_T \boldsymbol{J}_{TW} \\ \lambda_R \boldsymbol{J}_{RA} & \lambda_R \boldsymbol{J}_{RW} \end{bmatrix} \tag{5}$$

The submatrices $\boldsymbol{J}_{TA}, \boldsymbol{J}_{TW}, \boldsymbol{J}_{RA}$ and $\boldsymbol{J}_{TW}$ in (5) are the 3×3 partitions of $\boldsymbol{J}$ located as follows:

$$\boldsymbol{J} = \begin{bmatrix} \boldsymbol{J}_{TA} & \boldsymbol{J}_{TW} \\ \boldsymbol{J}_{RA} & \boldsymbol{J}_{RW} \end{bmatrix} \tag{6}$$

The scalars λ_T and λ_R in (5) are inertial parameters of the spheric end-effector related to the translational and rotational motions, respectively, which are determined as

$$\lambda_T = \sqrt{\frac{m_s}{2}} \qquad \lambda_R = \sqrt{\frac{2 m_s r_s^2}{5}} \tag{7}$$

Therefore, $\boldsymbol{J}^*$ is a homogenized Jacobian matrix, whose units of all its entries are $\mathrm{kg}^{\frac{1}{2}} \cdot \mathrm{m}$ (kg denotes kilograms and m designates meters in SI units). Clearly, through (4), $\boldsymbol{J}^*$ determine the precursors vectors of kinetic energy of the EE.

III. PERFORMANCE OF MANIPULATORS

The index of manipulability w of a manipulator [1] can be computed by

$$w = \sqrt{det(\boldsymbol{J}\boldsymbol{J}^T)} \tag{8}$$

where $det(\boldsymbol{J}\boldsymbol{J}^T)$ is the determinant of the product of matrices $\boldsymbol{J}$ and $\boldsymbol{J}^T$. $\boldsymbol{J}$ is the Jacobian matrix of the kinematic chain. In most of industrial robots, the architecture of a robot is such that $\boldsymbol{J}$ includes terms with dimensions to compute both, linear and angular velocities of the EE, as considered in Section 2. Consequently, we cannot use the manipulability index in the form of (8) for simultaneously evaluating the ease to obtain translational and rotational motions of the EE. As an alternative, the extended notions of the manipulability, the translational and the rotational manipulabilities [2], can be applied. For a six DOF manipulator, the translational manipulability w_A of the arm (the three first joints and links) and the rotational manipulability w_W of the wrist (the three last joints and links) are given, respectively, by:

$$w_A = \sqrt{det\left(\boldsymbol{J_{TA}}\boldsymbol{J_{TA}}^T\right)} \tag{9}$$

$$w_W = \sqrt{det\left(\boldsymbol{J_{RW}}\boldsymbol{J_{RW}}^T\right)} \tag{10}$$

$\boldsymbol{J_{TA}}$ and $\boldsymbol{J_{RW}}$ in the last two equations are the partitions of $\boldsymbol{J}$ shown in (6). The translational and the rotational manipulabilities of a manipulator can be used to assess the efficacy to convert join-rates in translational or rotational velocities, respectively, of the EE.

It is observed that using the matrix $\boldsymbol{J}^*$ in (8), instead of $\boldsymbol{J}$, we evaluate the ability of the whole kinematic chain of the manipulator to convert joint-rates into kinetic energy of the EE.

On the other hand, the Kinetostatic Conditioning Index (KCI) of a manipulator is defined [4] as

$$\text{KCI} = \left(\frac{1}{\kappa}\right) \times 100 \tag{11}$$

where κ is the condition number of $\boldsymbol{J}$, which can be obtained dividing the largest singular value σ_1 of $\boldsymbol{J}$ by the smallest one σ_s. When the entries of $\boldsymbol{J}$ have different units, the the condition number of the homogenized matrix $\bar{J}$ can be used to compute the condition number. In [4] the *characteristic length* is defined in such a way that the condition number of the homogenized Jacobian matrix is minimized with the optimum posture of the manipulator. Thus, the condition of isotropy of $\bar{\boldsymbol{J}}$ has been used as a criterion for designing manipulators [11]. It is observed that such a condition holds when $\kappa = 1$, i.e., when $\sigma_l = \sigma_s$. This is the smallest value that can have κ with the optimum posture of a manipulator whose geometric parameters have been properly stablished under the criterion of the isotropy. If the criterion of isotropy is employed in designing, then the manipulator is referred to as an *isotropic* one. However, it must be observed that $\bar{\boldsymbol{J}}$ can no longer be applied in (1) because it hasn't the original units. Moreover, $\bar{\boldsymbol{J}}$ has not units and then its condition number doesn't have a clear physical meaning. Nevertheless, we will use the matrix $\bar{\boldsymbol{J}}$ to evaluate the KCI, as proposed in [4].

IV. DESCRIPTION OF THE MANIPULATOR AND DESIRED TASK

The manipulator studied in this work is the model PA10, from Mitsubishi, which is shown in Fig. 1. The kinematic scheme of this manipulator can be seen in Fig. 2. In such a figure the orthogonal frames assigned to each link are also observed. The Table I provides the geometric parameters based on the modified Denavit and Hartenberg convention (MDH). The direct and inverse kinematic position models and the basic jacobian matrix $\boldsymbol{J}$ are obtained through the SYMORO (SYmbolic MOdeling of RObots) software package [20].

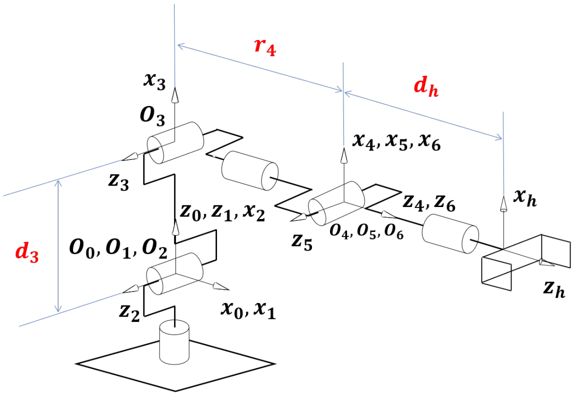


Fig. 2. Kinematic scheme of the robotic manipulator Mitsubishi PA-10 (with $d_h = 0.154$ m). The third real joint is considered as a fixed one such that the manipulator be non-redundant. This joint is not shown on this figure.

TABLE I
GEOMETRIC PARAMETERS OF THE MANIPULATOR.

$\boldsymbol{j}$	α_j	d_j	θ_j	r_j
1	0	0	θ_1	0
2	90°	0	θ_2	0
3	0	d_3	θ_2	0
4	90°	d_4	θ_2	r_4
5	−90°	0	θ_2	0
6	90°	0	θ_2	0

$d_3 = 225$ mm, $d_4 = 10$ mm, $r_4 = 250$ mm.

To carry out an effective comparison of the approaches studied here, a reference task is specified into the workspace of the manipulator. The desired path of the EE is a straight line that should be achieved by following a cycloidal motion in a total time T of 5 sec. The coordinates of the origin O_h of the frame Σ_h, attached to the EE, are x_{Oh}, y_{Oh}, z_{Oh}, with respect to the frame Σ_p, attached to the task. Σ_p is located with respect to the reference frame Σ_e of the workstation. The motion equations of the EE are given by (12) to (15):

$$x_{0h} = x_{0h_{ini}} + \delta x_{0h} f_c(t) \tag{12}$$

$$y_{0h} = 0 \tag{13}$$

$$z_{0h} = -0.154 \text{ m} \tag{14}$$

where the cycloidal function is given by:

$$f_c = \frac{t}{T} - \frac{1}{2\pi}\sin\left(\frac{2\pi t}{T}\right) \qquad for \quad 0 \leq t \leq T \tag{15}$$

Furthermore:

$$x_{0h_{ini}} = -0.10 \text{ m} \qquad \text{and} \qquad \delta_{x_{0h}} = 0.40 \text{ m} \tag{16}$$

The frames Σ_e, attached to the workstation, Σ_p and Σ_h are shown in Fig. 3. During the full displacement of the EE, the orientation of the frame Σ_h must keep the values of $\alpha_h = 90°$,$\beta_h = 0°$,$\gamma = 180°$. These ones are the Euler angles $z-y-x$ of the frame Σ_h, with respect to Σ_p. On the other hand, the placement of the frame Σ_p with respect to Σ_e is given by

$${}^{e}_{p}T = \begin{bmatrix} -c\beta_p & 0 & -s\beta_p & x_{Op} \\ 0 & -1 & 0 & y_{Op} \\ -s\beta_p & 0 & c\beta_p & z_{Op} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{17}$$

where s and c denote the functions $sine$ and $cosine$, respectively, β_p is the angle that should rotate Σ_e with respect to y_e , after a rotation of an angle $\alpha_p = 180°$ with respect to z_e in order to obtain the desired orientation of Σ_p. The third angle of Euler, γ_p, will be specified as $0°$. This angle corresponds to a rotation with respect to x_e after the precedent two rotations.

On the other hand, the coordinates x_{Op} ,y_{Op} and z_{Op} are those that specifiy the position of the origin O_p, while α_p , β_p and γ_p specify the orientation, of the frame Σ_p with respect to Σ_e. In the formulation presented here, y_{Op}, α_p and γ_p are specified as 0, 180° and 0°, respectively. So, the coordinates x_{Op}, z_{Op} and β_p will be considered as the independent variables in the optimization process of the placement of the path into the workspace of the manipulator. Such variables will be computed as described in the next Section such that some index of performance be optimized by a sample of postures of the manipulator during the achievement of the task.

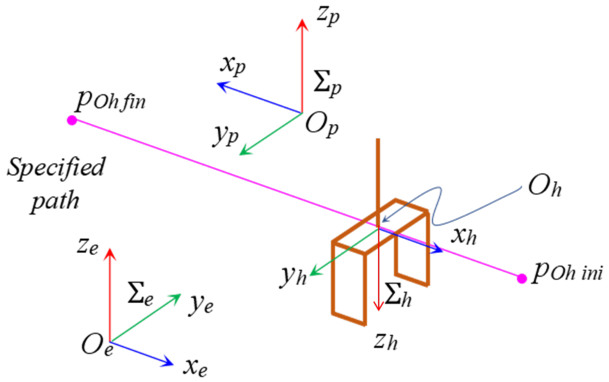


Fig. 3. Desired path of the EE

V. OPTIMAL PLACEMENT OF THE TASK. CASE STUDIES

We want to complete a comparison of performances of the manipulator applying the different homogenized matrices with both indices, the manipulability and the KCI. This comparison must be accomplished considering postures of the robot along the whole path. So, we must find the optimal placement of the path into the workspace of the manipulator for each study case. Then, the curves giving the behavior of the indices at each case will be compared.

The problem of the optimal placement of the task is stated as follows:

To find the placement $\boldsymbol{x}_e = [\beta_p, x_{Op}, z_{Op}]^T$ of the frame task Σ_p that minimize the function

$$f_{obj} = -\left(\bar{\phi} - \phi_\sigma\right) \tag{18}$$

subject to

$$90° \leq \beta_p \leq 270° \tag{19a}$$

$$-0.05 \text{ m} \leq x_p \leq 0.60 \text{ m} \tag{19b}$$

$$0 \text{ m} \leq z_p \leq 1.0 \text{ m} \tag{19c}$$

In (18) $\bar{\phi}$ is the mean value of a sample of n_s values of the index of performance ϕ to be optimized along the path. For this study, the size of the sample is defined as $n_s = 50$. The scalar ϕ_σ is the standard deviation of the sample. Depending on the case study, the index of performance ϕ could be the manipulability or the KCI of the considered Jacobian matrix. Likewise, this matrix, could be $\boldsymbol{J}^*$ or $\bar{\boldsymbol{J}}$ depending on the case study. The function (19) was proposed in [21] to find the optimal placement of manipulators in executing specified tasks into obstructed environments. In such a reference an algorithm was also proposed to solve the placement problem as previoulsy stated. The solutions corresponding to each study case are presented in the following paragraphs.

A. Case studies

A set of case studies is considered now to compare the effectiveness of both indices the manipulability and the KCI, when they are evaluated by using the homogenized Jacobian matrices $\boldsymbol{J}^*$ or $\bar{\boldsymbol{J}}$ described in Section 2.

Before analyzing the study cases, the optimal postures of the manipulator are presented in figures 4 and 5, as reference postures. Likewise, the values of both indices, the manipulability (w) and the KCI, corresponding to these postures are displayed in Table II. The values of other indices, supplementary to that one employed as objective function, were computed and they are also displayed at each column of this Table. It is observed that the posture obtained by optimizing $w(\bar{\boldsymbol{J}})$ is identical to that one computed by optimizing $w(\boldsymbol{J}^*)$. This posture is shown in Fig. 4a. Similarly, the posture determined by optimizing $KCI(\bar{\boldsymbol{J}})$ is the same than that one obtained by optimizing $KCI(\boldsymbol{J}^*)$. Such a posture is displayed in Fig. 4b.

The considered study cases are described as follows:

Case 1. Arbitrary placement of the task, such that the posture at the middle of the path is the optimal one with the manipulability of $\boldsymbol{J}^*$, and the path being horizontal. This placement will be considered as the initial one for the processes of optimization of placement in the other cases.
Case 2. Optimal placement of the path for maximum $w(\boldsymbol{J}^*)$.
Case 3. Optimal placement of the path for maximum $w(\bar{\boldsymbol{J}})$.
Case 4. Optimal placement of the path for maximum $KCI(\boldsymbol{J}^*)$.
Case 5. Optimal placement of the path for maximum $KCI(\bar{\boldsymbol{J}})$.

The values of the independent variables corresponding to the initial and the optimal placement at each case are summarized in Table III. The numerical results for the objective function are presented in Table IV, and graphics that show some postures employed during the task and the behavior of the indices of performance are presented in figures 5 to 19.

TABLE II
VALUES OF THE INDICES OF PERFORMANCE CORRESPONDING TO THE OPTIMAL POSTURE

	Values of indices for the optimal posture			
Objective function	$w(\boldsymbol{J}^*)$	$w(\boldsymbol{J})$	$KCI(\boldsymbol{J}^*)$	$KCI(\boldsymbol{J})$
$w(\boldsymbol{J}^*)$	0.0049	3.7278	16.3739	20.8951
$w(\bar{\boldsymbol{J}})$	0.0049	3.7279	16.3391	20.8482
$KCI(\boldsymbol{J}^*)$	0.0028	2.1174	25.9946	31.7813
$KCI(\bar{\boldsymbol{J}})$	0.0028	2.1421	25.9952	31.7837

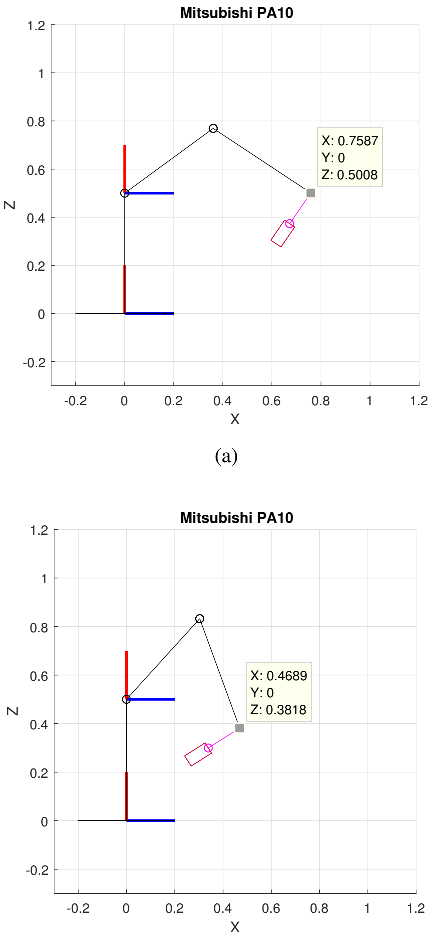
TABLE III
RESULTS OF OPTIMIZATION. INDEPENDENT VARIABLES

	Optimal placement of Σ_p		
Case	β_p (deg)	x_{Op} (m)	z_{Op} (m)
1*	180	0.7500	0.5000
2	237.33	0.5718	0.3416
3	237.33	0.5718	0.3416
4	270.00	0.2733	0.2334
5	251.35	0.2830	0.2531

*Initial placement employed in the process of optimization for all the other cases.

TABLE IV
RESULTS OF OPTIMIZATION. OBJECTIVE FUNCTION AND MAXIMUM VALUE OBTAINED FOR EACH INDEX OF PERFORMANCE AT SOME PATH POINT.

	Case 1	Case 2	Case 3	Case 4	Case 5
$\lvert f_{obj} \rvert$	0.0016	0.0042	3.180	22.8481	21.8922
$w(\boldsymbol{J}^*)$	0.0042	0.0047	0.0047	0.0029	0.0033
$w(\bar{\boldsymbol{J}})$	3.180	3.5837	3.5837	2.2156	2.5093
$KCI(\boldsymbol{J}^*)$	22.8481	22.6920	22.6920	23.3444	24.8876
$KCI(\bar{\boldsymbol{J}})$	28.416	28.0862	28.0862	29.9123	31.0117



(a)

(b)

Fig. 4. Side views of optimal postures of the manipulator (a) for maximum values of $w(\boldsymbol{J}^*)$ and $w(\bar{\boldsymbol{J}})$, (b) for maximum values of $KCI(\boldsymbol{J}^*)$ and $KCI(\bar{\boldsymbol{J}})$.

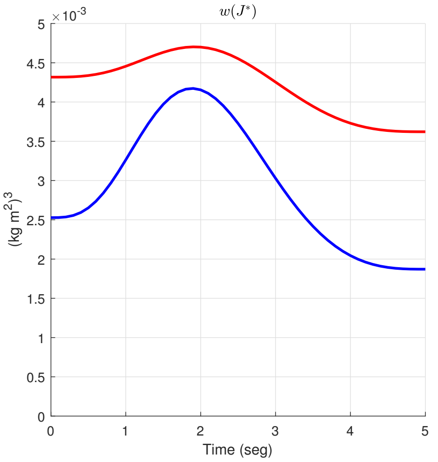


Fig. 5. Behavior of indices of performance. The red line represents $w(J^*)$ in the case 2. The blue line represents $w(J^*)$ in case 1.

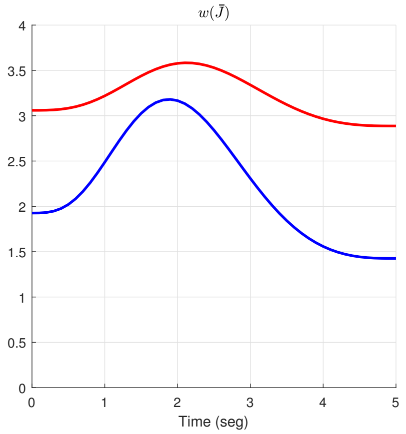


Fig. 6. Behavior of indices of performance. The red line represents $w(\bar{J})$ in the case 3. The blue line represents $w(\bar{J})$ in case 1.

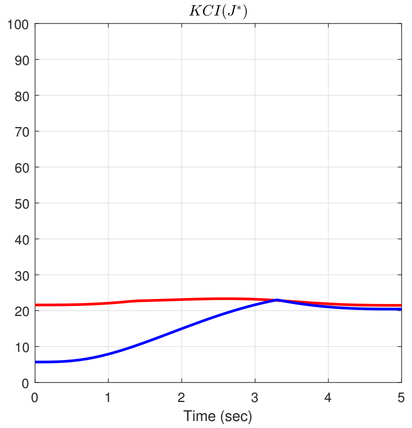


Fig. 7. Behavior of indices of performance. The red line represents $KCI(J^*)$ in the case 4. The blue line represents $KCI(J^*)$ in case 1.

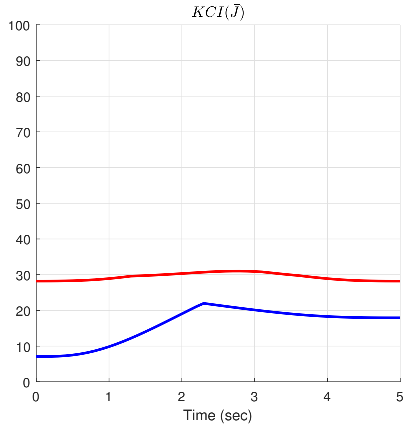
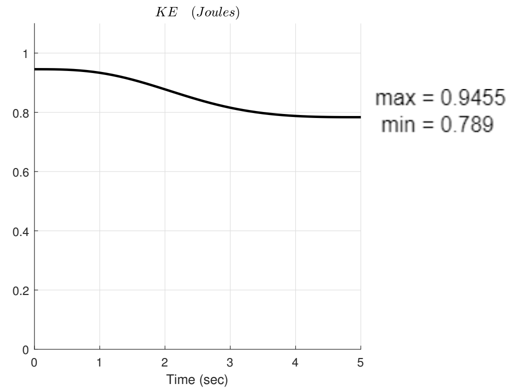
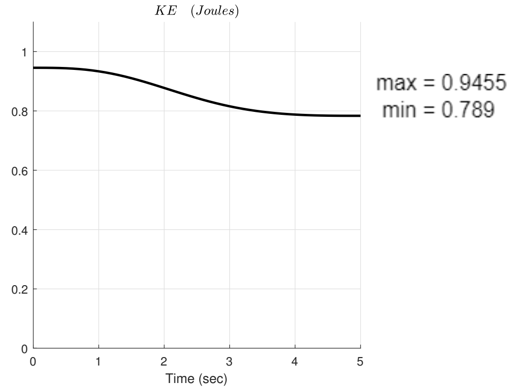


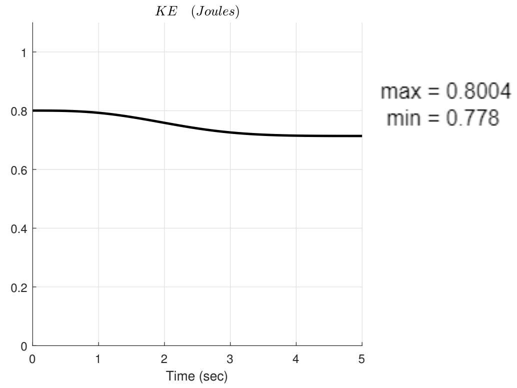
Fig. 8. Behavior of indices of performance. The red line represents $KCI(\bar{J})$ in the case 5. The blue line represents $KCI(\bar{J})$ in case 1.



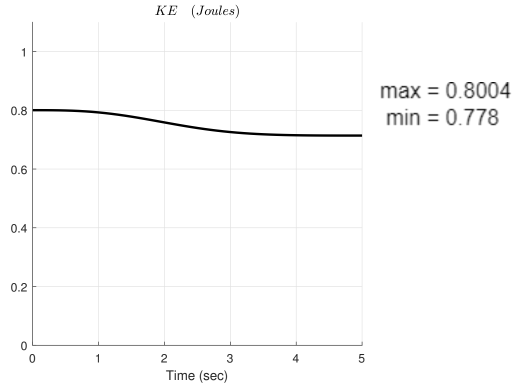
(a) Case 2



(b) Case 3



(c) Case 4



(d) Case 5

Fig. 9. Maximum kinetic energy of the EE for postures of the task in each case with joint-rates that hold the condition $\| \dot{\boldsymbol{q}} \| = 1$

The obtained results show that the features that are measured by employing the two considered indices of performance are quite different. The manipulability privilege the magnitude itself of the kinetic energy of the EE, while the KCI is focused to get uniformity of magnitudes of the such energy. Indeed, it is observed in Fig. 9 that optimizing the manipulability we obtain bigger values of the maximum kinetic energy compared with those obtained by optimizing the KCI. In contrast, we note in figures 9 (c) and (d) that the difference beteween the two extremes of the curves is smaller than the same difference in figures 9 (a) and (b). The figures 9 (c) and (d) were obtained by optimizing the KCI.

VI. CONCLUSION

In this study it was shown that the performance measured by the index of manipulability evaluated by considering both matrices, $\boldsymbol{J}^*$ and $\bar{\boldsymbol{J}}$, yield the same optimal posture and identical optimal placement of the path. A similar observation is made for the KCI, when evaluated by $\boldsymbol{J}^*$ and $\bar{\boldsymbol{J}}$. However, the physical meaning of each index by employing $\boldsymbol{J}^*$ is quite different than the physical meaning of the same index by applying $\bar{\boldsymbol{J}}$. Indeed, taking into account Eq. (4), the manipulability evaluated with $\boldsymbol{J}^*$ assess specifically the conversion of joint-rates into kinetic energy of the manipulator's EE for all direction of both, translational and rotational, motions. In contrast with $\boldsymbol{J}^*$, the "manipulability" evaluated with $\bar{\boldsymbol{J}}$ doesn't have a physical meaning. Though, thanks to the results of the present work, now we can interpretate such a manipulability as the ease to convert joint-rates into translational and rotational motions of the EE.

On the other hand, the physical meaning of the KCI evaluated with $\boldsymbol{J}^*$ can be reasoned also based on Eq. (4). Indeed, since small values of the condition number of a matrix is related to the uniformity of its singular values, then large values of the KCI of $\boldsymbol{J}^*$ in Eq. (4) can be interpreted as a good uniformity of the norm of precursor vectors of the kinetic energy. To have nearly the same ability for displacing the EE in all directions, is a relevant feature of a posture of a manipulator. For certain applications, this feature could be preferred compared to the ability to displace more quickly the EE on certain specific direction.

Unlike the KCI evaluated with $\boldsymbol{J}^*$, the KCI evaluated with $\bar{\boldsymbol{J}}$ doesn't have a physical meaning because$\bar{\boldsymbol{J}}$ is dimensionless. Nevertheless, thanks to the results obtained here, we can interpretate the KCI based on $\bar{\boldsymbol{J}}$ as the ease to convert joint-rates into motions of the EE with nearly uniform levels of kinetic energy for rotational and translational motions in all directions. This feature has not been pointed out in previous works related to the subject. Specifically, the manipulability based on J* allows to measure the ability to move the end-effector with kinetic energy associated to both displacements traslational and rotational simultaneously. This feature of the global manipulability based on J* is applied for the first time to evaluate the performance of a serial robotic manipulator during the achievement of a task.

VII. ACKNOWLEDGMENTS

The work reported in this paper was sponsored by the TecNM-ITLag and the CONACyT of Mexico. Thanks to these agencies for their support.

REFERENCES

[1] T. Yoshikawa, "*Manipulability of robotic mechanisms*", International Journal of Robotics Research, Vol. 4, No. 2, pp. 3-9, 1985.

[2] T. Yoshikawa, "*Translational and Rotational Manipulability of Robotic Manipulators*", Proceedings of the American Control Conference, pp. 228-233, 1990.

[3] J.K. Salisbury, J.J. Craig, "*Articulated hands: force control and kinematic issues*", International Journal of Robotics Research, Vol. 1, No. 1, pp. 4-17, 1982.

[4] J. Angeles, CS. López-Cajún, "*Kinematic Isotropy and the Conditioning Index of Serial Robotic Manipulators*", The international Journal of Robotics Research, 1992, pp. 560-571

[5] R. Stoughton, T. Kokkins, "*Some properties of a new kinematic structure for robot manipulators*", Proc. 13th ASME Design Automation Conference, 73-79, Boston, 1987.

[6] F. Xie, X.-J. Liu, J. Wang "*Performance Evaluation of Redundant Parallel Manipulators Assimilating Motion/Force Transmissibility*", International journal of Advanced Robotic Systems, Vol. 8, No. 5, pp. 113-124, 2011.

[7] C. Chen, J. Angeles, "*Generalized transmission index and transmission quality for spatial linkages*", Mechanism and Machine Theory, Vol. 42, pp. 1225-1237, 2007

[8] J. Wang, C. Wu, X.-J. Liu, "*Performance evaluation for parallel manipulators: Motion/force transmissibility and its index*", Mechanism and Machine Theory, Vol. 45, pp. 1462-1476, 2010.

[9] I. Mansouri, M. Ouali, "*A new homogeneous manipulability measure of robot manipulators, based on power concept*", Mechatronics, Vol. 19, No. 6, pp. 927–944, 2009.

[10] Y. Nakamura, H. Hanafusa, T. Yoshikawa, "*Task-priority based redundancy control of robot manipulators*", Int. J. Robot. Res. 6, pp. 3–15, 1987.

[11] O. Ma, J. Angeles, "Optimum Architecture Design of Platform Manipulators", Proc. IEEE Int. Conf. Advanced Robotics, 1991.

[12] J. Angeles, F. Ranjbaran, R.V. Patel, "*On the design of the kinematic structure of seven axes redundant manipulators for maximum conditioning*", IEEE International Conference on Robotics and Automation, Nice, France, 1992.

[13] C. M. Gosselin, "*The optimum design of robotic manipulators using dexterity indices*", Robotics and Autonomous Systems, Vol. 9, No. 4, pp. 213-226, 1992.

[14] F. C. Park, J. W. Kim, "*Manipulability of closed kinematic chains*", ASME J. Mechanical Design. Vol.120. No.4. 1998.

[15] L. Stocco, S. E. Salcudean and F. Sassani, "*Matrix Normalization for Optimal Robot Design*", in Proc. of IEEE ICRA, Leuven, Belgium, May 16-21, 1998.

[16] D. Reveles,J.A. Pamanes, Ph. Wenger, "*Trajectory planning of kinematically redundant parallel manipulators by using multiple working modes*", Mech. Mach. Theory 98, pp. 216-230, 2016.

[17] Pamanes, J. A., and H. A. Moreno. "Homogenization of the Jacobian Matrix of Manipulators by Using Inertial Parameters". IFToMM Symposium on Mechanism Design for Robotics. Springer,2018.

[18] D. R. Soto, "*Esudio sobre la homogeneización de la matriz jacobiana de robots manipuladores*", Tesis de maestría, Instituto Tecnológico de la Laguna, Torreón, Coahuila de Zaragoza, México, 2019.

[19] Nabavi, S., Akbarzadeh, J. Enferadi, I. Kardan, *A homogeneous payload specific performance index for robot manipulators based on the kinetic energy.* Mechanism and Machine Theory 130, pp 330-345, 2018.

[20] W. Khalil W, Ph. Lemoine, "*SYMORO+ Symbolic Modeling of Robots: User Guide*" IRCCyN (*Institut de Recherche en Communication et Cybernétique de Nantes*), 2003.

[21] J.A. Pámanes, "*Contribution à l'étude de l'accessibilité aux tâches et à la détermination du placement optimal de robots manipulateurs*", PhD. Thesis, Université de Poitiers, France, 1992.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Análisis por Elemento Finito de un Robot Industrial de 6-GDL Bajo Condición de Falla

Salvador Martinez-Cruz[1]*, Gerardo U. Nieto-Sánchez[1], David de Jesús-Díaz, Luis A. Morales-Hernández[1]

***Resumen*—En este artículo se presenta el análisis por elemento finito de un robot manipulador KR-16 de 6 grados de libertad (GDL) mediante diferentes condiciones en las articulaciones; sin daño, nivel de daño moderado y alto nivel de daño mediante el software ANSYS© Workbench. Los resultados muestran que un robot cambia sus frecuencias naturales dependiendo el nivel de daño que tenga debido a que el coeficiente de fricción en las articulaciones aumenta cuando los engranes están dañados. La diferencia entre las frecuencias naturales del robot cuando es sano y con daño moderado es de 18%, por otra parte, entre un robot sano y un robot con daño severo, las frecuencias varían hasta un 68%.**

Palabras clave—robot manipulador, análisis por elemento finito, falla en engranajes.

I. INTRODUCCIÓN

Los robots manipuladores se han usado ampliamente en las últimas décadas debido a su precisión, alto nivel de productividad y eficiencia en la industria. Sus aplicaciones se encuentran en el área de manufactura para soldadura, ensamble de piezas, clasificadores de productos sanos y dañados, entre otros. Por ejemplo, Luis Pérez *et al.* [1] presentaron un trabajo con robots industriales donde se hace una sinergia entre la realidad virtual y la robótica, basada en el uso de tecnologías de videojuegos comerciales para crear un ambiente totalmente inmerso basado en realidad virtual. Además, Zhang *et al.* [2] propusieron un nuevo marco de detección de fuerzas externas para robots blandos basado en la fusión de mediciones basadas en la visión y técnicas de modelos de elementos finitos (FEM). Otra aplicación de los robots manipuladores es la que presentan Nikolaos *et al.* [3] quienes analizaron un sistema ciber-físico para permitir y controlar operaciones seguras de ensamblaje colaborativo entre humanos y robots. El enfoque considera un espacio de trabajo compartido sin vallas donde pueden operar humanos, robots industriales u otros objetos en movimiento, como vehículos auto guiados. Por su parte, Tingting Liu *et al.* [4] desarrollaron un algoritmo de mejora de resolución con restricciones de variación total (TV) para el espectro IR (FTIR) de transformada de Fourier degradado debido a la superposición y degradación del ruido en la detección de visión del robot. Aumentando la precisión de detección de visión del robot en sistemas industriales inteligentes. Otro trabajo similar es el que presentaron Munawar *et al.* [5], donde se desarrolla un sistema de monitoreo para robots industriales mediante una cámara monocular. Además, proponen un nuevo método de aprendizaje no supervisado para entrenar un extractor de características profundas a partir de imágenes sin etiquetas.

El análisis por elemento finito en robots manipuladores es muy útil para conocer el comportamiento del robot bajo diversas condiciones. Ortiz *et al.* [6] presentaron un sistema de detección de fallas en un Generador Eléctrico Doblemente Alimentado (DFIG) usando un análisis por elemento finito para aplicaciones en turbinas eólicas. Por su parte, Arora y Dhami [7]. Presentaron el diseño de un robot industrial 6-GLD junto con los diferentes componentes de transmisión como la caja de engranajes, el análisis por elemento finito y multi-cuerpo de un robot industrial completo. Un trabajo similar es el de Sahu y Choudhury [8], quienes presentaron una investigación sobre deformación, esfuerzo cortante, deformación cortante y energía de deformación, basados en el modelo 3D y el análisis estructural por medio del método de elemento finito para aumentar la productividad y la calidad del entorno de trabajo de alta tecnología en un robot industrial. Por otra parte, Raza *et al.* [9] investigaron la viabilidad de integrar diferentes herramientas como el modelado en CAD y el análisis por elemento finito (FEA) para obtener un diseño óptimo de un brazo de robot industrial, y de esta forma, incrementar la tasa de producción. Además, Supriya y Choudhury [10]. Usaron el método de elemento finito para conocer la deformación, las tensiones, la deformación elástica de corte y la energía de deformación en diferentes puntos de un robot industrial, y de esta manera, determinar su factor de seguridad. Por su parte, Sahu *et al.* [11] emplearon el método de elemento finito para analizar los diferentes modos de frecuencias de un robot industrial con fisura y sin fisura, con el fin de descubrir las partes débiles del brazo del robot y finalmente, poder mejorar su eficiencia.

El análisis por elemento finito en el software ANSYS© Workbench para un robot manipulador del tipo industrial se puede encontrar en la literatura con diversos propósitos. Supriya y Choudhury [12] Presentaron un trabajo con robots industriales de 6-GLD para determinar el esfuerzo cortante máximo, las frecuencias naturales y las deformaciones modales mediante un análisis por elemento finito en ANSYS© Workbench. Otro trabajo similar es el de Tang *et al.* [13], quienes realizaron un análisis modal en ANSYS del robot industrial, obteniendo la frecuencia de vibración natural y el efecto de respuesta de los eslabones bajo carga para poder evitar malas condiciones de trabajo causadas por la resonancia, además de tener un valor referenciado para diseñar y mejorar la estructura del brazo robótico. Los trabajos anteriormente mencionados han demostrado que el análisis por elemento finito de robots industriales es de gran ayuda para conocer el comportamiento del robot bajo

[1]Facultad de Ingeniería, Campus San Juan del Río, Universidad Autónoma de Querétaro, San Juan del Río, Querétaro 76807, México; salvador.martinez.cruz@uaq.mx, gnieto11@alumnos.uaq.mx, davodejesus@outlook.com, luis.morales@uaq.mx
*Correspondence author

diversas condiciones.

En este artículo se presenta el análisis por elemento finito de un robot manipulador de 6-GDL bajo diferentes condiciones de falla. Los resultados muestran que las frecuencias naturales del robot varían en un 60% cuando tiene un nivel moderado de daño y cambian hasta un 78% cuando el nivel de daño es alto.

II. DESCRIPCIÓN DEL ROBOT

El modelo en 3D del robot usado en este artículo se muestra en la Figura 1. Este robot es el KR-16.

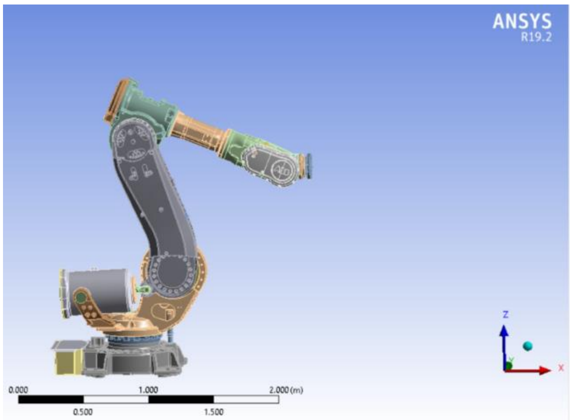


Fig. 1. Modelo en 3D del Robot KR-16.

La Tabla I muestra las características del Robot KR-16.

TABLA I. CARACTERÍSTICAS DEL ROBOT KR-16

Empresa	KUKA©	
Modelo	KR-16	
Carga útil	16 kg	
Número de Ejes	6	
Datos de los Ejes	Campo (Software)	Velocidad
Eje 1 (A 1) Eje 2 (A 2) Eje 3 (A 3) Eje 4 (A 4) Eje 5 (A 5) Eje 6 (A 6)	± **185°** $^{2)}$ + 35°/- 155° +154°/-130° ±350° **±130°** ±350°	156°/s 156°/s 156°/s 330°/s 330°/s 615°/s

En la Figura 2 se muestran las dimensiones del robot en mm.

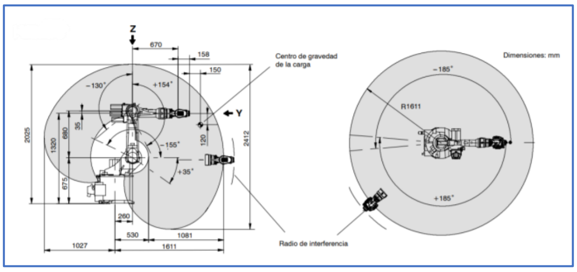


Fig. 2. Diagrama de dimensiones en milímetros (mm).

En la Figura 3 se observan las articulaciones del robot usado en el presente trabajo,

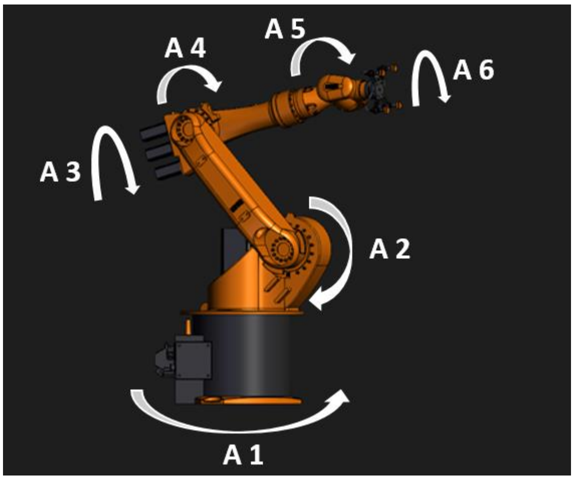
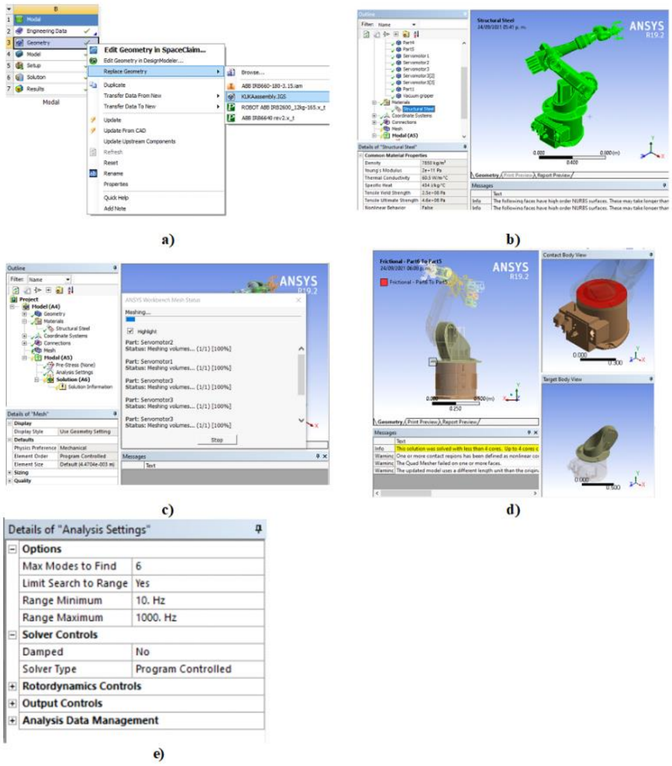


Fig. 3. Diagrama que muestra los 6 ejes del robot descritos anteriormente.

III. ANÁLISIS POR ELEMENTO FINITO

En esta sección se presenta el análisis por elemento finito del robot en el software ANSYS© Workbench.



Details of "Analysis Settings"

Options	
Max Modes to Find	6
Limit Search to Range	Yes
Range Minimum	10. Hz
Range Maximum	1000. Hz
Solver Controls	
Damped	No
Solver Type	Program Controlled
Rotordynamics Controls	
Output Controls	
Analysis Data Management	

Fig. 4. Análisis en Software ANSYS©: **a)** Selección del archivo fuente. **b)** Asignación de materiales. **c)** Aplicación de mallado. **d)** Asignación de coeficientes de fricción. **e)** Parámetros de análisis.

Como se observa en la Figura 4, se selecciona el archivo fuente que en este caso es un modelo 3D del robot KR-16, posteriormente se realizó la asignación de los materiales a las diferentes partes del robot, en este caso el material implementado fue el acero estructural. Para realizar este análisis se aplicó un mallado el cual está compuesto por diversos métodos de mallado que contiene el software, tales como; *hex-dominant, multizone* y *automatic*. Además, se fijó la base del robot al suelo mediante un método llamado *Fixed support*. Como siguiente paso, se procede a la asignación de los coeficientes de fricción en las zonas de contacto o rozamiento del robot para finalmente, configurar los parámetros de análisis modal, tales como la configuración del rango de frecuencias en las que se desea encontrar un número definido de frecuencias naturales.

En la Figura 5, se puede apreciar el resultado del mallado aplicado.

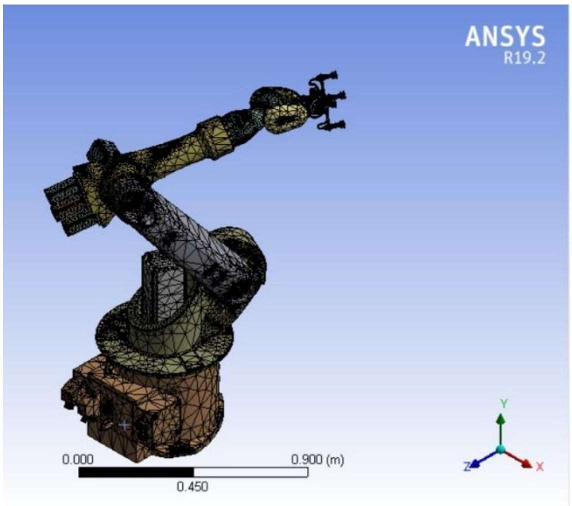


Fig. 5. Mallado usado en robot.

Una vez que se realizó el proceso de mallado se configuran los parámetros para el análisis modal, en este caso se miden las primeras seis frecuencias naturales del robot en un rango de 10 Hz a 1000 Hz dado que en este rango es donde se encuentran las frecuencias de falla por desgaste en los engranes.

La Tabla II muestra los coeficientes de fricción usados en las articulaciones para los diferentes estados del robot.

TABLA II. PARÁMETROS DEL ANÁLISIS

Coeficientes de fricción en las articulaciones		
Robot sano	***Daño moderado***	***Daño severo***
0.01	0.5	0.99

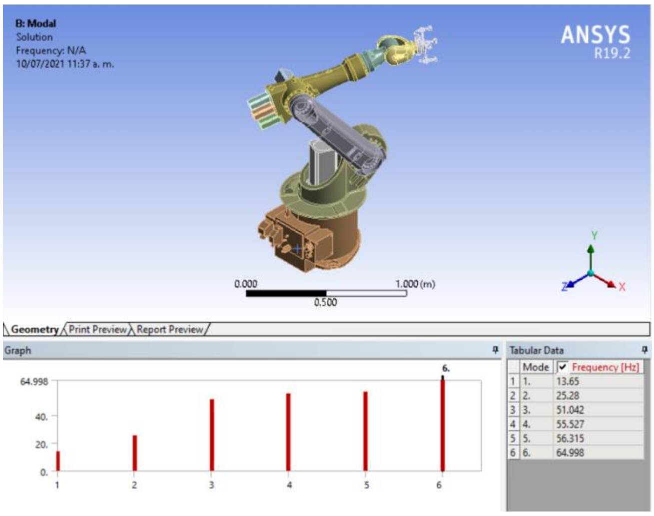
Los valores de fricción rotacional mostrados en la Tabla II se proponen considerando que el daño severo en las articulaciones del robot impide por completo el movimiento del eje y que cuando no hay daño el este coeficiente es prácticamente cero. Como se muestra en la Tabla II, un robot sano tiene un coeficiente de fricción rotacional normalizada bajo en las articulaciones dado que no hay desgaste en los engranes, y este coeficiente aumenta con respecto al nivel de daño.

Se llevaron a cabo tres análisis en el robot con diferentes coeficientes de fricción en las articulaciones que corresponden a tres estados diferentes del robot; (1) motor sano, (2) motor con daño moderado, (3) motor con daño severo.

IV. RESULTADOS Y DISCUSIÓN

En esta sección se muestran los resultados obtenidos en el análisis modal bajo las tres condiciones diferentes.

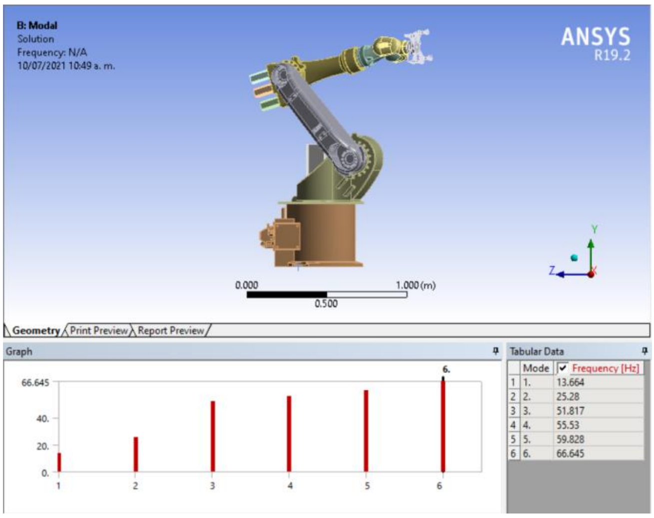
La Figura 5 muestra los resultados del análisis por elemento finito para el robot descrito bajo una condición de robot sano, esto significa que tiene un coeficiente de fricción muy bajo en las articulaciones (0.01), esto representa que los engranes del robot se encuentran en buen estado.


B: Modal
Solution
Frequency: N/A
10/07/2021 11:37 a. m.

	Mode	☑ Frequency [Hz]
1	1.	13.65
2	2.	25.28
3	3.	51.042
4	4.	55.527
5	5.	56.315
6	6.	64.998

Fig. 6. Resultados para robot sano.

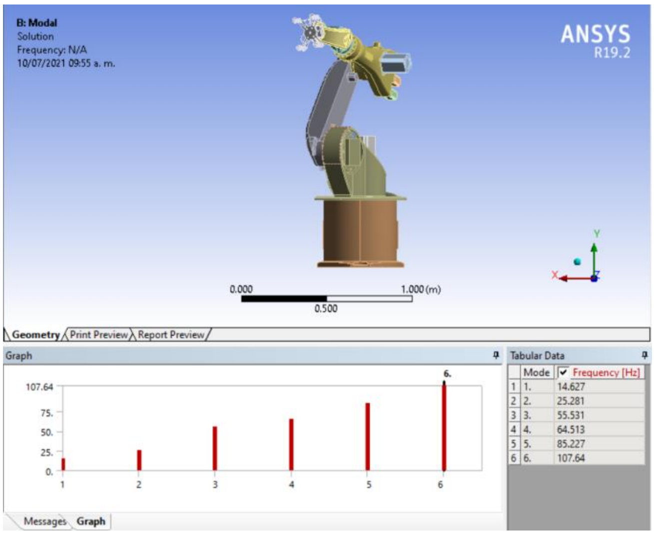
La Figura 6 muestra los resultados del análisis para el robot en condición de daño moderado, para llevar a cabo este análisis se configuró un coeficiente de fricción en las articulaciones de 0.5, lo cual representa que los engranes en las articulaciones están moderadamente degradados.


B: Modal
Solution
Frequency: N/A
10/07/2021 10:49 a. m.

	Mode	☑ Frequency [Hz]
1	1.	13.664
2	2.	25.28
3	3.	51.817
4	4.	55.53
5	5.	59.828
6	6.	66.645

Fig. 7. Resultados para el robot con daño moderado.

Finalmente, la Figura 7 muestra los resultados obtenidos para el robot con daño alto, en este caso se usó un coeficiente de fricción de 0.99 en las articulaciones, lo que representa un daño severo y una alta dificultad de movimiento para el robot.



B: Modal
Solution
Frequency: N/A
10/07/2021 09:55 a. m.

	Mode	☑ Frequency [Hz]
1	1.	14.627
2	2.	25.281
3	3.	55.531
4	4.	64.513
5	5.	85.227
6	6.	107.64

Fig. 8. Resultados para el robot bajo condición de daño alto.

En la Figura 8 se muestra una gráfica de los resultados obtenidos en un contexto comparativo, donde se puede observar que, con respecto a las frecuencias naturales del sistema en una condición de robot sano, la respuesta con daño moderado no es significativamente distinta, mientras que con daño alto la diferencia es mucho más evidente.

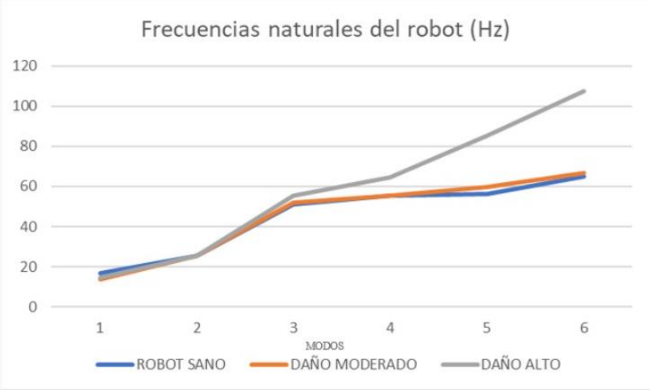


Fig. 9. Respuesta del robot bajo diferentes condiciones.

Para conocer la diferencia en la respuesta del sistema cuantitativamente se calcula el porcentaje de diferencia en cada articulación bajo las condiciones de daño moderado y daño alto con respecto al robot sano. La Figura 9 muestra el resultado de este análisis.

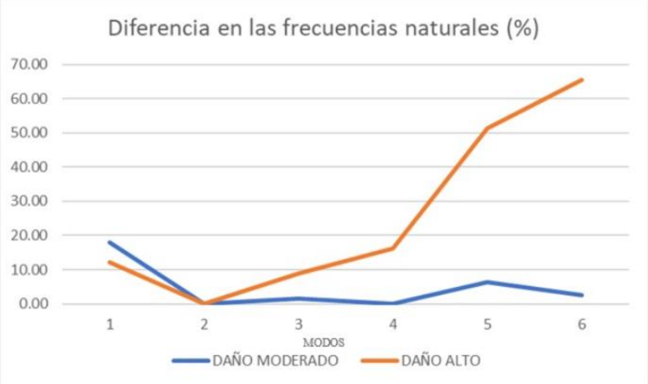


Fig. 10. Porcentaje de diferencia entre el análisis de daño moderado y daño alto con respecto al robot sano.

De la Figura 9 se observa que cuando el robot tiene un daño moderado las frecuencias naturales no varían más del 20% en la frecuencia 1, y en las otras 5 frecuencias principales no llega al 10%. Para el robot con daño severo esta situación es completamente distinta, en la frecuencia 1 tiene una diferencia aproximada al 12%, para la frecuencia 2 este valor prácticamente se vuelve 0%, pero a partir de la frecuencia 3 la diferencia aumenta llegando casi al 70% en la frecuencia 6.

V. CONCLUSIONES

En este artículo se presenta el análisis por elemento finito de un robot de 6 GDL mediante el software ANSYS© Workbench bajo diferentes condiciones de falla; robot sano, daño moderado, daño alto.

Los resultados muestran que las frecuencias naturales del sistema no cambian significativamente cuando el daño en las articulaciones es moderado con respecto al robot sano, ya que el porcentaje de diferencia más alto es de 18%. Mientras que, para el daño alto, la diferencia es significativa, va desde el 0% en la frecuencia 2, hasta el 68% en la frecuencia 6. Con lo que se concluye que los coeficientes de fricción provocados por el daño en los engranes del robot pueden modificar las frecuencias naturales del sistema a medida que el daño aumenta.

El análisis presentado en este artículo puede ser usado para la detección de fallas en el robot mediante un análisis de vibraciones comparado con las frecuencias naturales estimadas.

REFERENCES

[1] L. Pérez, E. Diez, R. Usamentiaga, and D. F. García, “Industrial robot control and operator training using virtual reality interfaces,” *Comput. Ind.*, vol. 109, pp. 114–120, 2019, doi: 10.1016/j.compind.2019.05.001.

[2] Z. Zhang, J. Dequidt, and C. Duriez, “Vision-Based Sensing of External Forces Acting on Soft Robots Using Finite Element Method,” *IEEE Robot. Autom. Lett.*, vol. 3, no. 3, pp. 1529–1536, 2018, doi: 10.1109/LRA.2018.2800781.

[3] N. Nikolakis, V. Maratos, and S. Makris, “A cyber physical system (CPS) approach for safe human-robot collaboration in a shared workplace,” *Robot. Comput. Integr. Manuf.*, vol. 56, no. June 2017, pp. 233–243, 2019, doi: 10.1016/j.rcim.2018.10.003.

[4] T. Liu, H. Liu, Y. F. Li, Z. Chen, Z. Zhang, and S. Liu, “Flexible FTIR spectral imaging enhancement for industrial robot infrared vision sensing,” *IEEE Trans. Ind. Informatics*, vol. 16, no. 1, pp. 544–554, 2020, doi: 10.1109/TII.2019.2934728.

[5] A. Munawar, P. Vinayavekhin, and G. De Magistris, “Spatio-Temporal anomaly detection for industrial robots through prediction in unsupervised feature space,” *Proc. - 2017 IEEE Winter Conf. Appl. Comput. Vision, WACV 2017*, pp. 1017–1025, 2017, doi: 10.1109/WACV.2017.118.

[6] R. Ortiz-Medina, R. Alvarez-Salas, L. Irvin, and V. M. Jim, “Detecci ´ on de Fallas en un DFIG empleando el M ´ etodo del Elemento Finito,” no. 2018, pp. 23–25, 2019.

[7] R. Arora and S. S. Dhami, “Finite element analysis and multibody dynamics of 6-DOF industrial robot,” *Int. J. Mech. Prod. Eng. Res. Dev.*, vol. 7, no. 5, pp. 1–12, 2017, doi:

10.24247/ijmperdoct20171.

[8] S. Sahu and B. B. Choudhury, *Dynamic behaviour analysis of an industrial robot using FEM*, vol. 758. Springer Singapore, 2018.

[9] K. Raza, T. A. Khan, and N. Abbas, “Kinematic analysis and geometrical improvement of an industrial robotic arm,” *J. King Saud Univ. - Eng. Sci.*, vol. 30, no. 3, pp. 218–223, 2018, doi: 10.1016/j.jksues.2018.03.005.

[10] Supriya Sahu and B.B. Choudhury, *Static Analysis of a 6-Axis Industrial Robot Using Finite Element Analysis*. International Journal of Mechanical Engineering and Technology, 8(3), 2017, pp. 49–55.

http://www.iaeme.com/IJMET/issues.asp?JType=IJMET&VType=8&IType=3

[11] S. Sahu, B. B. Choudhury, and B. B. Biswal, “A Vibration Analysis of a 6 Axis Industrial Robot Using FEA,” *Mater. Today Proc.*, vol. 4, no. 2, pp. 2403–2410, 2017, doi: 10.1016/j.matpr.2017.02.090.

[12] S. Sahu and B. B. Choudhury, *Stress and Modal Analysis of Six-Axis Articulated Robot Using ANSYS*, vol. 40. Springer Singapore, 2019.

[13] Y. Tang, Y. Yu, J. Shi, and S. Zhang, “Modal and harmonic response analysis of key components of robotic arm based on ANSYS,” *Vibroengineering Procedia*, vol. 12, pp. 109–114, 2017, doi: 10.21595/vp.2017.18703.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Optimal Motions of a Welding Robot for Cooperative Tasks with a 2 DOF Positioner

Oscar I. Santellano J. [1], *J. Alfonso Pámanes G.* [2], *Julio C. Rodríguez C.* [3], *J. Jesús Pámanes G.* [4]

[1 2 3 4]Division of Postgraduate Studies and Research, Tecnológico Nacional de México - Instituto Tecnológico de la Laguna
Blvd. Revolución y Calz. Cuauhtémoc Torreón, Coah, Mexico C.P. 27000
Email: [1]oscar.santellano@hotmail.com, [2]japamanesg@correo.itlalaguna.edu.mx,
[3]rodriguezcervantesjc@gmail.com, [4]jdejpamanesg@correo.itlalaguna.edu.mx

***Abstract*— The application of cooperative positioner devices for industrial robotic workstations is becoming an interesting alternative. Such devices ease the accessibility of robots to tasks and increase their productivity. Thus, motion planning of robots working in cooperation with positioners becomes a noteworthy issue for corporations. Indeed, simultaneous and continuous motions of both the positioner and the robot should be suitably designed to increase the performance of the manufacturing system. In this paper, the simultaneous motions of a robot and a positioner table are synthesized such that a complex 3D welding seam is applied on a workpiece. The degree of freedom (DOF) of the considered robot and the cooperating table are 6 and 2, respectively. The applied strategy to obtain the joint trajectories of the system allows optimizing the kinematic performance of the robot during the execution of the task. Two indices of performance of robots are optimized: the translational and the rotational manipulabilities.**

I. INTRODUCCIÓN

Nowadays, more and more corporations of the metal-mechanical industry are coming to apply auxiliary positioner devices (or cooperating tables) in their robotic workstations. Such devices provide easier access to complex zones of the workpieces of the robots. Additionally, the kinematic performance and the productivity of these machines could be notably improved. The motion planning of robots with cooperating tables has been studied under different approaches and distinct levels of complexity since a few years ago [2]. It can be considered, for instance, that the positioner possesses a single degree of freedom (DOF) v.gr. [2], or that the workpiece is moved by another 6-DOF robot [6]. It can be also assumed that both the robot and the positioner device move simultaneously [4] or sequentially [3].

In this paper, the development of a method for motion planning of a 6-DOF robotic manipulator is proposed for achieving tasks with the assistance of a 2-DOF positioner device. It is considered that motions of both machines are executed simultaneously. The motions are planned in such a way that the kinematic performance of the manipulator is optimized while the task is completed. The method is validated by applying it to describe a 3D path with a specified orientation of the end-effector.

In the next section, the architecture of the system is described and the basic equations to solve the kinematic modelling are presented. A brief review of the kinematic performances indices to be optimized of the manipulator is given in Section III. Then, the formulation to solve the motion planning problem is developed in Section IV. The final considerations about the proposed approach are presented in Section V.

II. KINEMATICS MODELLING

The welding robot-positioner system considered is composed by a Robot *Fanuc ARC Mate 100iC* and a rotational positioner *T Solda 100Kg*. This system is installed at the Laboratory of Mechatronics and Control (LaMyC) of the Tecnologico Nacional de México - Instituto Tecnologico de la Laguna (TecNM - ITLag).

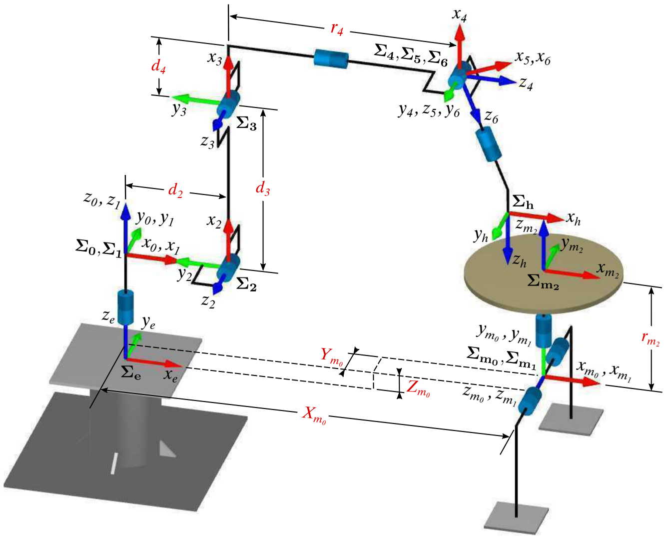


Fig. 1: The kinematic scheme of the robot–positioner system

The kinematic scheme of the system is illustrated in Figure 1, Σ_e denotes the coordinate frame attached to the workstation and Σ_0 designates the coordinate frame attached to the base of the robot, Σ_1, Σ_2,$\cdots$,Σ_6 represents the coordinate frames attached to the successive mobile links of the robot (from the shoulder to the end-effector), respectively. Σ_h designates the coordinate frame attached to the welding torch, with the origin at the tip of the tool. Σ_{m_0} denotates the coordinate frame attached to the base of the positioner table. Σ_{m_1} and Σ_{m_2} represent the coordinate frames attached to the mobile links 1 and 2 (platform) of the positioner, respectively. All of frames of the manipulator are defined based on the modified Denavit-Hartenberg (D-H) notation [7].

The modified D-H parameters for the kinematic chain of the robot are displayed in Table I. The corresponding parameters of the positioner table are given in Table II.

TABLE I: MODIFIED DENAVIT-HARTENBERG PARAMETERS OF THE *FANUC ROBOT ARC MATE® 100/IC*.

j	σ_j	$\alpha_j(^\circ)$	d_j	θ_j	r_j
1	0	0	0	θ_1	0
2	0	90	d_2	θ_2	0
3	0	0	d_3	θ_3	0
4	0	90	d_4	θ_4	r_4
5	0	-90	0	θ_5	0
6	0	90	0	θ_6	0

$d_2 = 150\ mm$; $d_3 = 600\ mm$; $d_4 = 200\ mm$; $r_4 = 640\ mm$

TABLE II: MODIFIED DENAVIT-HARTENBERG PARAMETERS OF THE *T SOLDA®* POSITIONER.

j	σ_{m_j}	$\alpha_{m_j}(^\circ)$	d_{m_j}	ϕ_j	r_{m_j}
1	0	0	0	ϕ_1	0
2	0	-90	0	ϕ_2	r_{m2}

$r_{m_2} = 130\ mm$

From the geometric parameters and employing the SYMORO (SYmbolic MOdeling of RObots) software [8], the kinematic models of the robot and the positioner are obtained. The inverse kinematic model is applied in the formulation to determine the joint variables corresponding to the desired task, with simultaneous motions of the positioner that optimize the kinematic performance of the manipulator. The Jacobian matrix of the robot is considered to evaluate the kinematic performance of the manipulator by using the manipulability indices.

The tool frame Σ_h shown in Figure 2 , allows to specify the desired motion of the welding torch. The pose of the frame Σ_h with respect to Σ_6 is defined by the matrix (1):

$$ {}^6_hT = \begin{bmatrix} \sin\rho & 0 & -\sin\rho & d_{hx} \\ 0 & 1 & 0 & d_{hy} \\ \cos\rho & 0 & \cos\rho & d_{hz} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1} $$

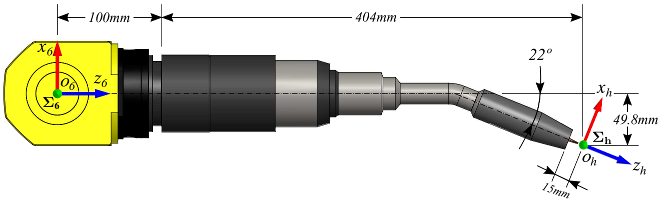


Fig. 2: Torch *ABIROB A500 22* of the robot *Fanuc 100iC/12*.

where $\rho = -22^\circ$, $d_{hx} = -49.8\ mm$, $d_{hy} = 0\ mm$ and $d_{hz} = 504\ mm$.

The homogeneous transformation matrix that specifies the robot location into the workstation, is given by:

$$ {}^e_0T = \begin{bmatrix} 1 & 0 & 0 & e_x \\ 0 & 1 & 0 & e_y \\ 0 & 0 & 1 & e_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2} $$

where $e_x = 0\,mm$, $e_y = 0\,mm$ and $e_z = 450\ mm$. Similarly, the placement of the positioner table with respect to the frame Σ_e is specified by the matrix (3):

$$ {}^{e}_{m_0}T = \begin{bmatrix} \cos\mu & 0 & \sin\mu & X_{m_0} \\ \sin\mu & 0 & -\cos\mu & Y_{m_0} \\ 0 & 1 & 0 & Z_{m_0} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3} $$

The orientation of Σ_{m_0} with respect to Σ_e in (3) is specified by the Bryant's angles λ, μ and ν (rotations in the order Z-Y-Z). The angles λ and ν are specified, 90° and 0°, respectively, in such a way that the base frame of the table match with the plane of the floor.

On the other hand, the placement of the frame attached to the rotating platform of the positioner table, with respect to the fixed base of the table is obtained as:

$$ {}^{m_2}_{m_0}T = \begin{bmatrix} c\phi_1 c\phi_2 & -c\phi_1 s\phi_2 & -s\phi_1 & -r_{m_2} s\phi_1 \\ s\phi_1 c\phi_2 & -s\phi_1 s\phi_2 & c\phi_1 & r_{m_2} c\phi_1 \\ -s\phi_2 & c\phi_1 c\phi_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4} $$

The parameter r_{m_2} in (4) is known from the architecture of the table. The variables μ, X_{m_0}, Y_{m_0} and Z_{m_0} in (3), and ϕ_1 in (4) are related with the design of the workstation. Suitable values of such variables were obtained in [14]. All these variables remain constant during the execution of the task. Its values are kept in 90°, $1200\,mm$, $500\,mm$, $-10\,mm$ and 10.4°, respectively. The variables ϕ_2 in (4), and ψ defining the orientation of the torch will be synthesized in Section IV, as time functions such that the manipulator's kinematic performance be optimized during the execution of the desired task.

The pose of the torch whit respect to the workstation can be obtained via both the manipulator and the positioner table. The corresponding equations are (5) and (6), respectively:

$$ {}^e_hT = {}^e_0T\ {}^0_1T\ {}^1_2T\ {}^2_3T\ {}^3_4T\ {}^4_5T\ {}^5_6T\ {}^6_hT \tag{5} $$

$$ {}^e_hT = {}^e_{m_0}T\ {}^{m_0}_{m_2}T\ {}^{m_2}_{k}T\ {}^k_hT \tag{6} $$

where ${}^{m_2}_{k}T$ is a matrix that defines the pose of the auxiliary frame Σ_k attached to workpiece with respect to the frame Σ_{m_2}, and the matrix ${}^k_hT$ defines the desired pose of the end effector with respect to the frame Σ_k.

On the other hand, the desired pose of Σ_6 with respect to Σ_0, termed as ${}^0_6T^*$ can be found by equating (5) and (6) and solving for ${}^0_6T^*$, as folows:

$$ {}^0_6T^* = {}^e_0T^{-1}\ {}^e_{m_0}T\ {}^{m_0}_{m_2}T\ {}^{m_2}_{k}T\ {}^k_hT\ {}^6_hT^{-1} \tag{7} $$

Once ${}^{0}_{6}T^{*}$ is known the inverse kinematics can be explicitily solved and the corresponding posture of the manipulator is defined for a given pose of the torch.

III. KINEMATIC PERFORMANCE OF THE MANIPULATOR

On the other hand, the twist of the end-effector of the robot is given by:

$$\boldsymbol{t} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{8}$$

where the elements of the vector $\dot{\boldsymbol{q}} \in \mathbb{R}^n$ are the joint velocities of the manipulator, and $\boldsymbol{J}(\boldsymbol{q}) \in \mathbb{R}^{m\times n}$ is the basic Jacobian matrix of the manipulator [5]. The entries of such a matrix are designated as shown in (8), which can be obtained by symbolic computation using the SYMORO software.

$$\boldsymbol{J}(\boldsymbol{q}) = \begin{bmatrix} J_{11} & J_{12} & J_{13} & \cdots & J_{16} \\ J_{21} & J_{22} & J_{23} & \cdots & J_{26} \\ J_{31} & J_{32} & J_{33} & \cdots & J_{36} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ J_{61} & J_{62} & J_{63} & \cdots & J_{66} \end{bmatrix} \tag{9}$$

The twist $\boldsymbol{t}$ of the tool $\in \mathbb{R}^m$ is defined as follows:

$$\boldsymbol{t} \equiv \begin{bmatrix} \dot{\boldsymbol{p}} & \boldsymbol{\omega} \end{bmatrix}^T \tag{10}$$

In (10), $\dot{\boldsymbol{p}}$ is the velocity of the tool center point (TCP) of the end-effector, $\boldsymbol{p}$ being the position vector of such a point with respect to the frame Σ_0. The TCP considered in this study is O_6, shown in Figure 2. The vector $\boldsymbol{\omega}$ is the angular velocity of the end-effector. Both $\dot{\boldsymbol{p}}$ and $\boldsymbol{\omega}$ are expressed in terms of their Cartesian components with respect to the frame Σ_0.

Several kinematic performance indices have been proposed in the scientific literature, based on the Jacobian matrix of manipulators to assess their quality of motions. The *manipulability* index proposed by Yoshikawa [15] is one of the most useful indices for measuring the kinematic performance. Such an index evaluates the ease to displace the end-effector with both, translational and rotational motions. These features were considered as quite suitable for the present work. Therefore, the indices of translational and rotational manipulabilities [16] should be optimized by the manipulator during the execution of the desired task. To be able to apply such indices, we consider the Fanuc robot as composed by two kinematic sub-chains: the first one associated with the arm and the other with the wrist. The arm has $n_{\mathrm{A}} = 3$ joints and the wrist has $n_{\mathrm{W}} = 3$ joints. Thus, the translational manipulability (w_T) of the arm and the rotational manipulability (w_R) of the wrist, are respectively defined as:

$$w_T = \sqrt{\det(J_{TA} {J_{TA}}^T)} \tag{11}$$

$$w_R = \sqrt{\det(J_{RW} {J_{RW}}^T)} \tag{12}$$

where J_{TA} and J_{RW} are the partitions of the manipulator's basic Jacobian matrix (9), defined as:

$$J_{TA} = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{bmatrix} \tag{13}$$

$$J_{RW} = \begin{bmatrix} J_{44} & J_{45} & J_{46} \\ J_{54} & J_{55} & J_{56} \\ J_{64} & J_{65} & J_{66} \end{bmatrix} \tag{14}$$

The Rotational manipulability w_R is bounded as $0 \le w_R \le 1$. However, the translational manipulability should be normalized in order to consistently compare the values of w_T with those of w_R. The normalized translational manipulability is defined as:

$$w_T{}^{*} = \frac{w_T}{w_{Tmax}} \tag{15}$$

where $w_{T_{max}}$ is the maximal translational manipulability that could obtain the manipulator with the optimal posture. In the case of the robot considered in this paper, $w_{Tmax} = 0.45\,m^3$. Thus, $w_T{}^{*}$ is bounded as $0 \le w_T{}^{*} \le 1$. The greater the manipulabilities are, the better the ease to obtain translational and rotational motions of the end-effector. In contrast, when any of the manipulabilities is equal zero, the manipulator gets a singular posture, and it loses the ability to produce translational or rotational motions of the end-effector on a specific direction. Moreover, in the neighborhood of a singular posture, one or both manipulabilities have small amounts and, consequently, the ability of the manipulator is quite poor to generate arbitrary displacements on specific directions.

IV. MOTION PLANNING

The motion planning problem considered in this Section consists in synthesizing the joint trajectories $\phi_1(t)$, $\phi_2(t)$ and $\psi(t)$ that eases the execution of the desired task to the robot. As observed in Figure 3, the former variable defines the posture of the positioner table, and the last one corresponds to one coordinate of orientation of the robot tool. By defining such trajectories, the whole postures of the robot can be computed by applying the inverse kinematic model of the robot. As indicated in Section II, a preliminary value was suitably specified for ϕ_1 [14]. We will see that such an angle doesn't require be defined by a time function.

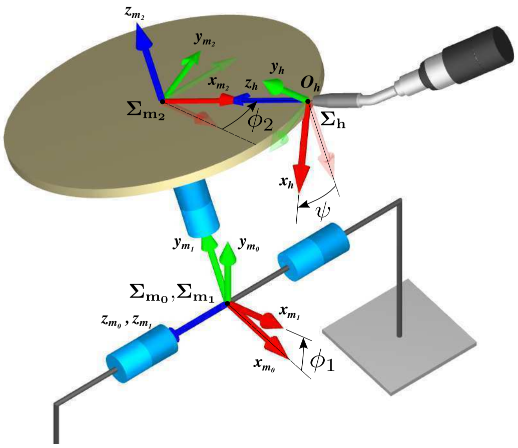


Fig. 3: Angles subject to the motion optimization process.

The synthesis should be achieved in such a way that both the translational and the rotational manipulabilities of the robot be optimized during the execution of the desired task. It must be noted that, even if the orientation of the torch should be specified considering the quality of the welding process, the coordinate ψ could be defined under other criteria since the welding can be properly achieved by using any value of such an angle because of the symmetry of the tool tip.

To present the approach proposed in this paper, a task is considered in our formulation such that the robot has to apply a welding seam on the external surface of a cylinder in a time T of 40 seconds at a constant speed. The geometry of the welding seam is defined by a one cycle helical curve with a radius r_t of 100 mm and a height h_t of 450 mm. This path is shown in Figure 4. The longitudinal axis of the helical path matches the z axis of the frame Σ_k attached to the base of the cylinder, as shown in Figure 4.

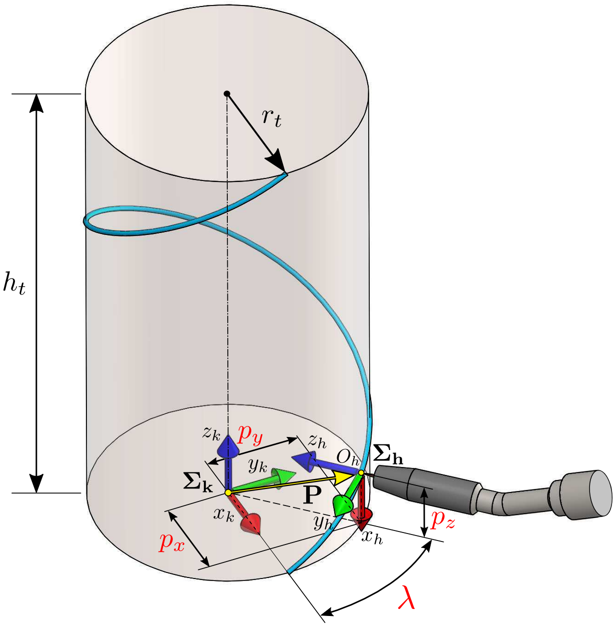


Fig. 4: Desired tool path and orientation for the welding task.

The desired position of the origin O_h with respect to Σ_k is defined by the position vector defined in (16), whose Cartesian components are given by the time functions (17):

$$\boldsymbol{P} = \begin{bmatrix} p_x(t) & p_y(t) & p_z(t) \end{bmatrix}^T \tag{16}$$

where,

$$p_x(t) = r_t \cos(\lambda(t)) \tag{17a}$$

$$p_y(t) = r_t \sin(\lambda(t)) \tag{17b}$$

$$p_z(t) = h_t \frac{t}{T} \tag{17c}$$

In (17a) $\lambda(t)$ is the angle between the x_k axis and the projection of $\boldsymbol{P}$ in the x_k - y_k plane, as observed in Figure 4. The time function that defines this angle is expressed as:

$$\lambda(t) = 2\pi t/T \tag{18}$$

The orientation of the end-effector is specified by the Euler angles ϕ, θ and ψ (in Z-Y-Z order) of the frame Σ_h, with respect to Σ_k. These coordinates are specified as follows:

$$\phi(t) = 180^\circ + \lambda(t) \tag{19}$$

$$\theta = 90^\circ \tag{20}$$

$$\psi(t) = \psi(t) \tag{21}$$

where, (19) keep the tip of the torch normal to the external surface of the cylinder curve with radius r_t. $\psi(t)$ is a time function to be proposed in the optimization process. This angle take values from 180° to -180°.

The homogeneous matrix that specifies the pose of the torch at each point of the path is:

$${}^k_hT = \begin{bmatrix} {}^k_h\boldsymbol{R} & \boldsymbol{P} \\ & \\ \mathbf{0}^T & 1 \end{bmatrix} \tag{22}$$

where the rotation matrix ${}^k_h\boldsymbol{R}$ is given by

$${}^k_h\boldsymbol{R} = \begin{bmatrix} c\phi c\theta c\psi - s\phi s\psi & -c\phi c\theta s\psi - s\phi c\psi & c\phi s\theta \\ s\phi c\theta c\psi + c\phi s\psi & -s\phi c\theta s\psi + c\phi c\psi & s\phi s\theta \\ -s\theta c\psi & s\theta s\psi & c\theta \end{bmatrix} \tag{23}$$

where $c\bullet \equiv \cos\bullet$ and $s\bullet \equiv \sin\bullet$.

The objective function to be minimized in the optimization process is the one defined in [11] as:

$$f = -(\overline{w} - w_\sigma) \tag{24}$$

where $\overline{w}$ is the mean of the mixed manipulabilities corresponding to a sample of n_p points of the desired path of the end-effector. And w_σ is the standard deviation of the same sample. Thus, the difference $\overline{w} - w_\sigma$ represents a typically small *mixed manipulability* in the set of the mixed manipulabilities of the sample. Consequently, by minimizing the function (24), we globally maximize the whole set of mixed manipulabilities corresponding to the task.

The mixed manipulability associated with the *i*-th path point $(i = 1, 2, \cdots, n_p)$ is defined as:

$$w_i \equiv \overline{w_i{}^*} - w^*{}_{\sigma i} \tag{25}$$

where $\overline{w_i{}^*}$ is the mean of the normalized translational manipulability $(w^*{}_{Ti})$ and the rotational manipulability (w_{Ri}) corresponding to the posture of the manipulator at the *i*-th path point. Besides, $w^*{}_{\sigma i}$ is the standard deviation of both manipulabilities at the same point. Therefore, the mixed manipulability represents a typically small manipulability (among $w^*{}_{Ti}$ and w_{Ri}) of the robot at the *i*-th path point.

On the other hand, because of the welding seam must be applied on a cylindrical surface, the joint variable ϕ_2 could be employed to rotate, about its own axis, the cylinder to be welded. As a result, the robot couldn't require moving the torch around the cylinder. Only motions of the torch in the neighborhood of a plane should be necessaries to describe the helicoidal path and to keep the desired orientation of the tool. Thus, the time function considered for the joint variable ϕ_2 will be defined such that its first derivative be constant, with semi-cycloidal motions for starting and stopping motion. We propose:

$$\phi_2(t)=\begin{cases}\phi_2(t_0)+\Delta_1\left[\dfrac{t-t_0}{T_1}-\dfrac{\sin\left(\frac{\pi(t-t_0)}{T_1}\right)}{\pi}\right] & ,\quad t_0\le t\le t_1\\[2ex] \phi_2(t_1)+\dfrac{\Delta_2}{T_2}(t-t_1) & ,\quad t_1\le t\le t_2\\[2ex] \phi_2(t_2)+\Delta_3\left[\dfrac{t-t_2}{T_3}+\dfrac{\sin\left(\frac{\pi(t-t_2)}{T_3}\right)}{\pi}\right] & ,\quad t_2\le t\le T\end{cases}\tag{26}$$

$$\dot\phi_2(t)=\begin{cases}\dfrac{\Delta_1}{T_1}\left[1-\cos\left(\frac{\pi(t-t_0)}{T_1}\right)\right] & ,\quad t_0\le t\le t_1\\[2ex] \dfrac{\Delta_2}{T_2} & ,\quad t_1\le t\le t_2\\[2ex] \dfrac{\Delta_3}{T_3}\left[1+\cos\left(\frac{\pi(t-t_2)}{T_3}\right)\right] & ,\quad t_2\le t\le T\end{cases}\tag{27}$$

where the parameters of these equations are shown in the Table III.

TABLE III: PARAMETERS OF THE EQUATIONS (26) AND (27).

i	t_{i-1}	T_i	$\phi_2(t_{i-1})$	Δ_i
1	$0\ s$	$2.5\ s$	180°	-12°
2	$2.5\ s$	$35\ s$	168°	-336°
3	$37.5\ s$	$2.5\ s$	-168°	-12°

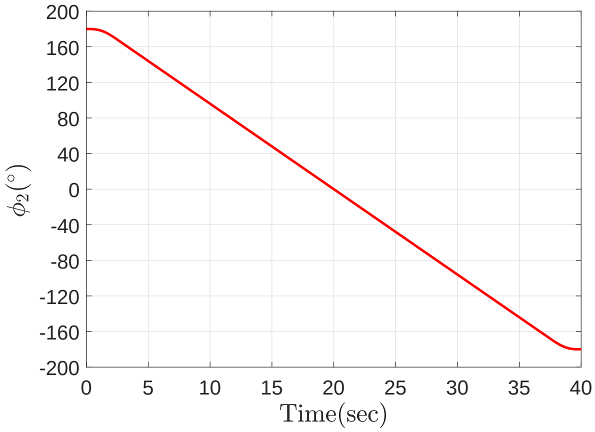


Fig. 5: Synthesized time function for $\phi_2(t)$.

Once the time function of ϕ_2 has been defined, an exploration is carried out in order to identify a time function for the variable ψ associated with the orientation of the tool. Our analysis is based on the use of performance maps [13]. In the problem considered here, an interesting performance map allow to observe the zones in the plane defined by the variables t and ψ that provide the postures corresponding to the desired task that give the better mixed manipulabilities of the robot. In Figure 6 it may be observed this performance map. The top view of the map is displayed in Figure 7. In this projection it is possible to visualize a continuous path for ψ, that goes from $t_0 = 0$ until $T = 40$ seconds, such that higher values of *w* may be obtained. To synthesize a continuous path on this map, five via points are strategically selected, which give higher values of *w* along the path. These points are then jointed by using cycloidal functions in such a way that continuous and smooth motions of the tool be gotten. As a result, the curve in black color is obtained. This curve is composed by four cycloidal functions ($n_f = 4$), that may be represented by (28).

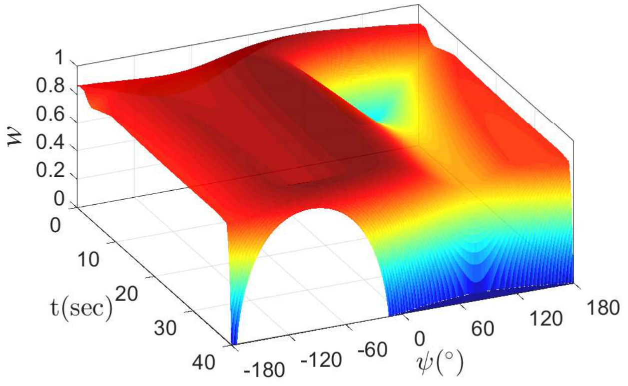


Fig. 6: Performance map of the robot *Fanuc 100iC/12*.

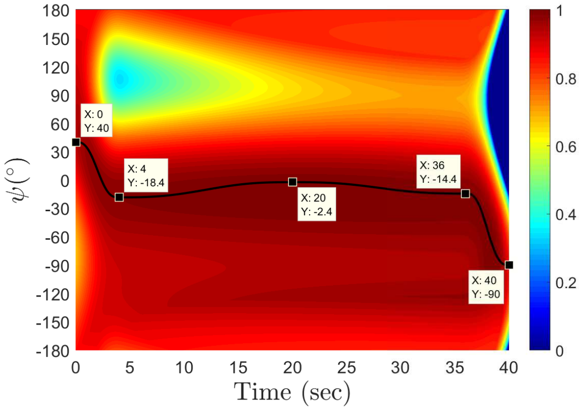


Fig. 7: Contour lines of the robot performance map for the desired task, and synthesized optimal trajectory for the angle ψ.

$$\psi_i(t)=\psi(t_{i-1})+\delta\psi_i\left[\frac{t-t_{i-i}}{T_i}-\frac{1}{2\pi}\sin\left(\frac{2\pi(t-t_{i-1})}{T_i}\right)\right]\tag{28}$$

The *i*-th cycloidal function is bounded by $t_{i-1}\le t\le t_i$ where $i=\{1,2,\cdots,n_f\}$. The corresponding parameters for each interval are given in Table IV.

Thus, by employing $\phi_1 = 10.4^\circ$, and the time functions (26) and (27) for $\phi_2(t)$, and (28) for $\psi(t)$, the matrices ${}^{m_0}_{m_2}T$, ${}^{m_2}_{k}T$ and ${}^{k}_{h}T$ may be finally specified on each time sample in the interval from 0 to $40\,s$. These matrices are applied in (7) to determine ${}^{0}_{6}T^{*}$, which is the desired *SNAP* matrix of the Paul's method to solve the inverse kinematic problem [5] on each time sample.

TABLE IV: PARAMETERS OF EQUATION (28).

i	t_{i-1}	T_i	$\psi(t_{i-1})$	$\delta\psi_i$
1	$0\,s$	$4\,s$	$40°$	$-58.4°$
2	$4\,s$	$16\,s$	$-18.4°$	$16°$
3	$20\,s$	$16\,s$	$-2.4°$	$-12°$
4	$36\,s$	$4\,s$	$-14.4°$	$-76.6°$

The obtained joint trajectories are presented in Figure 8. They allow us to carry out an analysis of the robot's translational and rotational manipulabilities in order to assess the effectiveness of our approach. The curves in Figure 9 show the behavior of such indices of performance. It can be appreciated that both indices are basically equal to 1 on all the points along the task. Notice that it is not required to move the value of the joint variable ϕ_1 along the time to improve the performance of the robot. Only time functions are necessary for ϕ_2 and ψ.

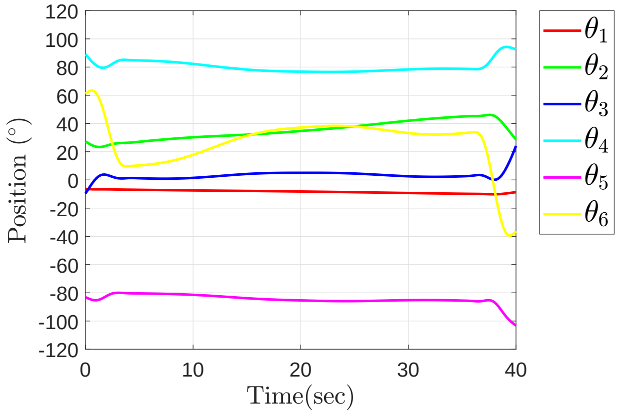


Fig. 8: Joint variables of the robot *Fanuc 100iC/12*.

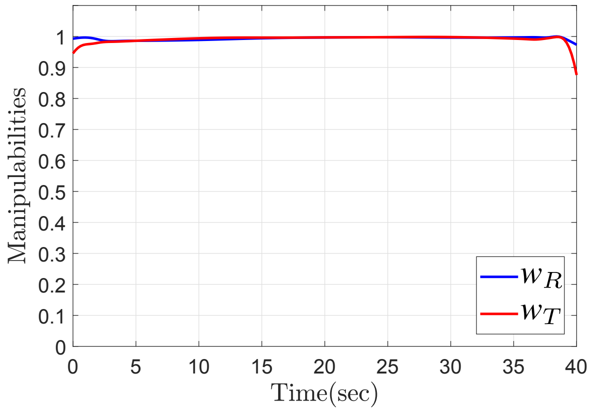


Fig. 9: Translational and rotational manipulabilities of the robot during the execution of the task.

To validate the results, the obtained joint trajectories of the robot were finally applied as control signals during an experiment in the laboratory. A sequence of postures of the robot during the experiment is displayed in Figure 10.

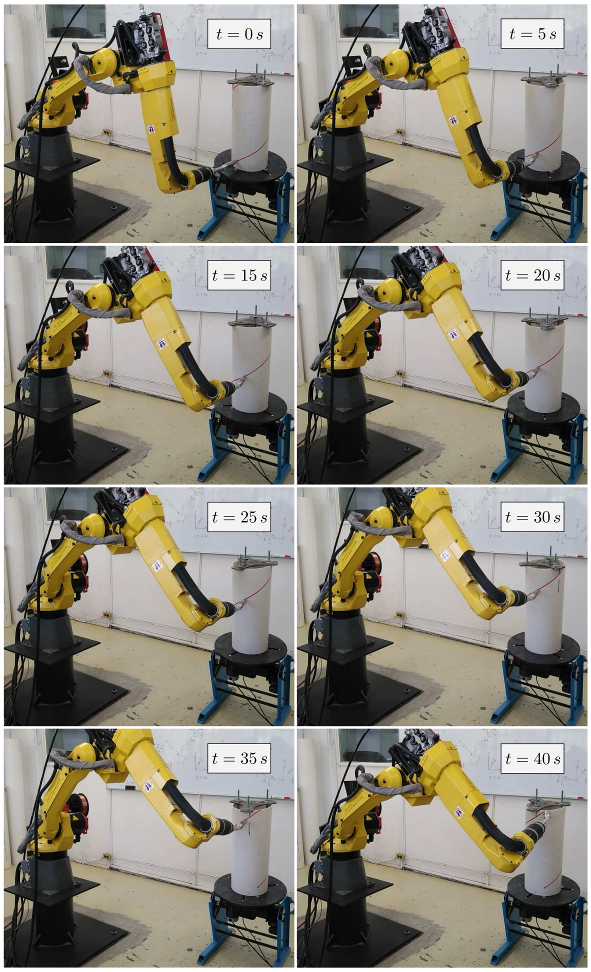


Fig. 10: Sequence of postures of the robot and the positioner table, obtained during the task execution.

V. CONCLUSION

The proposed approach for motion planning of a 6-DOF robotic manipulator, working in cooperation with a 2-DOF positioner table, is effective for optimizing the kinematic performance of the robot during the task execution. The synthesized trajectories were validated by achieving experimental tests in the LaMyC of the TecNM-ITLag. The method was used to optimize the translational and rotational manipulabilities of the robot. However, other indices of performance could be optimized.

An experimental site was developed by incorporating a positioner table and its control system to the workstation of the industrial robot of the TecNM-ITLag. Thus, the LaMyC is already equipped with human and material resources of hardware and software to provide service to the industry if necessary for motion planning of cooperative robotic systems.

Even if constraints to the proposed formulation were not necessary for avoiding collisions or for preventing transgressions of joint limits, in future works such constraints should be incorporated to the method in order to enable our software tools to solve more complex problems.

VI. ACKNOWLEDGMENTS

The work reported in this paper was sponsored by the Tecnologico Nacional de México, the Instituto Tecnologico de la Laguna and the CONACyT of Mexico. Thanks to these agencies for their support.

REFERENCES

[1] Alatartsev S., Stellmacher S., and Ortmeier F., "*Robotic Task Sequencing Problem: A Survey*"; Journal of Intelligent and Robotic Systems, Vol. 80, Issue 2, pp. 279-298 (2015).
[2] Chen H., Liu Z., "*Offline Programming for an Arc Welding Robot with Redundant GDL*", Applied Mechanics and Materials, Vols. 184-185, pp. 1623-1627 (2012).
[3] De Anda J., "*Motion Planning of a Welding Robot for Cooperative Tasks with a Sequential Positioning Device*", XX Congreso Mexicano de Robótica (COMRob), pp. 1-6 (2018)
[4] Dena O., "*Optimal motion planning of an industrial robot operating simultaneously with a positioner table in welding tasks*", Master in Science Thesis, Tecnológico Nacional de México - Instituto Tecnológico de la Laguna (2019).
[5] Dombre E., and Khalil W., "*Modélisation et Commande des Robots*", Paris, Edition Hermes 1988.
[6] Hemmerle J.S., and Prinz F.B., "*Optimal path placement for kinematically redundant manipulators*", Proceedings of the 1991 International Conference on Robotics and Automation, pp. 1234-1243 (1991).
[7] Khalil W., and Kleinfinger J.F., "*A new geometric notation for open and closed- loop robots*", Proc. of the 1986 IEEE Conference on robotics and Automation; pp. 1174-1180 (1986).
[8] Khalil W., and Lemoine Ph., SYMORO+ Symbolic Modeling of Robots: User Guide, IRCCyN (*Institut de Recherche en Communication et Cybernétique de Nantes*), Nantes, Francia, 2003.
[9] Lin Y., Zhao H., and Ding H., "*Posture optimization methodology of 6R industrial robots for machining using performance evaluation indexes*", Robotics and Computer Integrated Manufacturing, 48, pp 59-72 (2017).
[10] Pamanes J.A., Blásquez J.R., and Llama M.A., "*Análisis de la redundancia cinemática de robots industriales en tareas de soldadura*", Memorias del XII Congreso Internacional Anual de la SOMIM (Sociedad Mexicana de Ingeniería Mecánica), pp. 571-579, (2016).
[11] Pámanes J.A., Cuán E. and Zeghloul S., "*Single and multi-objective optimization of path placement for redundant robotic manipulators*", INGENIERÍA Investigación y Tecnología, Vol. IX, No. 3, pp. 231-257; México, Jul-Sept 2008.
[12] Pámanes J.A., Soto D., Llama M.A., and Reveles D., "*Optimization of the design of an experimental robotic welding station*", Memorias del XVIII Congreso Mexicano de Robótica (COMRob), de la AMRob, pp. 179-188 (2016).
[13] Reveles D, Pámanes JA, and Wenger Ph, "*Trajectory planning of kinematically redundant parallel manipulators by using multiple working modes*", Mechanism and Machine Theory. Vol. 98, pp. 216-230, (2016).
[14] Santellano O.I., "*Motion planning of an industrial robot operating simultaneously with a 2 DOF positioner table*", Master in Science Thesis, Tecnológico Nacional de México - Instituto Tecnológico de la Laguna (2021).
[15] Yoshikawa T, "*Manipulability of Robotic Mechanisms*", International Journal of Robotics Research, Vol. 4, No. 2, pp. 3-9 (1985).
[16] Yoshikawa T, "*Translational and Rotational Manipulability of robotic Manipulators*", Proceedings of the American Control Conference, pp. 228-233 (1990).
[17] Zeghloul S., Pamanes J.A., "*Multi-criteria optimal placement of robotics in constrained environments*", Robotica, Vol. 11, pp 105-110, (1993).

Capítulo 4.

Mobile Robotics and Teloperation

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Navegación autónoma de un robot móvil en un ambiente desconocido usando una red de neuronas tipo Hindmarsh-Rose

Enrique Martinez Sanchez
Electrical Engineering Department
CINVESTAV-IPN
Mexico City, Mexico
e-mail: enrique.mtz@cinvestav.mx

Jonatan Pena Ramirez
Electronics and Telecommunications Department
CICESE
Ensenada, Mexico
e-mail: jpena@cicese.mx

Alejandro Rodriguez Angeles
Electrical Engineering Department
CINVESTAV-IPN
Mexico City, Mexico
e-mail: aangeles@cinvestav.mx

***Resumen*—En este trabajo se propone un control híbrido para la navegación autónoma de un robot móvil en un espacio con obstáculos estáticos y dinámicos. Este control híbrido está conformado por la conmutación entre un control que permite el seguimiento de trayectoria y un control de evasión basado en neuronas tipo Hindmarsh-Rose (HR), interconectadas mediante un acoplamiento difuso. La conmutación se da en función de la lectura de los sensores integrados al robot y se opera a nivel de velocidad de rueda. Para que se dé la evasión de obstáculos, se utiliza una fuerza de acoplamiento pequeña, lo cual permite aprovechar la desincronización de los potenciales de membrana ante entradas de corriente externas diferentes para cada neurona, de esta manera se obtiene una mayor sensibilidad ante alguna detección. Por otra parte, se hace uso de un acoplamiento dinámico para establecer una estrategia de evasión frontal. Para mostrar el desempeño del control se presentan resultados de simulación.**

***Index Terms*—Neuronas Hindmarsh-Rose (HR), acoplamiento difuso, acoplamiento dinámico, control basado en el modelo neuronal, espigueo**

I. INTRODUCCIÓN

Existen diversas aplicaciones que pueden desempeñar los robots móviles autónomos: el transporte de material, limpieza, vigilancia, localización de objetos, exploración no tripulada, entre otros [1]–[11]. La navegación autónoma de los robots móviles puede realizarse por medio de la planificación previa de una ruta, esto suele realizarse en ambientes totalmente conocidos y que no cambian con el paso del tiempo, en estas condiciones se puede probar la existencia de un camino libre de colisiones y posiblemente optimo hacia la meta (en términos de distancia, tiempo y energía empleada) [5]–[7]. Desafortunadamente, el caso anterior es poco común en muchas aplicaciones donde se considera un entorno real, en las que el ambiente es desconocido. Este hecho requiere que un robot móvil sea capaz de identificar y evadir obstáculos (estáticos y dinámicos), realizar autolocalización, tomar decisiones, etc. [1], [2], [4], [11]. Dentro de la amplia gama de controles que abordan la problemática de la navegación autónoma de un robot móvil, se hallan controles con enfoques del tipo híbrido, que son conformados por controladores para la navegación reactiva y controladores para un seguimiento de trayectoria [5]–[9]. Ejemplo de ello, son los controladores de sistemas de interferencia difusa que requieren de un entorno parcialmente conocido y se conmutan de la forma más suave posible, para elaborar una fase de preprocesamiento dependiente a una tarea específica y así poder construir una ruta de referencia óptima en tiempo [5]. Por otra parte, está la combinación entre controles PI, y controladores de lógica difusa con algoritmos de retroceso que actúan de forma simultánea, y se encargan de establecer una fusión de datos obtenidos de sensores, construcción de mapas y tareas de planificación de rutas [6]. Para imitar comportamientos más naturales, se han desarrollado algunos controladores bio-inspirados los cuales poseen un radio de confort y un lazo externo que contiene fuerzas de evasión [4], ademas, estan aquellos controles que imitan el manejo de automóviles impulsados por una respuesta humana basados en lógica difusa, acciones reflexivas y toma de decisiones de alto nivel [10]. También, se hallan controles que poseen una intercomunicación inalámbrica entre una computadora móvil (que se incorpora al robot y se encarga de recopilar datos del entorno mediante sensores laser), y una computadora estacionaria que se encarga de elaborar un algoritmo de control para la evasión de objetos dinámicos y la búsqueda de una ruta adaptativa [9]. Otro aspecto importante a resaltar, es la variedad de sensores que se pueden integrar, ya que estos proporcionan al robot móvil la información necesaria para poder realizar sus tareas, entre más sofisticados sean estos sensores, mejor percepción tendrá el robot de su entorno, sin embargo, estos sensores también deben seleccionarse de acuerdo con el tipo de controlador a utilizar (infrarojos, Lidar, ultrasonicos, Bluetooth, cámaras, módulos GPS, módulos ZigBee, entre otros) [1]–[11]. Actualmente, una de las propuestas más atractivas de control para la navegación autónoma de robots móviles es el uso de un “cerebro artificial”, el cual está formado por redes de neuronas. Los candidatos para estos cerebros artificiales son las redes de neuronas tipo spiking (espigueo), interconectadas entre sí, y cuya respuesta en los potenciales de membrana de cada una de ellas es del tipo spiking, imitando de esa forma la arquitectura de un cerebro biológico [1], [2]. Lo interesante en un modelo neuronal y la forma en que se interconecta, es la variedad de comportamientos y patrones de espigueo que se pueden generar en su respuesta, estos patrones se pueden interpretar y asociar con una tarea o acción especifica (tres comportamientos principales que se pueden obtener son conocidos como spiking, bursting y chaotic bursting) [1]–[3].

En este trabajo, se propone utilizar el modelo neuronal Hindmarsh-Rose (HR), puesto que produce una respuesta adecuada para la idea que se tiene como control de evasión, además, su implementación en electrónica analógica es relativamente simple. Se contemplan cuatro neuronas tipo HR, dos de ellas son interconectadas mediante un acoplamiento difuso [2], y las dos restantes interactúan a través de un acoplamiento dinámico, el cual ofrece una mejor sincronización y permite operar en un modo de relajación completa [3].

Las contribuciones que presenta este artículo consisten en reducir el número de neuronas que permiten la evasión lateral, y en agregar tres parámetros que permiten ajustar de mejor forma las corrientes externas del modelo neuronal que modulan la frecuencia del espigueo en los potenciales de membrana, con el fin de hacer más preciso el ajuste del giro de evasión. Además, se propone una estrategia de evasión frontal utilizando dos neuronas interconectadas a través de un acoplamiento dinámico, que favorece un modo de relajación completa. Por otra parte, se añade la implementación de un control de seguimiento de trayectoria que junto con las estrategias de evasión, le permite al robot móvil establecer una navegación autónoma, en donde sigue una trayectoria deseada mientras evade obstáculos.

La Sección II de este articulo expone el planteamiento del problema y muestra el modelo cinemático del robot móvil a utilizar. En la Sección III se implementa una estrategia de control para el seguimiento de trayectoria. La Sección IV muestra el control de evasión basado en neuronas tipo HR. En la Sección V se expone el control híbrido. La Sección VI muestra los resultados de una simulación numérica, así como una discusión de los mismos. Finalmente, en la Sección VII se presentan las conclusiones y trabajo a futuro.

II. MODELO Y PLANTEAMIENTO DEL PROBLEMA

II-A. Planteamiento del problema

Se quiere hacer navegar a un robot móvil tipo 2.0 (robot diferencial) de forma autónoma en un ambiente desconocido. En este trabajo se considera como ambiente desconocido, a un mapa plano (entorno 2D) ya sea abierto o cerrado, en el cual se encuentran obstáculos del tipo estáticos y dinámicos (otros robots móviles). Adicional a ello, se quiere que el robot realice un seguimiento de trayectoria deseada, y si este se encuentra con algún obstáculo en su camino, sea capaz de evadirlo. Para abordar el problema, se pretende implementar un control híbrido conformado por un control de seguimiento de trayectoria y un control de evasión basado en neuronas tipo HR.

II-B. Modelo cinemático del robot móvil

Considere el robot móvil tipo 2.0 mostrado en la Figura 1, cuyo modelo cinemático está dado por

$$\begin{aligned} \dot{x} &= V\cos(\theta), \\ \dot{y} &= V\sin(\theta), \\ \dot{\theta} &= W, \end{aligned} \tag{1}$$

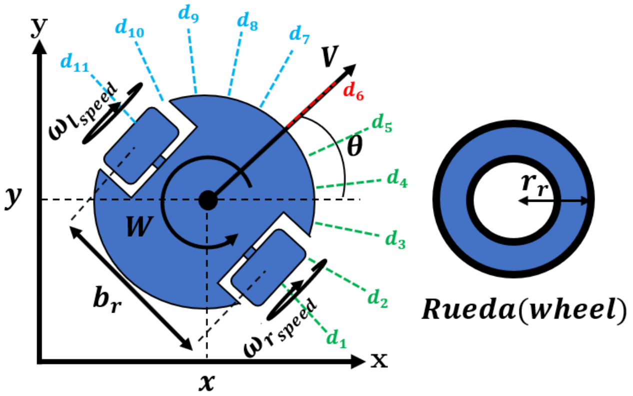


Figura 1. Robot móvil tipo 2.0 y orientación de los sensores.

donde (x, y) es el punto que representa el centro y posición del robot, el ángulo θ es su orientación con respecto al eje horizontal, mientras que V y W son las velocidades traslacional y rotacional, respectivamente, y las posiciones radiales de los sensores están espaciadas por $\frac{\pi}{10}$.

III. ESTRATEGIA DE CONTROL DE SEGUIMIENTO DE TRAYECTORIA

Para poder establecer el seguimiento de trayectoria se opta por la estrategia de un robot virtual de referencia el cual genera la trayectoria deseada, de esta forma el robot móvil se encargará de seguir al robot virtual de referencia.

III-A. Procedimiento de la estrategia de control

De acuerdo con [12], se debe determinar V y W que actúan como entradas de control para el robot móvil.

1) Se establecen las ecuaciones paramétricas de la posición deseada x_d y y_d, sus primeras y segundas derivadas $\dot{x}_d, \dot{y}_d$ y $\ddot{x}_d, \ddot{y}_d$, respectivamente.

2) Se calculan las velocidades V_r, W_r, y la posición angular θ_r del robot virtual

$$\begin{aligned} V_r &= \sqrt{\dot{x}_d^2 + \dot{y}_d^2}, \\ W_r &= \frac{\dot{x}_d \ddot{y}_d - \dot{y}_d \ddot{x}_d}{V_r^2}, \\ \theta_r &= \tan^{-1}\left(\frac{\dot{y}_d}{\dot{x}_d}\right). \end{aligned} \tag{2}$$

3) Se formula el error de seguimiento

$$e = \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_d - x \\ y_d - y \\ \theta_r - \theta \end{pmatrix}. \tag{3}$$

4) Se propone el control

$$\begin{aligned} u_1 &= -k_1 e_1, \\ u_2 &= -k_2 \tanh(10V_r) e_2 - k_3 e_3. \end{aligned} \tag{4}$$

5) Se sintonizan las ganancias k_1, k_2, k_3 del control

6) Se determina el valor de V y W

$$\begin{aligned} V &= -u_1 + V_r \cos(e_3), \\ W &= W_r - u_2. \end{aligned} \tag{5}$$

7) Finalmente, se obtienen las velocidades angulares de la rueda derecha ω_{r_T} e izquierda ω_{l_T}

$$\omega_{r_T} = \frac{1}{r_r}\left(V + \frac{b_r W}{2}\right), \tag{6}$$

$$\omega_{l_T} = \frac{1}{r_r}\left(V - \frac{b_r W}{2}\right), \tag{7}$$

donde r_r es el radio de las ruedas del robot, b_r es la separación entre las ruedas. Note que las Ecs. (6) y (7) permiten operar a nivel de velocidad de rueda.

IV. ESTRATEGIA DE CONTROL DE EVASIÓN BASADA EN NEURONAS TIPO HR

En esta estrategia de evasión se aprovecha la variación de espigueo en los potenciales de membrana de un primer par de neuronas (HR1 y HR2), que se asocian a las velocidades de cada rueda del robot móvil. También, se contempla una comunicación entre otro par de neuronas (HR3 y HR4), y se utiliza la respuesta del potencial de membrana en HR4 para establecer una evasión frontal. El procedimiento está inspirado en [2] y se muestra a continuación.

1) Se establecen los modelos neuronales [1]–[3]

$$HRi: \begin{cases} \dot{x}_i &= y_i + b x_i^2 - a x_i^3 - z_i + I_i + \upsilon_i, \\ \dot{y}_i &= c - d x_i^2 - y_i, \\ \dot{z}_i &= r_i[s(x_i - x_0) - z_i], \end{cases} \tag{8}$$

donde $i = 1, 2, 3, 4$, $x_i \in R$ es el potencial de membrana, $y_i \in R$ es la corriente de recuperación, $z_i \in R$ es la corriente de adaptación, $a, b, c, d, s, r_1 = r_2 = r, r_3 = r_4 = r_d, x_0$ son parámetros constantes. $a \in R$ permite conmutar entre comportamientos de bursting y spiking, además determina la frecuencia del spiking. Por otra parte $r \in R$ gobierna el comportamiento bursting, $s \in R$ controla la adaptación, si este posee valores pequeños (alrededor de 1) provocan un comportamiento spiking rápido. $x_0 \in R$ es el potencial de reposo del sistema que sirve como potencial de referencia, y $b, d, c \in R^+$ son parámetros positivos. I_1, I_2 e $I_3 = I_4 = I \in R$ son las corrientes externas (entradas del modelo), al tener una correcta configuración de los parámetros, se puede variar el valor de la corriente externa, entre mayor sea esta corriente, mayor número de picos se reproducirán en el potencial de membrana y viceversa.

Las neuronas HR1 y HR2, interactúan a través de un acoplamiento difuso:

$$\begin{cases} \upsilon_1 &= \varepsilon(y_2 - y_1), \\ \upsilon_2 &= \varepsilon(y_1 - y_2), \end{cases} \tag{9}$$

con fuerza de acoplamiento $\varepsilon > 0$.

Las neuronas HR3 y HR4, están interconectadas mediante un acoplamiento dinámico:

$$\begin{cases} \upsilon_3 &= h_2, \\ \upsilon_4 &= -G h_2, \end{cases} \tag{10}$$

$$\begin{cases} \dot{h}_1 &= -\alpha h_1 + h_2, \\ \dot{h}_2 &= -\gamma_1 h_1 - \gamma_2 h_2 - k(x_3 - x_4), \end{cases} \tag{11}$$

con fuerza de acoplamiento k y parámetros $\alpha, \gamma_1, \gamma_2$.

Finalmente G es una expresión propuesta, del tipo exponencial que está en función del sensor frontal.

$$G = -11e^{-3(d_{fron} - x_{fmin})} + 1. \tag{12}$$

Con la expresión G se obtienen valores que le permiten a x_3 y x_4 cambiar de un comportamiento del tipo espigueo a un comportamiento de relajación completa y viceversa. x_{fmin} es una distancia de umbral que cuando es alcanzada por la lectura del sensor frontal concede el cambio de un comportamiento a otro (esto se detalla de una mejor forma en el paso cinco del procedimiento).

2) Se distribuye equidistantemente la posición radial de 11 sensores tipo lidar, véase la Figura 1.

3) Se obtienen las distancias mínimas captadas por todos los sensores (d), por el grupo de sensores del lado izquierdo (d_{izq}), por el grupo del lado derecho (d_{der}) y la lectura del sensor frontal (d_{fron}).

$$\begin{aligned} d_{der} &= min(d_1, d_2, d_3, d_4, d_5), \\ d_{izq} &= min(d_7, d_8, d_9, d_{10}, d_{11}), \\ d_{fron} &= d_6, \\ d &= min(d_{izq}, d_{der}, d_{fron}). \end{aligned} \tag{13}$$

4) Se establecen las expresiones para las corrientes externas I_1, I_2 (entradas de HR1-HR2) en función de la distancia del grupo de sensores derecho e izquierdo.

$$I_j = \begin{cases} a_1 I_{min} & d_j \leq x_{min} \\ b_1 I_{min} \left(\dfrac{I_{max}}{I_{min}}\right)^{\dfrac{d_j - x_{min}}{x_{max} - x_{min}}} & x_{min} < d_j < x_{max} \\ c_1 I_{max} & d_j \geq x_{max} \end{cases} \tag{14}$$

donde $j \in \{der, izq\}$, $I_1 = I_{der}$, $I_2 = I_{izq}$. I_{min} e I_{max} son las corrientes externas mínima y máxima, respectivamente, x_{min} y x_{max} son las distancias mínima y máxima, respectivamente, que puede leer el sensor. Los parámetros a_1, b_1, c_1, se utilizan para un mejor ajuste en la caracterización de las neuronas.

5) Se realiza un conteo de espigas en x_1, x_2 durante una ventana de tiempo (en este trabajo se consideró una ventana de 100.0[ms]) y se obtienen las velocidades de cada rueda por medio de la siguiente expresión.

$$\omega_{l_S} = n_1 p; \quad \omega_{r_S} = x_{4A} n_2 p \tag{15}$$

donde ω_{l_S} y ω_{r_S} son las velocidades de las ruedas izquierda y derecha, respectivamente, $j = 1, 2$, n_j es el número de espigas contadas en el potencial de membrana x_j, y p una ganancia de sintonización para regular la velocidad del robot móvil. El potencial de membrana x_4 puede entrar en un comportamiento tipo espigueo, o en un modo de relajación completa, esto dependiendo del valor que tome G, ver Ec. (12). La señal x_{4A} es la señal x_4 acondicionada, x_{4A} toma un valor unitario positivo ($x_{4A} = 1.0$) cuando x_4 entra en modo espigueo, sin embargo, x_{4A} toma un valor unitario negativo ($x_{4A} = -1.0$) cuando x_4 entra en el modo de relajación completa. De acuerdo con una adecuada sintonización en los parámetros de HR3 y HR4, y la propuesta en (12), x_4 entra en modo de relajación completa cuando $d_{fron} \leq x_{fmin}$, si esto no es así, x_4 tendrá un modo espigueo. De esta forma al detectarse un obstáculo frontal a una distancia de x_{fmin}, se realiza un cambio en el sentido del giro en la rueda derecha del robot móvil, permitiendo establecer un giro de evasión.

V. CONTROL HÍBRIDO

El control que permite seguir una trayectoria deseada y a su vez la evasión de obstáculos, es un control híbrido que surge de la combinación de los dos controles antes mencionados. La condición que se establece para operar la conmutación de estos dos controles es la siguiente:

$$\left.\begin{aligned} \omega_{l_{speed}} &= \omega_{l_S} \\ \omega_{r_{speed}} &= \omega_{r_S} \end{aligned}\right\} d \leq d_c \tag{16}$$

$$\left.\begin{aligned} \omega_{l_{speed}} &= \omega_{l_T} \\ \omega_{r_{speed}} &= \omega_{r_T} \end{aligned}\right\} d > d_c \tag{17}$$

$\omega_{l_{speed}}$ y $\omega_{r_{speed}}$ son las velocidades de la rueda izquierda y derecha, respectivamente, que se entregan al robot, y d_c es un valor umbral.

La prueba de estabilidad del control híbrido consiste en el análisis por separado de cada estrategia, para el caso del control del seguimiento de trayectoria, dicha prueba se puede consultar en [11]. Por otro lado, para el caso de los modelos neuronales, la prueba de estabilidad consiste en garantizar la sincronización entre las neuronas [1]–[3].

VI. SIMULACIÓN Y RESULTADOS

Para la simulación del robot se utiliza el software Matlab-Simulink 2020b con herramientas Robotics System Toolbox y Mobile Robotics Simulation Toolbox. La simulación consiste en colocar a dos robots móviles dentro de un espacio confinado, el robot 1 tiene implementado el control híbrido mientras que el robot 2 únicamente contiene la estrategia de evasión. La tarea es que ambos salgan de un área con una sola salida y posterior a ello naveguen de forma autónoma (sin colisionar) alrededor del resto del mapa, en especial el robot 1 tendrá que desarrollar una trayectoria deseada (en este caso un círculo de radio 2.0[m]) y efectuar una acción de evasión ante un obstáculo estático que se ubica sobre la trayectoria deseada. Por otra parte, el robot 2 navegará de manera autónoma. A continuación, se muestra la sintonización de los parámetros utilizada, aclarando que en este trabajo se emplearon datos del robot móvil Turtlebot3 (r_r radio de las ruedas, b_r separación entre las ruedas, V_{max} velocidad traslacional máxima, y W_{max} velocidad rotacional máxima [13]), ya que se piensa realizar la implementación en un robot móvil de ese tipo.

Los parámetros usados son: $r_r = 33.0[mm]$, $b_r = 89.0[mm]$, $V_{max} = 0.22[m/s]$, $W_{max} = 2.84[rad/s]$, $r_{ciculo} = 2[m]$, $k_1 = k_2 = k_3 = 1.2$, $a_1 = 0.95$; $b_1 = 1.35$, $c_1 = 0.85$, $I_{min} = 3.25$, $I_{max} = 10$, $x_{min} = 0.4[m]$, $x_{max} = 5.0[m]$, $a = 1.0$, $b = 3.0$, $c = 1.0$, $d = 5.0$, $s = 4.0$, $x_0 = -1.6$, $r = 0.005$, $r_d = -0.0021$, $\epsilon = 0.5$, $\gamma_1 = \alpha = 1.0$, $\gamma_2 = k = 10.0$, $I = 3.281$, $x_{fmin} = 0.5[m]$, $p = 0.7$.
Las condiciones iniciales son: $x_1(0) = 3.7[m]$, $y_1(0) = -1.0[m]$, $\theta_1(0) = \frac{\pi}{2}[rad]$, y $x_2(0) = 6.0[m]$, $y_2(0) = 0.0[m]$, $\theta_2(0) = \pi[rad]$. Además, se considera un tiempo de muestreo de $0.05[s]$.

VI-A. *Resultados*

En la Figura 2 se puede observar cómo fue la evolución de las trayectorias deseadas del robot 1 (trayectoria color magenta) y el robot 2 (trayectoria color azul), es importante notar que posterior a la salida del rectángulo central por parte del robot 1, este trata de converger a la trayectoria deseada (trayectoria color rojo), en la cual se monta después de evadir el obstáculo estático que se colocó en medio con el fin de poner a prueba el control híbrido. En [14] se halla un hipervínculo, el cual dirige hacia un video que muestra de forma animada, la evolución de las trayectorias de los robots móviles que se consideran en la simulación numérica. De esta manera, se aprecian las evasiones realizadas por parte de ambos robots.

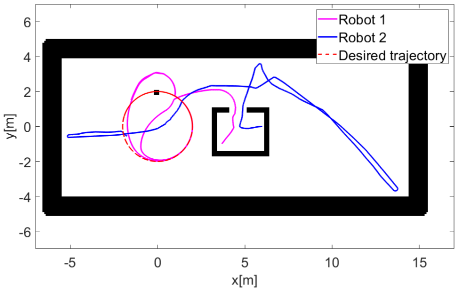


Figura 2. Trayectorias de los robots móviles.

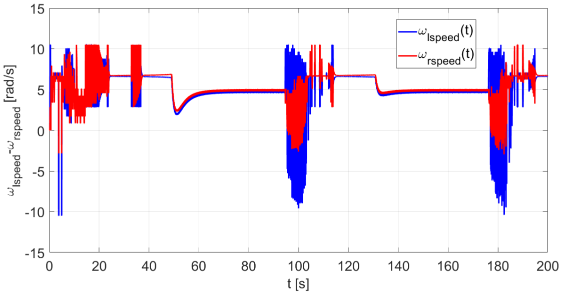
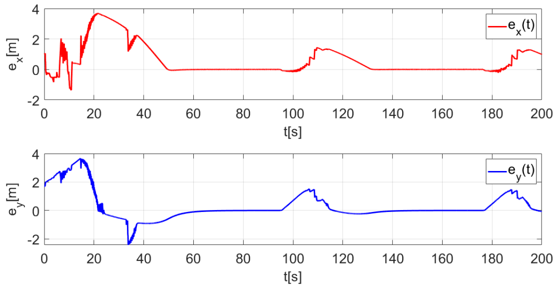
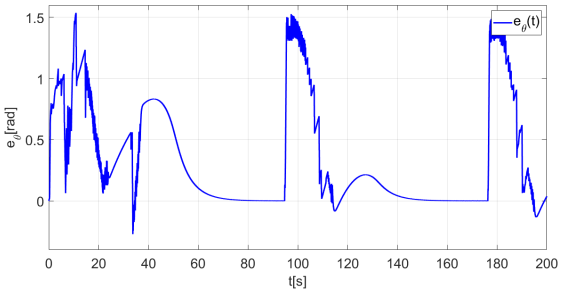


Figura 3. Velocidades de las ruedas $\omega_{l_{speed}}, \omega_{r_{speed}}$ del robot 1.



(a) Errores de posición en x e y. Robot 1.



(b) Error de posición en θ. Robot 1.

Figura 4. Errores de posición. Robot 1.

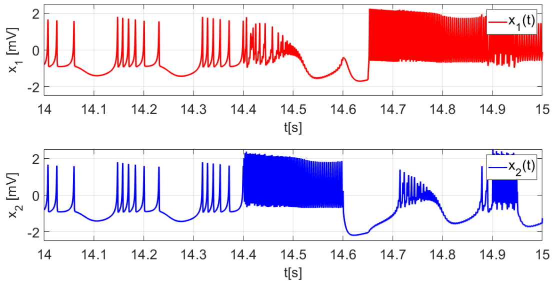


Figura 5. Zoom en la evolución de los potenciales de membrana x_1, x_2, que actúan en el control híbrido. Robot 1.

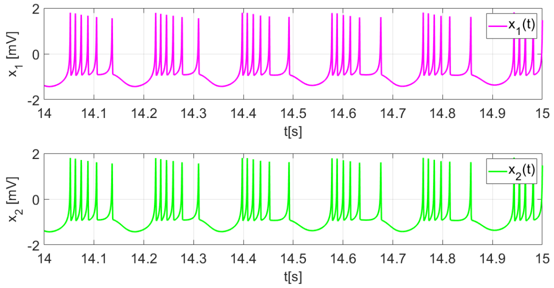


Figura 6. Zoom en la evolución de los potenciales de membrana x_1, x_2, que actúan en el robot 2.

Las señales de las velocidades de rueda $\omega_{l_{speed}}$ y $\omega_{r_{speed}}$ del robot 1 se muestran en la Figura 3, donde se puede observar como en un primer momento varían, siendo una mayor que la otra para efectuar giros de evasión, es decir el robot 1 es operado con la estrategia de evasión basada en el modelo neuronal y no tiende a la trayectoria deseada, sin embargo, después de que el robot 1 sale del área rectangular, es evidente como las velocidades de las ruedas izquierda y derecha son similares y constantes, esto significa que el robot 1 es operado por el control de seguimiento de trayectoria y tiende a la trayectoria deseada, la velocidad de la rueda derecha es ligeramente mayor que la velocidad de la rueda izquierda para que el robot móvil genere la trayectoria circular. Otra forma de apreciar el resultado que aporta el control híbrido es viendo las Figuras 4a y 4b que corresponden a los errores de posición. En la Figura 4a de igual forma, se nota como no se tiende a una convergencia directamente debido a que el robot 1 se encuentra dentro del rectángulo central junto al robot 2, posterior a ello se ve como en un instante de tiempo cercano a 95[s] se evade al obstáculo estático (otra evasión similar ocurre en un tiempo cercano a los 175[s]). En la Figura 4b es evidente como el robot 1 varia su orientación para generar las acciones de evasión.

Una forma más clara de notar los resultados aportados por el control basado en el modelo neuronal, es observando los gráficos de la Figura 5, que corresponden a los potenciales de membrana x_1 y x_2, en estos se nota como en el comienzo se tiene una sincronización en las señales x_1 y x_2, esto quiere decir que el robot móvil posee la misma velocidad en ambas ruedas y esto le permite desplazarse en línea recta, por otra parte, hay intervalos de tiempo donde se nota como hay un mayor número de espigueo en un potencial de membrana que en otro, lo cual significa que una velocidad de rueda es mayor que la otra y se efectúa un giro, que puede ser de evasión ó, debido a la trayectoria deseada.

En la Figura 6 se puede apreciar un zoom en la evolución de sus potenciales de membrana x_1 y x_2, aquí se puede notar como en ese instante x_1 y x_2 se encuentran en sincronización, esto significa que el robot 2 posee la misma velocidad en sus ruedas izquierda y derecha y se desplaza en linea recta.

VII. CONCLUSIONES Y TRABAJO A FUTURO

Tanto el control híbrido como el control de evasión basado en el modelo neuronal, han mostrado un resultado aceptable. Otros controles requieren un conocimiento completo o parcial del entorno para generar una planificación previa de una ruta, y así conseguir una navegación autónoma sin colisiones (donde solo se consideran obstáculos estáticos), sin embargo, el esquema tratado en este trabajo le permite al robot móvil navegar de forma autónoma mientras evade obstáculos estáticos o dinámicos sin el conocimiento del entorno. Otro aspecto es que, en otros esquemas de este tipo el robot móvil se detiene cuando detecta un obstáculo frontal, pero un plus de este trabajo es la estrategia de evasión frontal que permite al robot móvil navegar continuamente.
Los resultados de este trabajo dependen de una correcta sintonización de los parámetros del modelo, así como, el primer y segundo par de neuronas deben ser idénticos (configuración ideal). La distancia umbral x_{fmin} permite al robot móvil detectar obstáculos frontales, si este valor se reduce, entonces, el robot móvil puede explorar más área del entorno, pero, probablemente, el robot móvil requiere una mayor velocidad para evitar los obstáculos (recuerde, se utilizan datos de turtlebot3 y se tienen velocidades máximas como limites para la evasión). Hablando de la distancia umbral d_c, esta permite conmutar los controles cuando se detecta un obstáculo, por lo que, si este valor se disminuye, entonces, el robot móvil también requiere una mayor velocidad para evitar los obstáculos. De lo contrario, si el valor d_c se incrementa, entonces, el robot móvil puede detectar fácilmente obstáculos y áreas rectangulares para poder evitarlos (esto es útil cuando el robot móvil sólo está programado con el control basado en el modelo neuronal). Sin embargo, si el robot móvil tiene que seguir una trayectoria deseada, con un valor d_c incrementado, entonces, el robot móvil deja la trayectoria deseada más fácil si detecta un obstáculo que tiene que evitar.

Las ganancias del control de seguimiento de trayectoria k_1, k_2 y k_3 permiten obtener una convergencia más rápida o más lenta a la trayectoria deseada. En particular, si las ganancias de k_1, k_2 y k_3 son pequeñas, el robot móvil podrá converger a la trayectoria deseada lentamente, sin embargo, con esto puede evitar los obstáculos fácilmente.
Acerca de la elección de la ventana de tiempo, esta se realiza con base en el periodo mínimo que hay entre dos picos sucesivos, y tal que la entrega de la medición del número de picos sea lo suficientemente rápida para que el robot móvil cambie a tiempo su velocidad y así pueda efectuar una evasión.

Como trabajo a futuro, se plantea implementar el mismo esquema de evasión en un ambiente de simulación más real como lo ofrece el software de Gazebo, de igual forma, se buscará implementar de manera experimental.

VIII. AGRADECIMIENTOS

Los autores agradecen el apoyo del CONACYT a través del proyecto de ciencia básica A1-S-26123, “Análisis, control y sincronización de sistemas complejos con interconexiones dinámicas y acoplamientos flexibles.”

REFERENCIAS

[1] E. Steur, T. Vromen, and H. Nijmeijer, “Adaptive training of neural networks for control of autonomous mobile robots,” in *Sensing and Control for Autonomous Vehicles*. Springer, 2017, pp. 387–405.
[2] T. Vromen, E. Steur, and H. Nijmeijer, “Training a network of electronic neurons for control of a mobile robot,” *International Journal of Bifurcation and Chaos*, vol. 26, no. 12, p. 1650196, 2016.
[3] G. G. Velasco Equihua and J. P. Ramirez, “Synchronization of hindmarsh-rose neurons via huygens-like coupling,” *IFAC-PapersOnLine*, vol. 51, no. 33, pp. 186–191, 2018, 5th IFAC Conference on Analysis and Control of Chaotic Systems CHAOS 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405896318336590
[4] J. A. J. Lora and A. R. Angeles, “Formación autónoma bio-inspirada para robots diferenciales,” *National Congress of Automatic Control CNCA*, pp. 546–551, 2019.
[5] M. Hank and M. Haddad, “A hybrid approach for autonomous navigation of mobile robots in partially-known environments,” *Robotics and Autonomous Systems*, vol. 86, pp. 113–127, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889015302992
[6] J. Velagic, B. Lacevic, and B. Perunicic, “A 3-level autonomous mobile robot navigation system designed by using reasoning/search approaches,” *Robotics and Autonomous Systems*, vol. 54, no. 12, pp. 989–1004, 2006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889006000947
[7] M. S. Gharajeh and H. B. Jond, “Hybrid global positioning system-adaptive neuro-fuzzy inference system based autonomous mobile robot navigation,” *Robotics and Autonomous Systems*, vol. 134, p. 103669, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889020305091
[8] E. Baklouti, N. B. Amor, and M. Jallouli, “Reactive control architecture for mobile robot autonomous navigation,” *Robotics and Autonomous Systems*, vol. 89, pp. 9–14, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0921889016305413
[9] L. MUTLU and E. UYAR, “Control and navigation of an autonomous mobile robot with dynamic obstacle detection and adaptive path finding algorithm,” *IFAC Proceedings Volumes*, vol. 45, no. 22, pp. 355–360, 2012, 10th IFAC Symposium on Robot Control. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667016336369
[10] R. Chatterjee and F. Matsuno, “Use of single side reflex for autonomous navigation of mobile robots in unknown environments,” *Robotics and Autonomous Systems*, vol. 35, no. 2, pp. 77–96, 2001. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S092188900000124X
[11] P. K. Panigrahi and S. K. Bisoy, “Localization strategies for autonomous mobile robots: A review,” *Journal of King Saud University - Computer and Information Sciences*, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1319157821000550
[12] C. C. de Wit, B. Siciliano, and G. Bastin, *Theory of robot control*. Springer Science & Business Media, 2012.
[13] Std. [Online]. Available: https://emanual.robotis.com/docs/en/platform/turtlebot3/features/
[14] Std. [Online]. Available: https://youtu.be/fEByEFvW8ec

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Modelado y Control para el Seguimiento de Trayectorias de un Robot Móvil AutoMiny 4.0

Jesús Santiaguillo-Salinas
Departamento de Ing. Mecatrónica
Universidad del Papaloapan
Loma Bonita, Oaxaca, México
jsantiaguillo@outlook.com

Hiram N. García-Lozano
Departamento de Ing. Mecatrónica
Universidad del Papaloapan
Loma Bonita, Oaxaca, México
hnetgarcia@unpa.edu.mx

Gerardo Cruz-Herrera
Departamento de Ing. Mecatrónica
Universidad del Papaloapan
Loma Bonita, Oaxaca, México
gerardo.crhe@gmail.com

***Abstract*—Este trabajo está enfocado al control de seguimiento de trayectorias de un robot móvil terrestre en configuración Ackerman, en particular, para el robot AutoMiny versión 4.0. Con la estrategia de control diseñada se logra que un AutoMiny versión 4.0 converja de forma asintótica a una trayectoria preestablecida. Los resultados teóricos obtenidos son validados mediante simulación numérica así como experimentalmente. El programa utilizado para el control del AutoMiny versión 4.0 en los experimentos, se realiza a través de ROS.**

***Index Terms*—Robot Móvil Ackerman, AutoMiny versión 4.0, Seguimiento de Trayectorias, Modelo Cinemático, Validación Experimental, ROS.**

I. INTRODUCCIÓN

Desde que se construyeron los primeros medios de transporte, el deseo de un sistema automático que lo dirija ha estado presente en la mente tanto de diseñadores como de usuarios. Muchos visionarios y empresarios lo han intentado, y aunque las tecnologías necesarias para la conducción autónoma estaban disponibles en 2000, fue hasta el periodo del 2004 a 2012, que se produjeron avances rápidos y sorprendentes tanto en la academia como en la industria [1], [2]. Vehículos autónomos de Google [3], los autos semiautónomos de Tesla [4], el MIG (Made in Germany) [5], [6], entre otros, mantienen su carril y siguen las reglas de tránsito, pueden detectar semáforos, peatones, automóviles en las intersecciones y reaccionar de manera adecuada.

Es en las universidades y centros de investigación donde se presentan avances importantes, donde después de simular innovadoras estrategias, se diseñan prototipos para probar los resultados obtenidos. Existen algunas plataformas desarrolladas para este fin [7], [8], que de ser accesibles, facilitan la implementación de los resultados de las investigaciones sin tener que realizar el diseño y construcción del robot móvil.

El robot AutoMiny versión 4.0 (AutoMiny) es un modelo de vehículo autónomo (escala 1:10) desarrollado en la Freie Universität Berlin, con fines educativos y de investigación [9]. El objetivo de esta plataforma es brindar un robot con el que puedan programar la visión por computadora, el reconocimiento de objetos y la estrategia de navegación necesarios para conducir en una ciudad modelo. El automóvil cuenta con un sistema a bordo con sensores de percepción, de retroalimentación, CPU y GPU, así como LED para emular las luces del automóvil. Está diseñado como un sistema autosuficiente que no requiere sensores externos y puede ser controlado de forma remota o se puede programar para conducir en modo totalmente autónomo.

El robot AutoMiny (y versiones anteriores) ha sido utilizado para realizar diversas investigaciones, por ejemplo, para el desarrollo de algoritmos de reconstrucción de mapas a partir de imágenes [10], en el diseño e implementación de sistemas de navegación autónoma [11]. Otro resultado en el que este robot fue parte indispensable es en la implementación de una metodología basada en redes neuronales convencionales para el control de dirección [12].

En este trabajo se presenta el modelado, simulación e implementación, de un sistema de control para el seguimiento de trayectorias, en un robot móvil terrestre, a ruedas, con sistema de tracción con configuración Ackerman, usando la plataforma AutoMiny, para validar sus resultados teóricos.

Este trabajo está organizado de la siguiente manera, en la introducción se presenta un breve resumen del estado del arte de los vehículos autónomos, se presenta la plataforma AutoMiny, la cual se utiliza para validar los resultados teóricos. En la sección II se detalla el modelo matemático del sistema de locomoción del AutoMiny, el cual corresponde a una configuración Ackerman, 4x2 con tracción trasera. En la siguiente sección se establece en forma clara el problema al cual se le dará solución, detallando las condiciones sobre el tipo de trayectoria a seguir, así como el objetivo principal del trabajo. La sección IV detalla el diseño de la estrategia de control para el seguimiento de trayectorias, estableciendo las condiciones bajo las cuales el error decae asintóticamente. Con esta estrategia de control y utilizando los parámetros reales del robot AutoMiny, se realiza una simulación numérica, cuyos resultados se presentan en la sección V. La sección VI está dedicada a presentar los resultados de la implementación de esta estrategia de control en el AutoMiny. Finalmente, breves conclusiones y trabajos futuros son presentados en la sección VII.

Agradecimiento al fondo FORDECYT/07SE/2018/08/06-04 otorgado al proyecto CONACYT 296737 denominado “CONSORCIO EN INTELIGENCIA ARTIFICIAL”

II. MODELO CINEMÁTICO DEL ACKERMAN

Considere un robot móvil en configuración Ackerman como el que se muestra en la Fig. 1. Este tipo de robot móvil es similar a un automóvil, donde las ruedas delanteras pueden ser direccionables mientras que las traseras se encuentran fijas.

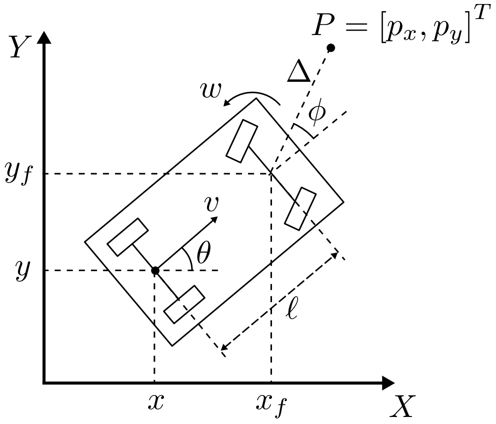


Fig. 1. Esquema del robot móvil Ackerman.

Basados en [13], el modelo cinemático del Ackerman, considerando tracción en las ruedas traseras, está dado por

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ \frac{\tan\phi}{\ell} & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \tag{1}$$

donde $\xi = [x, y]^T$ son las coordenadas cartesianas del eje medio de las ruedas traseras, θ es el ángulo de orientación del cuerpo del robot con respecto al eje X, ϕ es el ángulo de dirección, ℓ es la distancia entre los ejes de las ruedas traseras y delanteras, v es la velocidad del punto medio del eje de las ruedas traseras del robot y w es la velocidad angular de dirección.

Considerando la posición ξ como la salida a controlar del sistema (1), tenemos que

$$\dot{\xi} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} = A(\theta) \begin{bmatrix} v \\ w \end{bmatrix} \tag{2}$$

donde $A(\theta)$ es la matriz de desacoplamiento la cual es singular. Por esta razón, para evitar singularidades en la ley de control, se estudia la cinemática de un punto P, el cual se toma como un punto representativo del robot en lugar del punto medio del eje de las ruedas traseras y cuyas coordenadas están dadas por

$$P = \begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} x_f + \Delta\cos(\theta+\phi) \\ y_f + \Delta\sin(\theta+\phi) \end{bmatrix} \tag{3}$$

con $\Delta \neq 0$ y donde $\xi_f = [x_f, y_f]^T$ son las coordenadas cartesianas del punto medio del eje de las ruedas delanteras dadas por

$$\begin{bmatrix} x_f \\ y_f \end{bmatrix} = \begin{bmatrix} x + \ell\cos\theta \\ y + \ell\sin\theta \end{bmatrix}. \tag{4}$$

La cinemática del punto P está dada por

$$\begin{bmatrix} \dot{p}_x \\ \dot{p}_y \end{bmatrix} = A(\theta, \phi) \begin{bmatrix} v \\ w \end{bmatrix} \tag{5}$$

donde

$$A(\theta,\phi) = \begin{bmatrix} \cos\theta - \tan\phi(\sin\theta + \Delta\sin\delta/\ell) & -\Delta\sin\delta \\ \sin\theta - \tan\phi(\cos\theta + \Delta\cos\delta/\ell) & \Delta\cos\delta \end{bmatrix} \tag{6}$$

con $\delta = \theta + \phi$, es la nueva matriz de desacoplamiento, la cual es no singular ya que $\det(A(\theta,\phi)) = \Delta/\cos\phi \neq 0$ debido a las restricciones del ángulo de dirección $-\frac{\pi}{2} < \phi < \frac{\pi}{2}$ en la mayoría de los casos prácticos.

El modelo cinemático obtenido en esta sección, es el que se empleará en el diseño de la estrategia de control para el robot AutoMiny.

III. PLANTEAMIENTO DEL PROBLEMA

Sea $m(t) = [m_x(t), m_y(t)]^T$ una trayectoria preestablecida continuamente diferenciable. El objetivo de este trabajo es diseñar una estrategia de control $u = f(P(t))$ para un robot AutoMiny tal que

- Se logre el seguimiento asintótico de una trayectoria deseada (Control de Seguimiento), i.e.

$$\lim_{t\to\infty} (P(t) - m(t)) = 0.$$

IV. ESTRATEGIA DE CONTROL

Definiendo las variables auxiliares de control $u = [u_x, u_y]^T$ es posible obtener una estrategia para controlar la posición del punto P como

$$\begin{bmatrix} v \\ w \end{bmatrix} = A(\theta,\phi)^{-1} \begin{bmatrix} u_x \\ u_y \end{bmatrix} \tag{7}$$

donde $A(\theta,\phi)^{-1}$ es la inversa de la matriz de desacoplamiento. El sistema en lazo cerrado (5) y (7) produce

$$\dot{P} = u. \tag{8}$$

Para resolver el problema del seguimiento de trayectorias, proponemos una estrategia de control auxiliar

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} -k_x(p_x - m_x) + \dot{m}_x \\ -k_y(p_y - m_y) + \dot{m}_y \end{bmatrix} \tag{9}$$

donde $\dot{m}(t) = [\dot{m}_x(t), \dot{m}_y(t)]^T$ es la velocidad de la trayectoria, k_x y k_y son las ganancias de control.

Proposición 1: Considere el sistema (5) y la ley de control (7), (9). Suponga que $k_x, k_y > 0$. Entonces, en el sistema en lazo cerrado (5), (7), (9) el robot converge a la trayectoria deseada, i.e. $\lim_{t\to\infty}(P(t) - m(t)) = 0$.

Demostración. Definimos los errores del sistema como

$$\begin{bmatrix} e_x \\ e_y \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p_x \\ p_y \end{bmatrix} - \begin{bmatrix} m_x(t) \\ m_y(t) \end{bmatrix}. \tag{10}$$

La dinámica del error está dada por

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \end{bmatrix} = \begin{bmatrix} -k_x & 0 \\ 0 & -k_y \end{bmatrix} \begin{bmatrix} e_x \\ e_y \end{bmatrix} \tag{11}$$

y dado que $k_x, k_y > 0$ la dinámica del sistema en coordenadas del error es asintóticamente estable. ■

V. SIMULACIÓN NUMÉRICA

Los resultados de una simulación numérica utilizando la estrategia de control (7), (9) se presentan a continuación. El punto P a controlar se encuentra a una distancia $\Delta = 0.1$m al frente del punto medio del eje de las ruedas delanteras y la distancia entre los ejes trasero y delantero es $\ell = 0.26$m. Las ganancias de control son $k_x, k_y = 1$. La trayectoria deseada es un círculo dado por

$$m_x = \cos\left(\frac{2\pi t}{T}\right)$$
$$m_y = \sin\left(\frac{2\pi t}{T}\right)$$

con periodo $T = 60$s. Se considera que el robot parte del reposo con condiciones iniciales $x_0 = 0.25$m, $y_0 = -0.25$m, $\theta_0 = 0$rad y $\phi_0 = 0$rad. Para la simulación se utilizó el software MATLAB/Simulink R2018b, con ode4 (Runge-Kutta), un tiempo de muestreo fijo de 0.01s y un tiempo de simulación de 60s.

La Fig. 2 muestra el movimiento realizado por el robot en el plano. Se observa un buen desempeño del robot al realizar el seguimiento de la trayectoria deseada. La Fig. 3 muestra los errores de posición del robot. Se puede ver como a partir de los 5s, dichos errores convergen a cero. La Fig. 4 muestra el ángulo de orientación del cuerpo del robot θ y el ángulo de dirección de las ruedas ϕ durante el seguimiento de la trayectoria. Se puede ver como el ángulo de orientación del cuerpo del robot tiene un crecimiento lineal a lo largo del tiempo, mientras que el ángulo de dirección de las ruedas permanece constante en el tiempo, con ligeras variaciones al inicio. Ambos ángulos se comportan acorde a la trayectoria deseada. La Fig. 5 muestra las señales de control v y w del robot. Debido a la trayectoria deseada, ambas señales convergen a un valor constante.

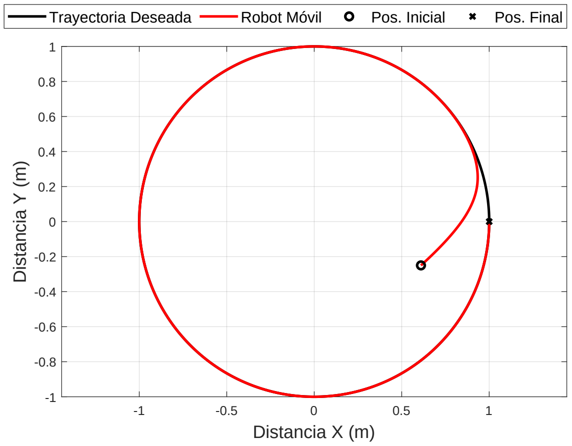


Fig. 2. Trayectoria del robot en el plano.

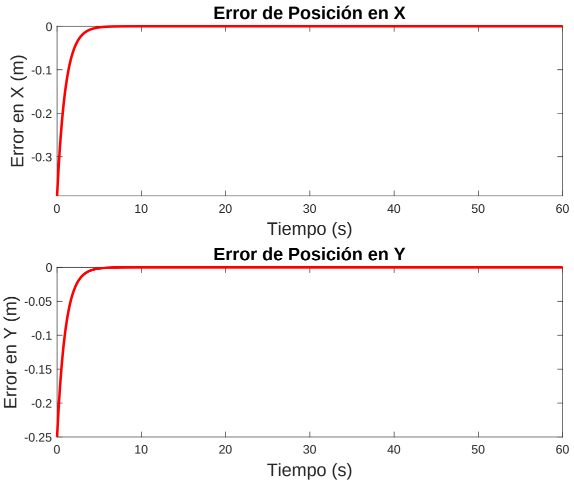


Fig. 3. Errores de posición del robot.

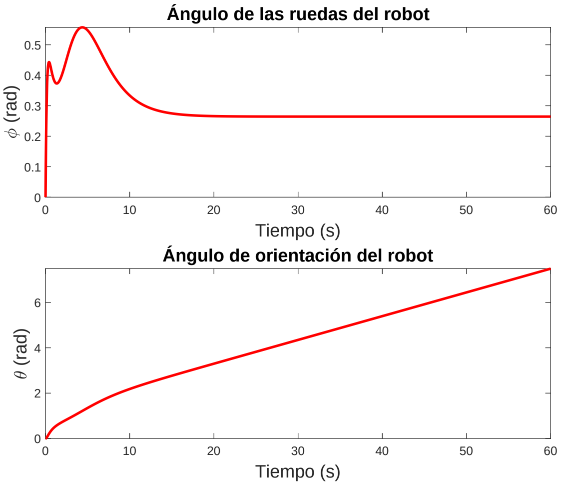


Fig. 4. Ángulos de orientación y dirección del robot.

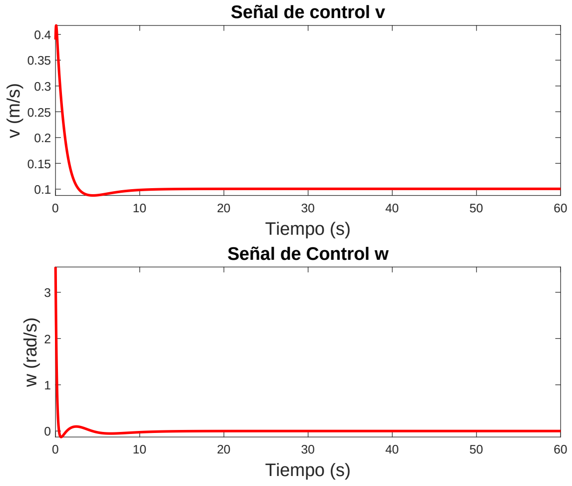


Fig. 5. Señales de control del robot.

VI. RESULTADOS EXPERIMENTALES

A. *AutoMiny*

Para la validación experimental de los resultados teóricos obtenidos, se utilizó el AutoMiny mostrado en la Fig. 6. El AutoMiny es un vehículo a escala 1:10, provisto de una cámara estéreo de infrarrojos, un LIDAR y una Unidad de Medición Inercial (IMU), cuyo funcionamiento es coordinado por ROS Melodic Morenia en Ubuntu 18.04. La forma de realizar el procesamiento es modular, con un sistema dedicado al control y una computadora Intel NUC.

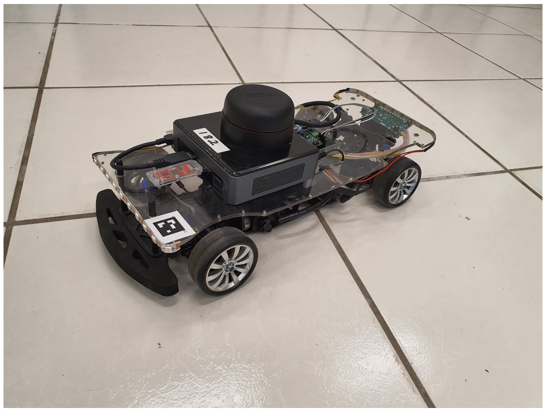


Fig. 6. AutoMiny versión 4.0.

El controlador está compuesto por un Arduino Nano, el cual recibe la información de velocidad y dirección, proporcionada por un encoder incremental y la salida analógica del servomotor, respectivamente. La información es recibida por la computadora NUC a través de un protocolo de comunicación en serie. La computadora NUC recibe los datos provenientes de la placa controladora, el LIDAR, la IMU y la Cámara estéreo para ejecutar los algoritmos de control programados.

Utilizando Phyton en ROS, se realizaron pruebas para la activación del motor de tracción y el de dirección, así como para la lectura de los sensores. Con base en estas pruebas, se estableció que la velocidad máxima/mínima del vehículo es ±2.5m/s, el ángulo máximo de giro de las ruedas 0.512rad y el mínimo -0.498rad. Estos límites se utilizan para que en la implementación de la estrategia de control sean considerados a fin de no saturar a ninguno de los actuadores.

Cabe mencionar que, a partir de las pruebas antes mencionadas, también se supo que a velocidades entre -0.06m/s y 0.06 m/s se entra en la zona muerta del motor de tracción, por lo que a velocidades entre este rango, el AutoMiny ya no alcanza a desplazarse, permaneciendo detenido.

Antes de usar la trayectoria circular establecida en la simulación, se realizaron experimentos con trayectorias lineales y cuadráticas, con la finalidad de sintonizar las ganancias del controlador.

Cabe señalar que las entradas de control obtenidas son las velocidades longitudinal v y angular w. Sin embargo, las entradas del robot AutoMiny son la velocidad longitudinal v y el ángulo de dirección de las ruedas ϕ, por lo que se realiza una integración del control w.

B. *Experimentos*

A continuación se muestran los resultados experimentales obtenidos utilizando la estrategia de control (7), (9). El punto P a controlar se encuentra a una distancia $\Delta = 0.1$m al frente del punto medio del eje de las ruedas delanteras y la distancia entre los ejes trasero y delantero es $\ell = 0.26$m.

a) Experimento 1: En un primer experimento se utilizó una parábola como trayectoria deseada, dada por

$$\begin{aligned} m_x &= 0.05t \\ m_y &= 2\left(m_x\right)^2 \end{aligned}$$

Las ganancias de control son $k_x = 1.25$ y $k_y = 1.5$.

La Fig. 7 muestra el movimiento del AutoMiny en el plano. Se observa como el robot realiza el seguimiento de la trayectoria deseada. La Fig. 8 muestra los errores de posición del AutoMiny. Dichos errores oscilan en una vecindad cercana a cero.

La Fig. 9 muestra las señales de control del AutoMiny. Se puede apreciar que la señal de control v entra a la zona muerta del motor ($v < 0.06$m/s), lo que ocasiona que el robot se detenga en esos instantes, se incremente el error de posición y el control crezca hasta volver a salir de la zona muerta, dando lugar a las oscilaciones mostradas tanto en el control como en el error de posición.

La Fig. 10 muestra el ángulo de orientación del cuerpo del robot θ y el ángulo de dirección de las ruedas ϕ durante el seguimiento de la trayectoria, tanto el leído de los sensores, como el enviado como entrada de control (integral). Se aprecia que el ángulo de las ruedas proveniente de la integral de la señal de control w, alcanza su valor máximo de 0.512rad, permaneciendo saturado entre los segundos 6 y 9. Esto también se ve reflejado en el valor del ángulo proveniente de los sensores.

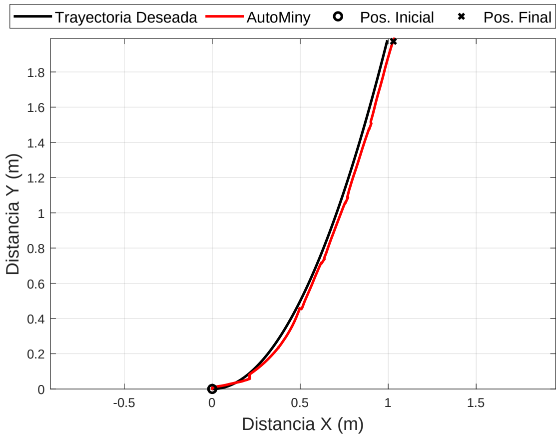


Fig. 7. Trayectoria del robot en el plano.

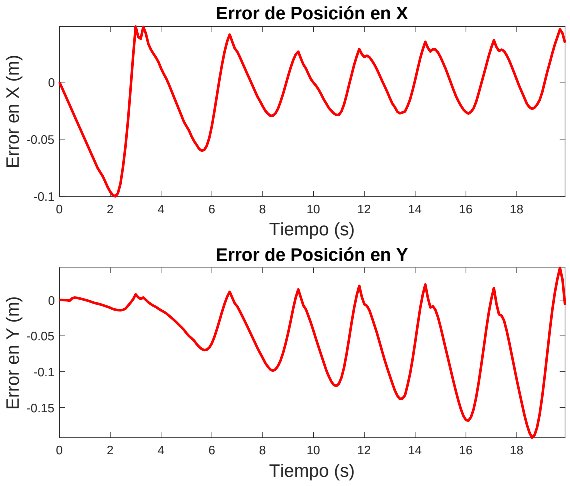


Fig. 8. Errores de posición del robot.

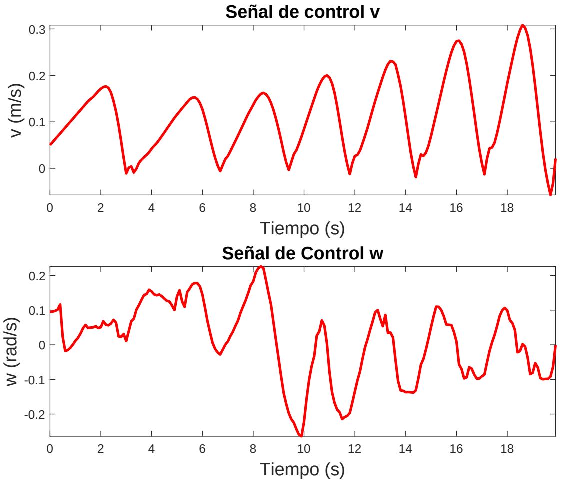


Fig. 9. Señales de control.

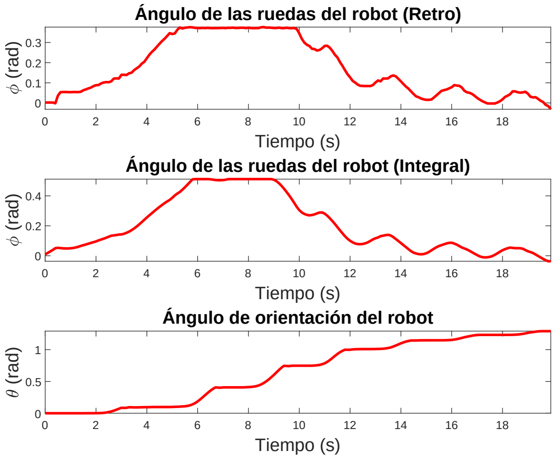


Fig. 10. Ángulos de orientación y dirección del robot.

b) Experimento 2: En un segundo experimento, la trayectoria deseada es un círculo, dada por

$$m_x = 1.0 - r\cos\left(\frac{2\pi t}{T}\right)$$
$$m_y = r\sin\left(\frac{2\pi t}{T}\right)$$

con $r = 1.0$m y un periodo $T = 60$s. Las ganancias de control son $k_x = 1.25$ y $k_y = 1.5$.

La Fig. 11 muestra el movimiento del AutoMiny en el plano. Se observa como el robot realiza el seguimiento de la trayectoria deseada. La Fig. 12 muestra los errores de posición del AutoMiny. Dichos errores oscilan en una vecindad cercana a cero.

La Fig. 13 muestra las señales de control del AutoMiny. Al igual que en el experimento anterior, la señal de control v entra a la zona muerta del motor ($v < 0.06$m/s), teniendo el mismo comportamiento de detenerse y volver avanzar al aumentar v, provocando las oscilaciones tanto en el control como en el error de posición.

La Fig. 14 muestra el ángulo de orientación del cuerpo del robot θ y el ángulo de dirección de las ruedas ϕ durante el seguimiento de la trayectoria, tanto el leído de los sensores, como el enviado como entrada de control (integral).

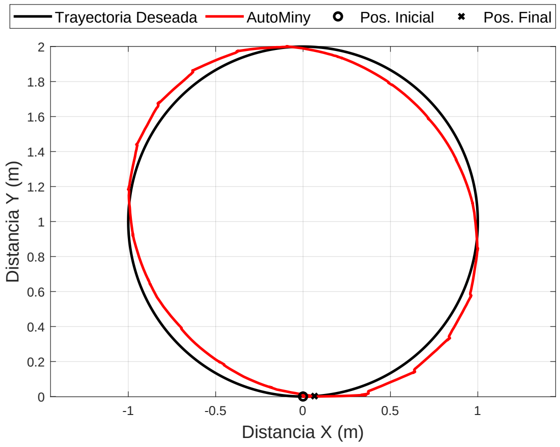


Fig. 11. Trayectoria del robot en el plano.

VII. CONCLUSIONES Y TRABAJO FUTURO

Se obtiene un modelo cinemático de un robot móvil Ackerman y se diseña una estrategia de control para el seguimiento de trayectorias teniendo como salida a controlar un punto al frente del robot, fuera del punto medio del eje de las ruedas.

La estrategia de control diseñada es validada experimentalmente en la plataforma AutoMiny. Los experimentos mostrados se realizaron utilizando como retroalimentación datos odométricos obtenidos de los sensores del AutoMiny, IMU, MPU y encoder incremental. Debido al ruido presente en los datos provenientes de los sensores a bordo del AutoMiny, se

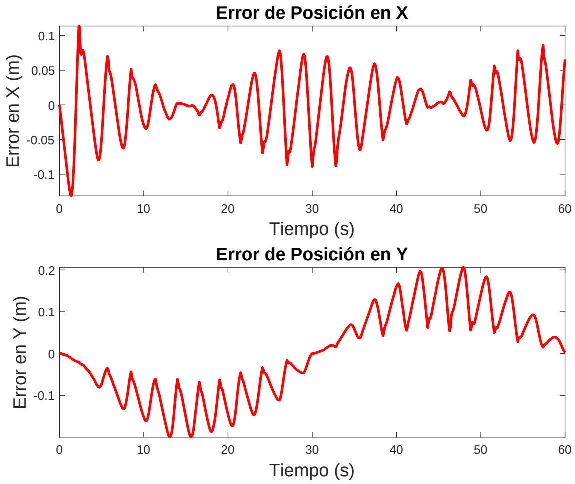


Fig. 12. Errores de posición del robot.

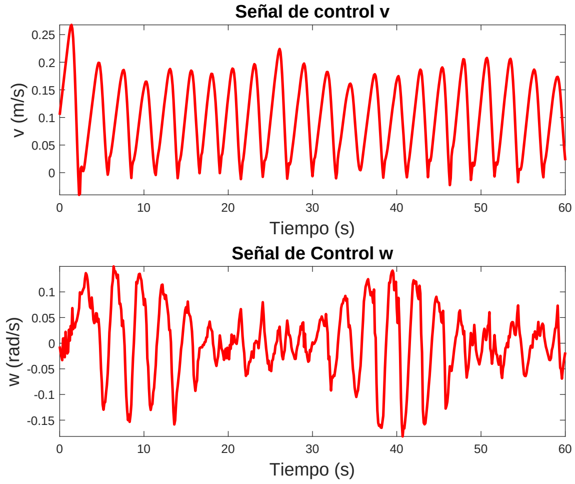


Fig. 13. Señales de control.

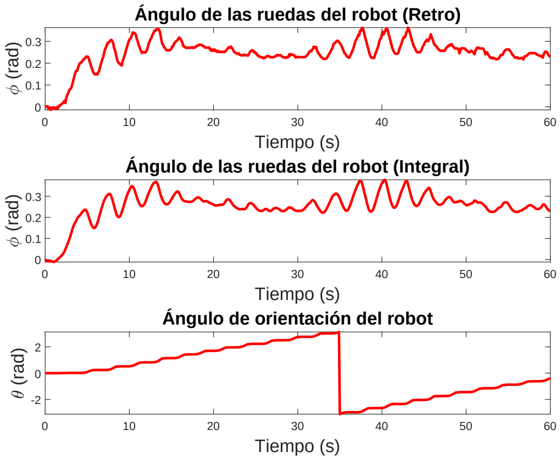


Fig. 14. Ángulos de orientación y dirección del robot.

tiene un error en el seguimiento de la trayectoria. Además de las oscilaciones provocadas por entrar y salir de la zona muerta del motor de tracción. Adicionalmente, consideramos que el tiempo de procesamiento de la ley de control así como la comunicación entre ROS y el AutoMiny es relativamente grande, lo que provoca efectos presentes en los sistemas con retardo. Un análisis de este tipo de sistemas aplicado al AutoMiny podría reducir las diferencias entre la posición deseada y la posición real del robot presente en la parte experimental, sin embargo, es considerado como trabajo a futuro.

A partir de los resultados obtenidos, se pretenden desarrollar nuevas estrategias de control más complejas. Para tener una mejor referencia de la posición, se plantea el uso de un sistemas de visión con cámaras OptiTrack en lugar del uso de la odometría. Se espera que con el uso del sistema de visión, así como de las nuevas estrategias de control se obtengan mejoras en el desempeño del robot tanto en posición como en velocidad.

REFERENCES

[1] S. A. Bagloee et al., “Autonomous vehicles: challenges, opportunities, and future implications for transportation policies,” J. Mod. Transport, vol 24, pp 284–303, 2016.
[2] Stanford University, Artificial Intelligence and Life in 2030: One Hundred Year Study on Artificial Intelligence (AI100). https://ai100.stanford.edu/, Stanford University, September, 2016.
[3] Google, Google Self-Driving Car Project. https://www.google.com/selfdrivingcar/, Google, August, 2016.
[4] M. McHugh, Tesla’s Cars Now Drive Themselves, Kinda. Wired, http://www.wired.com/2015/10/tesla-self-driving-over-air-update-live/, Tesla, October, 2015.
[5] J. Rojo y R. Rojas, Spirit of Berlin: An Autonomous Car for the DARPA Urban Challenge Hardware and Software Architecture. Freie Universität Berlin, June, 2007.
[6] T. Langner et al., “Traffic Awareness Driver Assistance based on Stereovision, Eye-tracking, and Head-Up Display,” In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), Stockholm, Sweden, 2016.
[7] S. Zug et al., “Technical evaluation of the Carolo-Cup 2014 - A competition for self-driving miniature cars,” In Proceedings of the 2014 IEEE International Symposium on Robotic and Sensors Environments (ROSE), pp. 100–105, 2014.
[8] https://github.com/AutoModelCar/AutoModelCarWiki/wiki.
[9] Quick-Start Guide, https://autominy.github.io/AutoMiny/docs/quick-start-guide/.
[10] E. G. Diaz Sarmientos, Generación del mapa métrico de una pista de competencias para la navegación del robot móvil autonomos mini v2.0. Tesis de Maestría, División de Estudios de Posgrado, Maestría en Robótica, Universidad Tecnológica de la Mixteca, 2020.
[11] C. G. Bravo Conejo, Navegación autónoma de un robot tipo automóvil en pista de carreras con obstáculos. Tesis de Maestría, Centro de Investigación en Computación IPN, 2018.
[12] T. D. Vanegas Sánchez, Navegación autónoma de un vehículo a escala por medio de redes neuronales profundas. Tesis de Maestría, Centro de Investigación en Computación IPN, 2018.
[13] A. De Luca et al., Feedback Control of a Nonholonomic Car-like Robot. In: J.-P. Laumond (eds.), Robot Motion Planning and Control, Lectures Notes in Control and Information Sciences, vol. 229, pp. 171–253, Springer-Verlag/Berlin/Heidelberg, ISBN 3-540-76219-1, 1998.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Complete mathematical modeling of a scale autonomous vehicle

González-Miranda, Oscar[1] and Ibarra-Zannatha, Juan Manuel[2]
Automatic Control Departament
CINVESTAV
México City, México
[1]ogonzalez@ctrl.cinvestav.mx, [2]jibarra@cinvestav.mx

***Abstract*—We present a math model more complete than the bicycle model used in the literature. To get it, we use the scale 1:10 vehicle AutoMiny in our experiments. This model takes into account the dynamic of the servo motor and the brushless motor, the Ackerman direction system, the traction system, and the friction. In the end, We obtained a model with visual feedback.**

***Index Terms*—Autonomous vehicles, lateral and logitudinal control, lane keeping**

I. INTRODUCTION

The good performance of a controller also depends on the math model used to represent reality. In the case of autonomous vehicles, the archetype model used for the lateral and longitudinal control (steering and speed control respectively) is the bicycle model [1], [2] which represents the four tires and the Ackerman direction with a fixed tire in the rear axle and a mobile tire in the front axle. The angle between the front tire and the longitudinal axle of the vehicle is named steering angle δ. Based on it, many models have been proposed and they can classify as geometrics, kinematics, and dynamics models [3], [4].

The geometric models are based on the trigonometric problem to take the car from its current pose to a goal position. For example, with the “Pure-Pursuit method” Amidi et al. drove the vehicle “NavLab” at a speed of 8 m/s [5]. Another example is the “Stanley method”, the Stanford Racing Team in cooperation with the Volkswagen Electronics Research Laboratory won the DARPA Grand Challenge in the year 2005. Their vehicle “Stanley” drove across the desert keeping its lane [6], [7]. In spite of their simplicity and easy implementation, these models don’t take into count the dynamic of the motors, the mechanical systems inside the car, and the effects of the friction. For this reason, these models are incomplete.

With respect to kinematic models in [2] P. Corcke proposes a controllable model as long as the speed $v > 0$; however, it requires the measurement of the pose of the car from some static referential. In practice this data not always are available. Alternatively, in [8] Sotelo proposes a nonlinear model based on state variables that can measure on board the car using a vision system.

With his model developed a controller and used it to drove the car “Babieca”. This car got up a maximum sped of 50 Km/h with a position error of ± 25 cm and orientation error of 1°. Although these models to be successful at low speeds; they don’t take into count the forces and torques that appear in the tires. For this reason, when the speed increases, the velocity vector of the car isn’t parallel to the longitudinal axle and these models don’t work.

Finally, the dynamic models of the bicycle incorporate these forces and torques to the state equations and get a more complete model [1], [3]. Using this model Ackermann et al. [9] proposed a PID controller to drove a bus O305 at maximum speed of 20 m/s and with yaw rate of 0.4 rad/s. In spite of this models are more complete than others they also suppose an instantaneous response of the actuators and the mechanical systems.

In this work, we use the autonomous vehicle AutoMiny. This is a scale 1:10 Ackerman vehicle developed by the Free University of Berlin and donated by Dr. Raul Rojas through the German embassy in Mexico. This robot has a RGBD camera, a servo motor to control the steering, a brushless motor to give traction to the tires, an inertial sensor of 6 degrees of freedom, a lidar and a computer Intel NUC8I5BEK [10].

The plant that we want to identify has two inputs: A speed signal $S_\omega(t)$ which takes values between ± 1000, and a steering signal $S_\gamma(t)$ which can take values between ± 1. These signals are sent to their respective motors’ controllers to move the vehicle. In our work, we identified the dynamics of motors and their mechanical systems with friction. The outputs of our plant are the velocity of the car v (measured in cm/s) and the steering angle δ (measured in rad). In previous works, we use these variables and the bicycle model to get the relative position and orientation of the car in the lane, using visual feedback [4], [11]. Concatenating all these models, we obtain the math model of AutoMiny. Fig. 13 shows the diagram block of the system and the subsystems that appear.

II. SERVO MOTOR IDENTIFICATION

The Autominy vehicle uses a servo motor to control the Ackerman steering system. It is brand Adafruit and has an analogic output that gives the measurement of its position. Using the optimal control presented in [4] the car drove in a rectangular path, changing signal S_ω from 50 to 200. At the same time, the steering reference S_γ and the feedback M_γ were saved. The plots of M_γ are in Fig. 2 (red).

Using the data of the plot 2(d) and the least-squares method, the servo motor modeled as:

$$\frac{\mathcal{M}_\gamma(s)}{\mathcal{S}_\gamma(s)} = \frac{b}{s+a} \tag{1}$$

With estimated parameters $\hat{a}_0$ and $\hat{b}_0$. Later, these parameters and the data in the plot 2(b) were used as initial values in a predictor to adjust them in the iterative form [12]. Fig. 1 shows the block diagram of the predictor. Here a and b are the real parameters, $\hat{a}$ and $\hat{b}$ are the estimate parameters; $\hat{M}_\gamma$ is the estimate output; e is the prediction error and α is the convergence velocity. Mathematically it's:

$$\dot{M}_\gamma = -aM_\gamma + bS_\gamma \tag{2}$$

$$\dot{\hat{M}}_\gamma = -\hat{a}M_\gamma + \hat{b}S_\gamma - \alpha e \tag{3}$$

$$e = \hat{M}_\gamma - M_\gamma \tag{4}$$

To determinate the adaptation law and the stability of the predictor, first derive (4):

$$\dot{e} = -\alpha e + \phi^T\tilde{\theta} \quad \text{with } \phi = \begin{pmatrix} M_\gamma & S_\gamma \end{pmatrix}^T \text{ and } \tilde{\theta} = \begin{pmatrix} -\tilde{a} & \tilde{b} \end{pmatrix}^T$$

$\tilde{a} = \hat{a} - a$ and $\tilde{b} = \hat{b} - b$ are called parametric errors. So proposing as Lyapunov function:

$$V = \frac{1}{2}e^2 + \frac{1}{2}\tilde{\theta}^T\Gamma^{-1}\tilde{\theta};$$
$$\Gamma = \Gamma^T > 0 \quad \text{Adaptation gains matrix}$$

We obtain:

$$\dot{V} = -\alpha e^2 + \tilde{\theta}(e\phi + \Gamma^{-1}\dot{\tilde{\theta}})$$

So that $-\dot{V}$ to be a local positive definite function and then the predictor to be asymptotically stable; we need:

$$e\phi + \Gamma^{-1}\dot{\tilde{\theta}} = \overrightarrow{0}$$
$$\Longrightarrow \dot{\tilde{\theta}} = -e\Gamma\phi \quad \text{But: } \dot{\tilde{\theta}} = \dot{\hat{\theta}}$$
$$\therefore \dot{\hat{\theta}} = -e\Gamma\phi \quad \text{Adaptation law} \tag{5}$$

The equations (3), (4) and (5) define, in continous form, how to get the $\hat{a}$ and $\hat{b}$ closest to real values a and b; respectively. Suposing h as sampling period of M_γ and γ_1 and γ_2 the eigen-values of Γ; the parameters $\hat{a}$ and $\hat{b}$ adjust in iterative form:

$$\hat{M}_{\gamma_{k+1}} = (1-\alpha h)\hat{M}_{\gamma_k} + h(\alpha - \hat{a}_k)M_{\gamma_k} + h\hat{b}_k S_{\gamma_k}$$
$$e_k = \hat{M}_{\gamma_k} - M_{\gamma_k}$$
$$\hat{a}_{k+1} = \hat{a}_k + h\gamma_1 e_k M_{\gamma_k}$$
$$\hat{b}_{k+1} = \hat{b}_k - h\gamma_2 e_k S_{\gamma_k}$$

With this method we get $a = 5.9649$ y $b = 2.3134$. The outputs of the math model (1) are comparate with the M_γ measurements in Fig. 2.

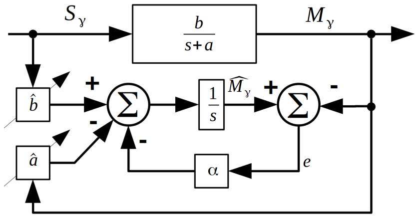


Fig. 1. Predictor's structure used in parameters identification.

III. MODEL OF ACKERMAN STEERING SYSTEM

To model the steering system, first, we fix the position M_γ of the servo motor and move the vehicle until its trajectory form a semi-circle. The plot of M_γ versus the turning radius R shows in Fig. 3. Using the toolbox Curve Fitting of Matlab R2021a, we fit these data to a curve with the form:

$$R = \frac{P}{|tan(M_\gamma + f)|} - r \tag{6}$$

With $P = 3.9\ cm$, $f = -0.0124$ and $r = 20.24\ cm$. On the other hand, the steering angle δ is related with the turning radius as [2]:

$$|\tan(\delta)| = \frac{L}{R} \tag{7}$$

In this case, $L = 0.26\ cm$ is the separation between the rear and front axles. Comparing (6) and (7) we can show for all measurements:

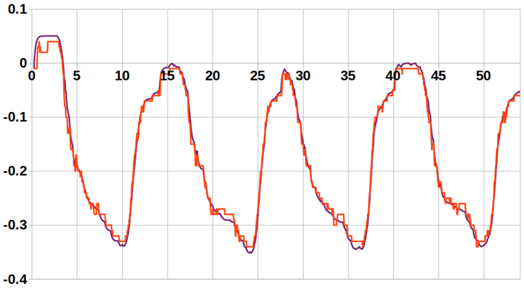
$$\tan(\delta) = \frac{L}{R} = \frac{P}{R-r} = \tan(M_\gamma + f)$$
$$\therefore M_\gamma + f = \delta \tag{8}$$

Therefore, the equation (8) converts the servo motor position M_γ to steering angle δ. In Fig. 4 there is a geometric interpretation of this relation.

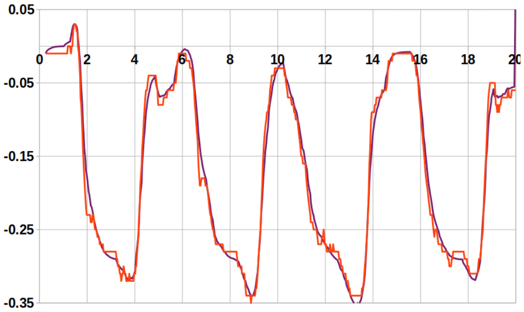
IV. BRUSHLESS MOTOR IDENTIFICATION

The AutoMiny has a brushless motor brand Faulhaber model 2232BX4SC. It gives traction to the tires and has a sensor that measures the angular velocity of the rotor M_ω. Its controller receives as a reference, a signal S_ω which takes values from -1000 to 1000. The positive values move forward the car and the negatives move it in reverse. However, the outputs values don't coincide with the inputs. For example, if we send $S_\omega = 500$; the rotor moves to 60% of its maximum velocity. Fig. 6 compares the normalized input S_ω and the normalized output M_ω. To rectify this, we adjusted a line to the data (red line) and whit this got the equation:

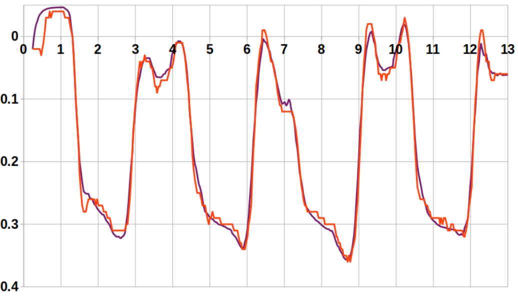
$$R_\omega = mS\omega + h \tag{9}$$



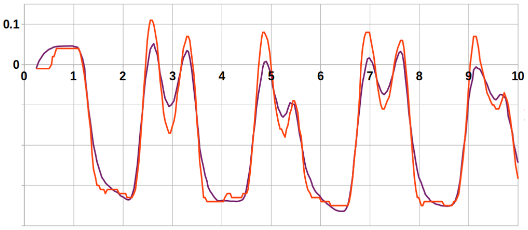
(a) Velocity S_ω=50.



(b) Velocity S_ω=100.



(c) Velocity S_ω=150.



(d) Velocity S_ω=200.

Fig. 2. Comparation between the measurements M_γ (red) and the outputs predicts by the math model (1) (purple), changing the velocities S_ω.

Here $m = 1.2619$ and $h = -0.0249$. With this equation, for example, to get the 50% of the maximum velocity of the rotor, we can calculate the rectified signal R_ω; for $S_\omega = 500$.

In spite of we try to model the friction with the Armstrong model [13], It hasn't a good performance. Fig 5 shows the response predicted by the Armstrong model (in red). Applied the method presented in [14] we obtain a similar respose when S_ω is a ramp but if S_ω is a step function or it varies in time, the respose is different.

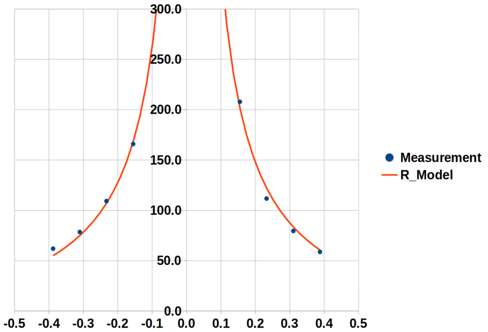


Fig. 3. Graph of the servo position M_γ versus turning radius R in cm. Every point in this graph is the average of three measurements.

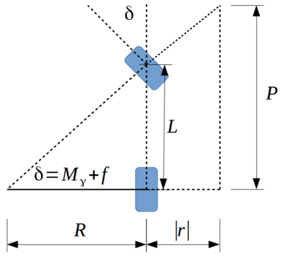


Fig. 4. Geometric interpretation of the equation (8).

So instead, We used Matlab R2021a and the data obtained when the car droves in the rectangular lane with $R_\omega = 150$ to found the delayed model:

$$\frac{\mathcal{M}_\omega(s)}{\mathcal{R}_\omega(s)} = \frac{c_1 s^3 + c_2 s^2 + c_3 s + c_4}{s^4 + d_1 s^3 + d_2 s^2 + d_3 s + d_4} e^{-h_1 s} \qquad (10)$$

With $c_1 = 0.003$, $c_2 = 0.028$, $c_3 = 0.010$, $c_4 = 0.084$, $d_1 = 1.762$, $d_2 = 9.033$, $d_3 = 5.633$, $d_4 = 18.110$ and $h_1 = 0.45$. In Fig 7 shows the comparsion between the measurements M_ω and the signals R_ω.

V. MODEL OF TRACTION SYSTEM

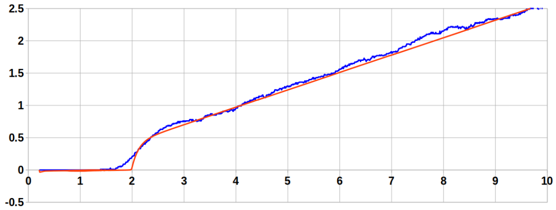
To identify the traction system is necessary to be able to measure the velocity v of the car and the brushless motor velocity M_ω at the same time. So we developed a ground truth system based on visual information.

A. *Visual ground truth system*

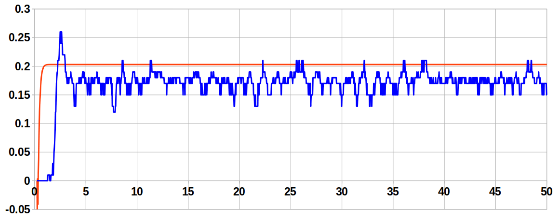
For this system, we use a fish-eye camera brand ELP model USB100W03M-BL170. Offline, this camera has calibrated with sixty photos of a chessboard and the library OpenCV of Python [15]. The camera matrix K and the distortion parameters are:

$$K = \begin{pmatrix} 625.7 & 0.0 & 715.0 \\ 0.0 & 623.3 & 397.8 \\ 0.0 & 0.0 & 1.0 \end{pmatrix}$$

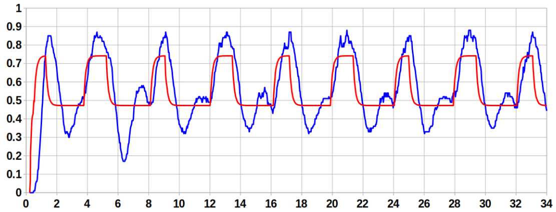
$$\text{dist. coeff} = (-3.3 \times 10^{-1}, 1.2 \times 10^{-1}, 4.2 \times 10^{-4}, \\ -1.6 \times 10^{-4}, -1.9 \times 10^{-2})$$



(a) Output signal when $S_\omega = 10t$.



(b) Output signal when $S_\omega = 50$.



(c) Output signal when S_ω varies from 100 to 150.

Fig. 5. Signal M_ω in blue and output predicted by the Armstrong model in red.

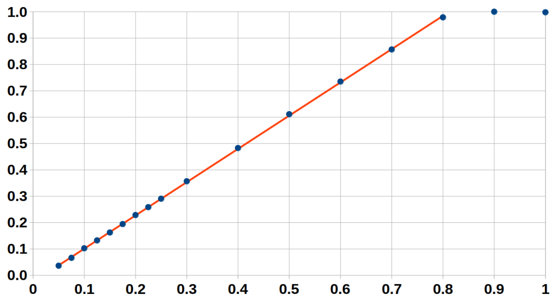


Fig. 6. Comparison between the input S_ω and the output M_ω. Both signals were normalized.

These data allow us to rectify the radial and tangential distortions in images; just like shows in the image 8(b).

Also, we identify the homography matrix H that transforms the pixels in the images to centimeters measured from some reference in the world. This was done by putting four markers on the ground and the same height as the car. Fig. 8(a) shows two of the four markers (green rectangles on white boxes).

Comparing the position in pixels of the markers and their position in centimeters is possible to calculate H [15]. In this case:

$$H = \begin{pmatrix} 3.9 \times 10^{-1} & -1.1 \times 10^{-3} & -1.3 \times 10^{2} \\ 7.5 \times 10^{-3} & 4.2 \times 10^{-1} & -12.2 \\ 3.3 \times 10^{-5} & -3.2 \times 10^{-6} & 1.0 \end{pmatrix}$$

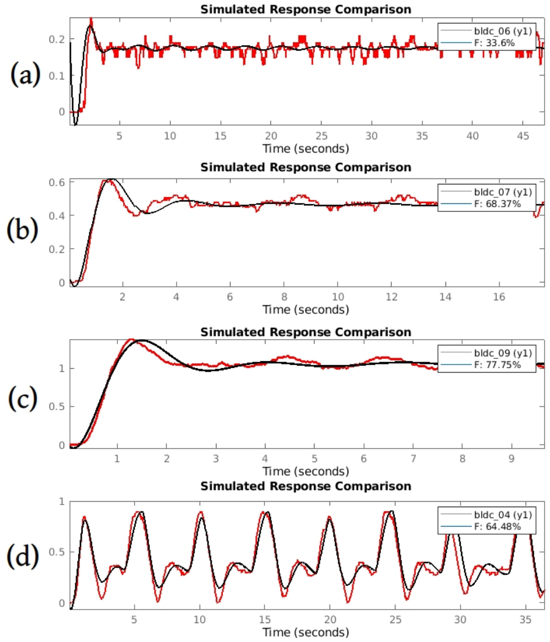


Fig. 7. Output M_ω measured by the sensor's motor (in red) and the output predicted by the math model (10) (in black); when (a) $R_\omega = 50$, (b) $R_\omega = 100$, (c) $R_\omega = 200$ and when (d) R_ω takes values from 75 to 150.

Fig 8(a) also shows a green rectangle on the car. It's at the same height as the other rectangles and is used to get the position of the car. Fig. 9 shows the online method applied.

After obtaining the image, rectify the distortions and apply the homography transform; we segment the green color of the image data. It results in a binary image with a white rectangle. Calculating its first and second moments we can get the centroid of the rectangle and therefore its position in the homography [2].

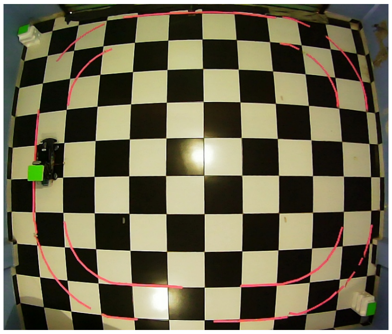
B. Parameter identification

In these experiments, AutoMiny also drove in the rectangular lane mentioned before. We varied the reference S_ω from 50 to 200 in each experiment and saved the data M_ω measured by the brushless motor's sensor and the coordinates (t, x, y) measured by the visual ground truth. With the last, we calculate the instant velocity v in cm/s. However, was necessary to filter the signal with a low-pass filter with cutoff frequency $f_c = 0.5\ Hz$. Fig. 10 compares both signals. On the other hand, Fig. 11 compares the input of the traction system and the velocity output.

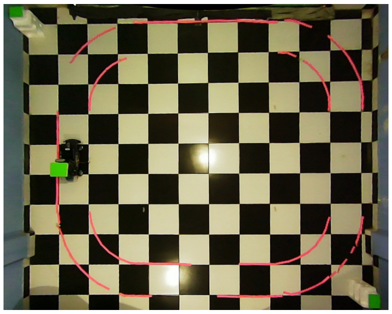
With these data and Matlab R2021a we found that we can model this system as:

$$\frac{\mathcal{V}(s)}{\mathcal{M}_\omega(s)} = \frac{f_1 s^2 + f_2 s + f_3}{s^3 + g_1 s^2 + g_2 s + g_3} e^{-h_2 s} \tag{11}$$

In this case, $f_1 = 1382.0$, $f_2 = 173.1$, $f_3 = 452.5$, $g_1 = 15.66$, $g_2 = 2.74$, $g_3 = 4.76$ and $h_2 = 1.2$.



(a) Original image.



(b) No radial and tangential distortions.

Fig. 8. Image processing.

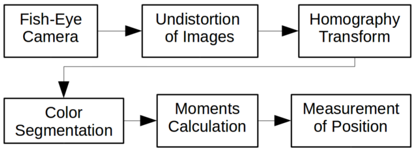


Fig. 9. Vision system to measure the position of the car.

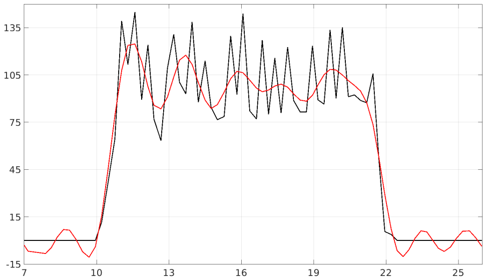


Fig. 10. Signal of Velocity of the car in cm/s without filter (in black) and filtered (in red). The reference is $S_\omega = 200$.

The outputs predicted by this delayed model are compared with the other signals in Fig 12.

VI. COMPLETE MODEL OF THE CAR

To make the lateral control of the car, we presented a visual feedback model based on the bicycle model [4], [11]. This is:

$$\dot{e}_x = ve_\theta + v\frac{L_h}{L}\tan(\delta) \tag{12}$$

$$\dot{e}_\theta = \frac{v}{L}\tan(\delta) \tag{13}$$

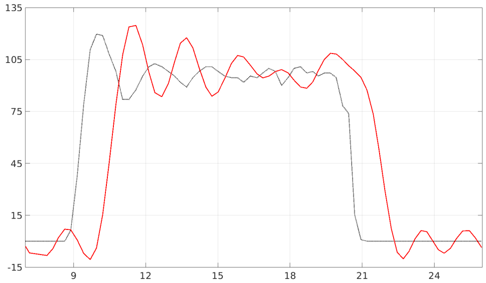


Fig. 11. Comparation between the signal M_ω (in black) and v (in red). The reference is $S_\omega = 200$.

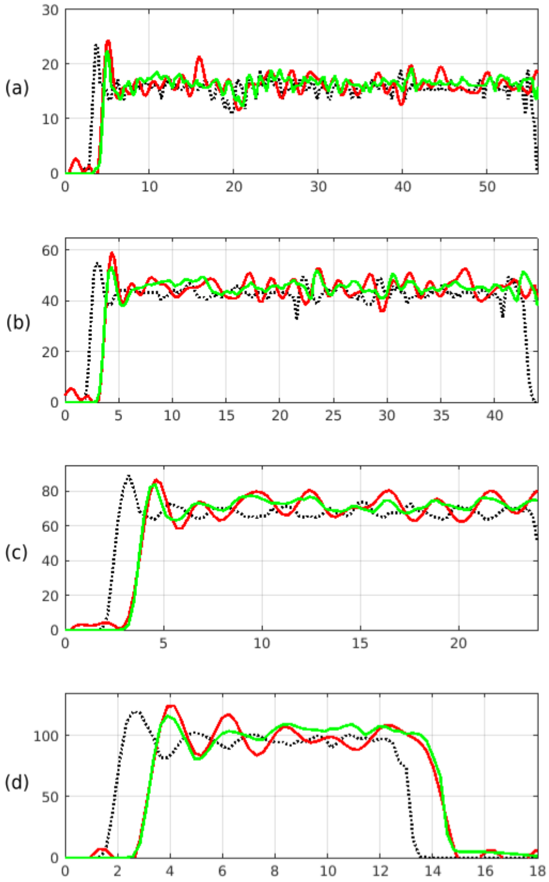


Fig. 12. Comparation between the signal M_ω (in black), v (in red) and the output predicted by the math model (11) (in green). (a) for $S_\omega = 50$, (b) for $S_\omega = 100$, (c) for $S_\omega = 150$ and (d) for $S_\omega = 200$.

L_h is the distance from the rear axle to the base of the homography of the road, L is the separation of the rear and front axles (in this case $L_h \approx L$), e_x is the position error and e_θ is the orientation error (both with respect to the road). If we suppose $\delta = atan(S_\gamma)$ and $S_\omega \propto v$ constant (i.e. ignoring the dynamic of motors, the traction system, and the friction), we can show that (12)-(13) is controllable, observable, and reachable.

So, to complement (12)-(13), we take (2) and (8) and add the state:

$$\dot{\delta} = -a(\delta - f) + bS_\gamma$$

$$\text{defining: } \delta_f \doteq \delta - f$$

$$\dot{\delta}_f = -a\delta_f + bS_\gamma \tag{14}$$

While the equations (12)-(13) describe the dynamic of the position and orientation errors; the equation (14) models the steering angle of the car as a response to the input S_γ (including the dynamic of the servomotor and the Ackerman direction). On the other hand, it's more dificult get the dynamic of the state v from S_ω because the input have to be rectify with (9) and then we need obtain its state representation from (11) and (10). In Fig. 13 there is the complete block diagram of the plant. The signals S_γ and S_ω are the inputs and the states e_x and e_θ are outputs that we can measure from the car with visual feedback.

VII. CONCLUSIONS

We obtained a more complete model than the bicycle model found in the literature. It's a model with visual feedback and has three parts. One for the pose errors relative to the road, the other for the dynamic of the servo and the Ackerman direction, and the other for the BLDC motor, the traction system, and the friction. These experiments let's concluded the last two can be treated as independent. In particular, our model of the BLDC motor with friction is more precise than Armstrong's model but it results in a delay system of seventh order with five zeros. So to control the velocity of the car we recommend explore time-delay control techniques.

REFERENCES

[1] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[2] P. Corke, *Robotics, vision and control: fundamental algorithms in MATLAB® second, completely revised*. Springer, 2017, vol. 118.

[3] J. M. Snider *et al.*, "Automatic steering methods for autonomous automobile path tracking," *Robotics Institute, Pittsburgh, PA, Tech. Rep. CMU-RITR-09-08*, 2009.

[4] O. González-Miranda, "Modelado y control de un vehículo autónomo," Master's thesis, CINVESTAV, México City, México, 12 2019.

[5] O. Amidi and C. E. Thorpe, "Integrated mobile robot control," in *Mobile Robots V*, vol. 1388. International Society for Optics and Photonics, 1991, pp. 504–523.

[6] S. Thrun, M. Montemerlo, H. Dahlkamp, D. Stavens, A. Aron, J. Diebel, P. Fong, J. Gale, M. Halpenny, G. Hoffmann *et al.*, "Stanley: The robot that won the darpa grand challenge," *Journal of field Robotics*, vol. 23, no. 9, pp. 661–692, 2006.

[7] R. D. H. Beleño, G. B. Vítor, J. V. Ferreira, and P. S. Meirelles, "Planeación y seguimiento de trayectorias de un vehículo terrestre con base en el control de dirección en un ambiente real," *Scientia et technica*, vol. 19, no. 4, pp. 407–412, 2014.

[8] M. A. Sotelo, "Lateral control strategy for autonomous steering of ackerman-like vehicles," *Robotics and Autonomous Systems*, vol. 45, no. 3-4, pp. 223–233, 2003.

[9] J. Ackermann, J. Guldner, W. Sienel, R. Steinhauser, and V. I. Utkin, "Linear and nonlinear controller design for robust automatic steering," *IEEE Transactions on Control Systems Technology*, vol. 3, no. 1, pp. 132–143, 1995.

[10] F. U. Berlin. (2020) Autominy. [Online]. Available: https://autominy.github.io/AutoMiny/

[11] O. González-Miranda, "Potential field-based preview control of an autonomous vehicle using visual feedback," in *Memorias del COMRob 2019*. Asociación Mexicana de Robótica e Industria A.C, 2019, pp. 98–103.

[12] K. S. Narendra, *Adaptive and learning systems: theory and applications*. Springer Science & Business Media, 2013.

[13] B. Armstrong-Helouvry, *Control of machines with friction*. Springer Science & Business Media, 2012, vol. 128.

[14] R. Kelly, J. Llamas, and R. Campa, "A measurement procedure for viscous and coulomb friction," *IEEE Transactions on instrumentation and measurement*, vol. 49, no. 4, pp. 857–861, 2000.

[15] W. Garage. (2000) Camera calibration. [Online]. Available: https://docs.opencv.org/master/dc/dbb/tutorial_py_calibration.html

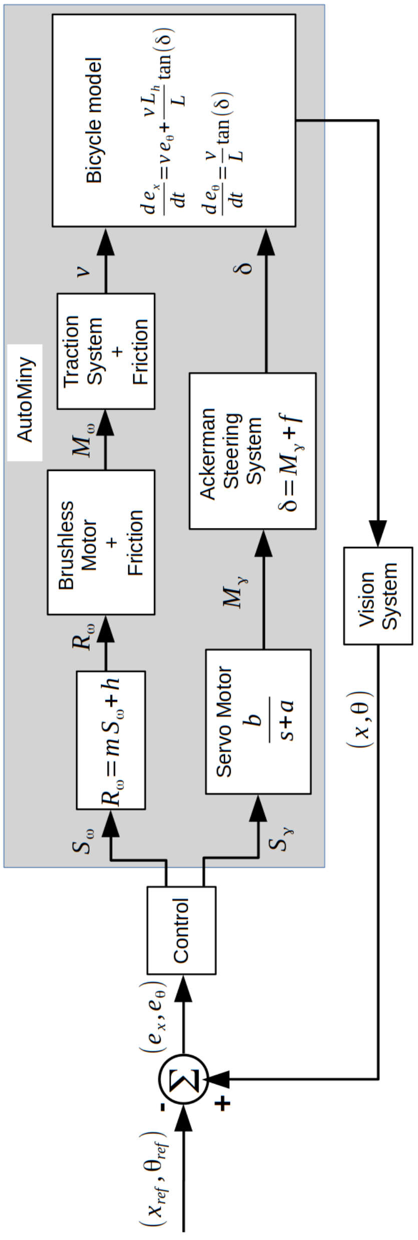


Fig. 13. Comparation between the signal M_ω (in black), v (in red) and the output predicted by the math model (11) (in green). (a) for $S_\omega = 50$, (b) for $S_\omega = 100$, (c) for $S_\omega = 150$ and (d) for $S_\omega = 200$.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

An experimental comparison of scattering-based time delay compensation schemes

R. Peredo-Cabada*, R. de J. Portillo-Vélez** J. A. Vásquez-Santacruz and Luis F. Marín-Urías
Facultad de Ingeniería Eléctrica y Electrónica, Universidad Veracruzana Veracruz, México.
* peredocr@hotmail.com, ** rportillo@uv.mx

Abstract*— The presence of time delays in the communication channel of bilateral teleoperation systems is a risk for the stability and performance of the system. Many solutions with a scattering wave variable codification approach have been proposed to solve this problem. It is hard to find in the literature fair comparisons of the proposed approaches for time delay compensation. In this work, different wave variable schemes are evaluated in a master-slave teleoperation testbed. The experimental results show the advantages and weaknesses of each scheme according to several parameters and considering the final application of the system.

***Keywords*—Time delay, teleoperation, wave variables.**

I. INTRODUCTION

The teleoperation systems allow humans to perform tasks and manipulate objects in remote places, also, they allow the operator to receive information from the environment. They are used in several application fields, such as surgeries, space explorations, submarine operations, etc. The structure of a teleoperation system is composed of a master robot which is manipulated by a human operator, a slave robot that follows the master's actions in another location and the communication channel, where the instructions and the feedback of the system are transmitted. For the correct following of the instructions sent by the master robot, a PI velocity controller (which is analog to a PD position controller) must be implemented on the slave robot side. The controller receives a signal as a velocity reference from the master and it must produce a force signal for the slave robot, which allows to approximate a velocity response similar to the master robot. The communication becomes bilateral when the slave sends a feedback to the human operator. Depending on the system architecture, the feedback could be a current, force or torque signal. With this information, the operator can have a sense of telepresence and control the slave robot trusting in his haptic sensations, besides the vision. From a control engineering point of view [1], the stability of the system and the capability to provide the operator a telepresence sensation are the two most important factors to optimize in a teleoperation system.

The presence of delays in the communication channel between the master and the slave robots could represent a risk to the stability of the system. Thus, the main goal of a time-delay compensation scheme is to ensure that the system never lose its stability. The time delay in the bilateral communication could be caused by different factors, for example, the distance between the robots or the communication channel type. When the delayed signal arrives to the other side of the system, the actions will be executed with obsolete information, originating instability in the system. Also, the increment in the magnitude of the delay increases the speed with which the teleoperation system loses its stability [2], [3].

To cope with the effects produced by the time delays in the communication channel, different authors have proposed several bilateral teleoperation control schemes based in wave variable or scatter codification. This approach is based on the passivity theory, which states that if the output energy flow in the system is more than the input, then the system is passive and stable [1], [4].

Since the first time that the scattering approach was presented by Spong in 1989, several proposals had been reported. An architecture with the wave variable approach was proposed by G. Niemeyer and J. Slotine [5], it consist in a codification in each side of the communication channel. The signal from one robot is converted in wave variable before being sent through the communication channel and decoded when arriving to the other robot. This scheme shows three different paths that the information follows before arriving to its destiny [6]. These paths provoke that the incoming wave signal is being reflected in the outgoing wave, disturbing the information, affecting the performance of the system, and causing disorientation for the human operator. This situation is called wave reflections. To reduce this problem, Niemeyer proposed two different options: the implementation of a low-pass filter, which attenuates the high frequency signals in the communication and an impedance matched teleoperator [5]. The use of a low-pass filter significantly reduces the effects of the reflections. However, this option restricts the bandwidth of the teleoperator. In the case of the impedance matched architecture, a previous knowledge of the slave environment is required to model the system and perfectly match the impedance.

To attack these restrictions, L. Bate proposed a novel wave variable architecture [7]. The main characteristic of the new scheme is that the wave variable from the slave side no longer contains velocity information, this means that the wave variable from the master is not being reflected in the feedback path, allowing to significantly reduce the wave reflections in the system. Furthermore, this new method negates any requirement for modeling the remote environment to match the impedance of the system, which mitigates the wave reflection, and does not restrict the teleoperator bandwidth, like the implementation of a low-pass filter in the communication channel. Another characteristic of the system is that when the master manipulator stops por moves slowly, an accurate force feedback is produced, this characteristic is desirable for a move and wait application. On the other hand, when the master manipulator is fast moving, the force feedback is inaccurate [7].

In the scheme proposed by Bate, the master robot velocity disturbs the force information that arrives from the slave robot. Also, the velocity information that is sent from the master robot to the slave robot contains a force component of the feedback from the slave. In the other side of the communication channel, the velocity information that arrives from the master gets disturbed by the force of the salve robot. These influences between the signals limitates the performance of the teleoperator. In 2013, M. Zheng proposed a modification to the Bate's teleoperator [8]. The main characteristic of this scheme is the addition of damping elements in each line of the wave variable conversion. This new addition reduces the influence and dependence between the signals, improving the performance of the system.

Due to the diversity of architectures, this work presents a comparative evaluation between the three options previously discussed. The different schemes are tested in an experimental bilateral teleoperation testbed to identify the advantages and weaknesses of each architecture. The rest of the paper is organized as follows. In section II the schemes are described in detail. Section III explains the testbed used in the experiments. Section IV demonstrates the experimental results and Section V concludes this paper.

II. TIME DELAY COMPENSATION SCHEMES

A. *Architecture proposed by G.Niemeyer*

Niemeyer proposed a control scheme to reduce the effects of delays in the communication channel [5], [6]. Based in the passivity theory, a conversion to a scattering wave variable is applied in the master's velocity and in the slave's force before being sent to the other side of the channel. The resultant block scheme of this transformation is shown in Figure 1, where $\dot{X}m$ is the master's velocity, $\dot{X}ds$ is the desired velocity on the slave robot, Fs is the slave robot force and Fm is the slave force that arrives to the master robot. Note that the PD position controller shown in Fig.1 is equal to a PI velocity controller. The wave variables u and v are represented by equations (1) and (2).

$$u = \frac{b\dot{x} + F}{\sqrt{2b}} \tag{1}$$

$$v = \frac{b\dot{x} - F}{\sqrt{2b}} \tag{2}$$

where b is the characteristic impedance of the system and represents an adjustable parameter. The characteristic impedance affects the system behavior, and its value must be computed with the trial-and-error approach. The scattering wave decodification can be expressed as shown in equations (3) and (4).

$$\dot{x} = \frac{1}{\sqrt{2b}}(u + v) \tag{3}$$

$$F = \sqrt{\frac{b}{2}}(u - v) \tag{4}$$

In equation (2), the wave v contains velocity information, producing a reflection between the incoming and outgoing waves. In Figure 2, the three information paths that exist in this architecture are shown. The first path is created by the master robot, for each movement there is an immediate feedback to the robot, this path is necessary for the wave variable implementation and does not represent a risk to the teleoperator performance.

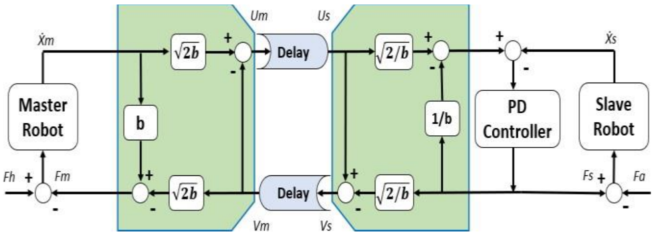


Fig. 1. Niemeyer's architecture.

The third path contains the slave robot feedback, this information is very important and can not be eliminated. Finally, the second path is the wave reflection, it is produced entirely by the wave variable codification and does not contains relevant information for the teleoperator. Also, it could produce disturbs to the system, leading to an incorrect performance [6].

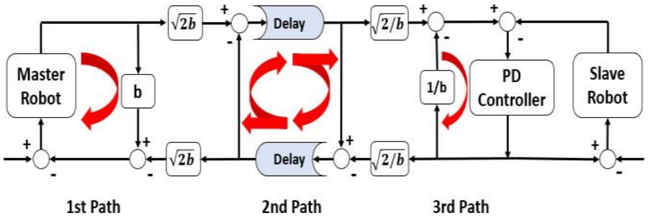


Fig. 2. Information paths of scattering wave-based time delay compensation.

B. *Architecture proposed by L. Bate.*

An alternative scheme to solve the wave variable reflections problems was proposed by L. Bate [7]. By modifying the encoding and decoding structure from the signals, an efficient option can be reached to face wave reflections without knowledge of the slave environment. The scheme proposed by Bate is shown in Figure 3.

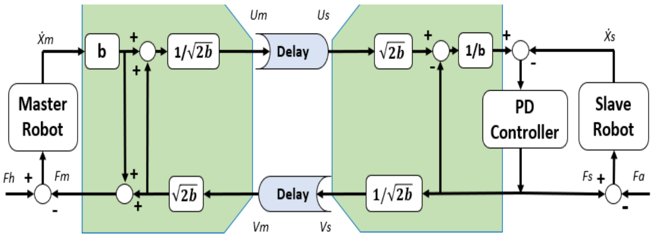


Fig. 3. Bate's architecture.

The equations for the master and slave waves um and vs are presented in equations (5) and (6).

$$u_m = \frac{b\,\dot{x}_m(t) + F_s(t - T)}{\sqrt{2b}} \tag{5}$$

$$v_s = \frac{F_s(t)}{\sqrt{2b}} \tag{6}$$

Equation (6) shows that the v wave variable does not contain information of the velocity; therefore, the outgoing u signal is not being reflected in the incoming wave. With this characteristic, the circular reflections are prevented because of the removal of the second information path, shown in Figure 2. The new equations for velocity and force are expressed as follows in (7) and (8).

$$\dot{x}_s(t) = \dot{x}_m(t-T) + \frac{F_s(t)}{b}(t-2T) - \frac{F_s(t)}{b} \quad (7)$$

$$F_m(t) = b\dot{x}_m(t-T) + F_s(t-2T) \quad (8)$$

From these expressions, one concludes that, when the master manipulator stops por move slowly, an accurate force feedback is produced, this characteristic is desirable for a move and wait application. In comparison, when the master manipulator is fast moving, the force feedback is inaccurate [7], [8]. This condition is caused by the influence of the incoming and outgoing wave with the master's velocity and the slave force.

C. *Architecture proposed by L. Bate.*

A modification to Bate's scheme was proposed by M. Zheng [8]. To reduce the effects and the influence between the incoming and the outgoing waves, two new blocks in the scheme are implemented. With the addition of the parameters α and β, the dependance between F_{md} and X_{md} becomes smaller, [8]. The velocity tracking and force reflecting equations are expressed in equations (9) and (10).

$$\dot{x}_{sd}(t) = \dot{x}_{md}(t-T) + \frac{\beta(F_s(t-T) - F_{sd}(t))}{b} \quad (9)$$

$$F_{md}(t) = \alpha b\dot{x}_m(t) + F_{sd}(t-T) \quad (10)$$

The adjusting factors α and β must be in a range of values between 0 and 1, the method to calculate these parameters is by trial-and-error approach. The proposed scheme is shown in Figure 4.

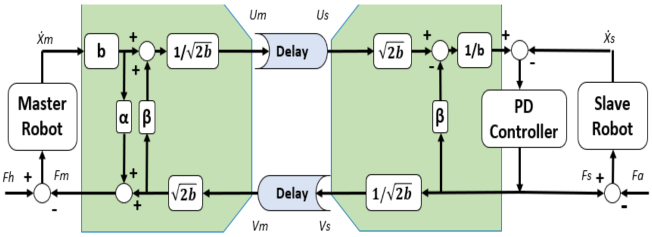


Fig. 4. Zheng's architecture.

D. *Low-pass filter implementation in the communication channel..*

As introduced in section I, G. Niemeyer also proposed the addition of a low-pass filter in the communication channel [6]. This, with the purpose to attenuate the wave reflections and high frequency signals that could pass through the communication channel. The representation of this implementation can be seen in Figure 5.

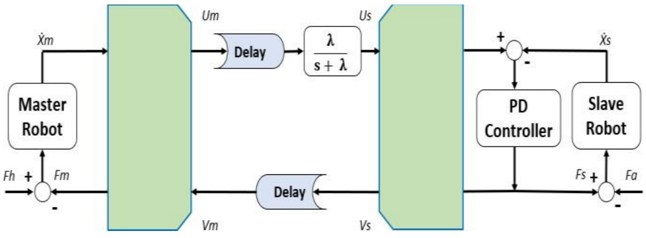


Fig. 5. Low-pass filter in the communication channel..

Where λ stands for the Bandwidth value of the teleoperation system. The addition of a low-pass filter can significantly reduce the wave reflections effects caused by the wave variable transformation, nevertheless, this signal attenuation could lead to a high frequency data lose due to the bandwidth restriction. In this work, the implementation of a low-pass filter in each architecture will be evaluated.

III. EXPERIMENTAL TELEOPERATOR WORKBENCH

For the implementation and evaluation of the architectures, a 1-DOF teleoperation testbed will be used. In [3], a detailed modeling process of the testbed is explained. Figure 6 depicts the physical experimental testbed, where a time delay is induced by software.

The experimental testbed under which the respective tests were carried out integrates a computer with Matlab® Simulink, 1 Sensoray® 626 data acquisition board, 2 ESCON 50/5® servo controllers, 1 PCB interface, 1 power supply, 1 test structure and 2 robots (master and slave), built from a BLDC maxon® motor and its respective joint and link. The teleoperation architecture to be evaluate is placed in a schematic in which the algorithm that controls the master robot and the slave robot are captured. Using Windows® Real Time Target and the Sensoray® 626 card it is possible to run the algorithm in real time at a sampling rate of 1khz. The PCB interface allows establishing synergy between the power supply, the servo controllers and the data acquisition board, for the respective exchange of sensor and control signals.

The transfer function for both motors, which represent the master and slave robots, and the PD controller are expressed in equations (11) and (12).

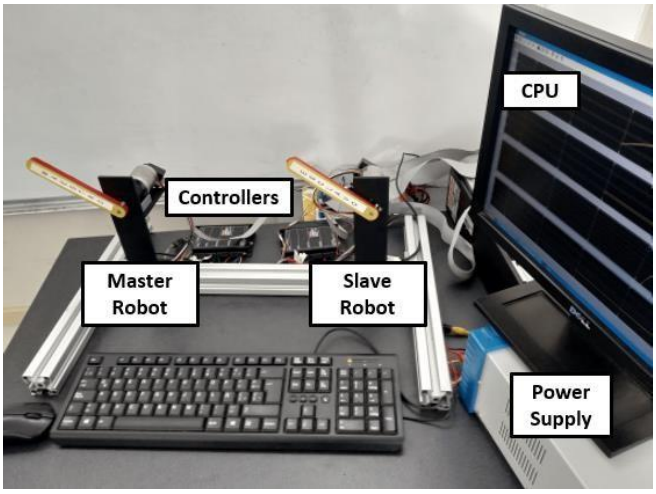


Fig. 6. Experimental 1 DOF bilateral teleoperation testbed.

$$G(s) = \frac{6015}{s+5.69} \quad (11)$$

$$C_{PD}(s) = \frac{0.0062s+0.025}{s} \quad (12)$$

Also, the transfer function of a low-pass filter for the system is expressed in equation (13).

$$G_{filter}(s) = \frac{5.67}{s+5.67} \quad (13)$$

The characteristic impedance that is used according to the system has set to a value of $b=0.001$, the adjusting factors for the architecture proposed by Zheng are chosen as $\alpha=0.7$ and $\beta=0.8$. The complete model of the teleoperation system is shown in Figure 7.

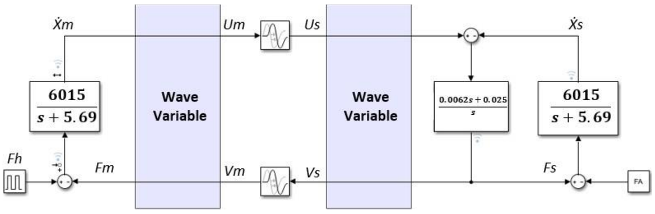


Fig. 7. Bilateral teleoperation model in Simulink.

IV. EXPERIMENTS

In this section, the different architectures are implemented and evaluated under the same conditions. The operator movement might correspond to a complex timevarying signal, nevertheless, as a first approach for a comparison, the schemes are tested with a delay time of 1 second and with a step reference signal. The experimental setup consists in a master robot excited by the operator (in this case simulated by the step reference signal), then the signals were delayed by software and sent to the slave robot which replicates the step command and finally send the feedback signals, again delayed, to the master robot. To have a more complete evaluation of the analyzed time delay compensation schemes, a low-pass filter is implemented in each architecture. For an accurate evaluation of the system, the following parameters are proposed:

- Steady-state motor velocity.
- Steady-state motor current.
- Stabilization time.
- Overshoot.

To define a fair index to measure the difference between the master and the slave responses, a performance factor (PF) is proposed. The quadratic error equation is used to calculate the area between the two curves, as expressed in equation (14).

$$PF = \int e^2(t)dt \qquad (14)$$

where the tracking error is defined as $e(t) = \dot{x}_m - \dot{x}_s$.

A. Experimental results without low-pass filter

In Figure 8, it can be observed that all architectures are capable to provide an accurate response in presence of time delays. In the case of the architecture proposed by Niemeyer, a high velocity value is obtained (240 rad/s) in comparison to the architectures proposed by Bate and Zheng, which have a velocity of 88 and 100 rad/s, respectively. Also, a steady-state is achieved in 9.5 seconds, while the stabilization time for Bate's and Zheng's architectures is 4.8 seconds. It is important to note that the wave reflections are significantly reduced in the architectures proposed by Bate and Zheng. Regarding the overshoot, the Bate architecture has a maximum value of 200 rad/s, this value is increased in Zheng's architecture, reaching a maximum value of 265 rad/s for the master robot.

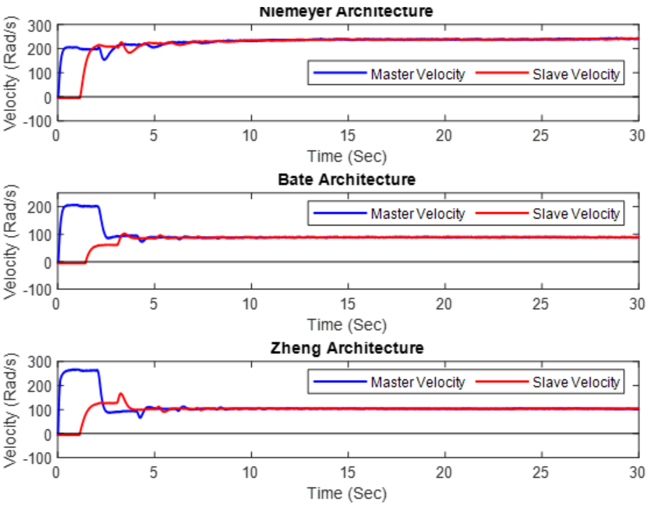


Fig. 8. Velocity response for a step signal for the three evaluated schemes.

Figure 9, presents the response in current and it better depicts the problem of the wave reflections in all the schemes. For the Niemeyer architecture, a steady-state value of 0.2 A and a very marked distortion can be observed for both master and slave robots. In the cases of Bate's and Zheng's architectures, a decrease of the steady-state motor current is observed to 0.16 A with a higher overshoot in the Zheng's controller response.

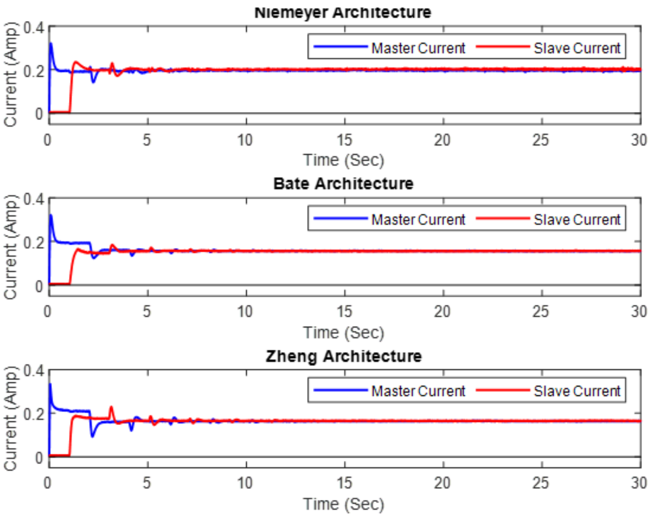


Fig. 9. Motor Current response for a step signal for the three evaluated schemes.

B. Experimental results with low-pass filter

With the implementation of a low-pass filter in the communication cannel, can be observed that the few wave reflections that appeared in the response without filter are attenuated. Out of that, the response is very similar with the one obtained in the past experiment. It is important to note that the stabilization time for the Niemeyer architecture increased to 10.7 seconds. These characteristics can be observed in Figure 10.

The response in current with a low-pass filter shows that the wave reflections are highly attenuated in all schemes, also, the distortion in Niemeyer's scheme is eliminated. The overshoot in the architectures is reduced in comparison with the scheme without filter; Figure 11 expresses this response.

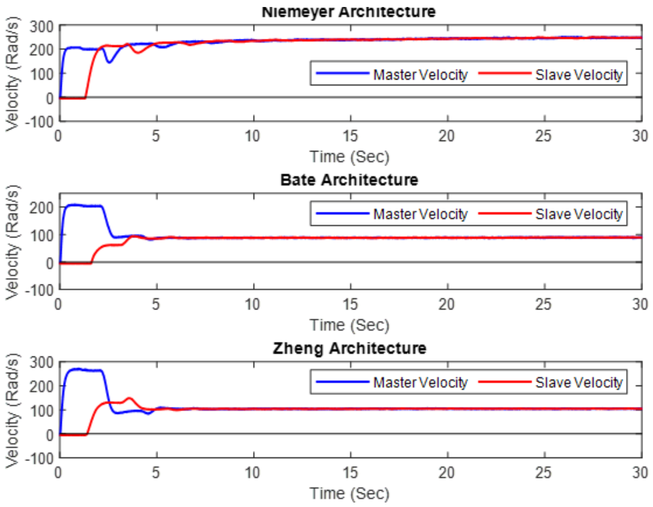


Fig. 10. Velocity response for a step signal with a low-pass filter.

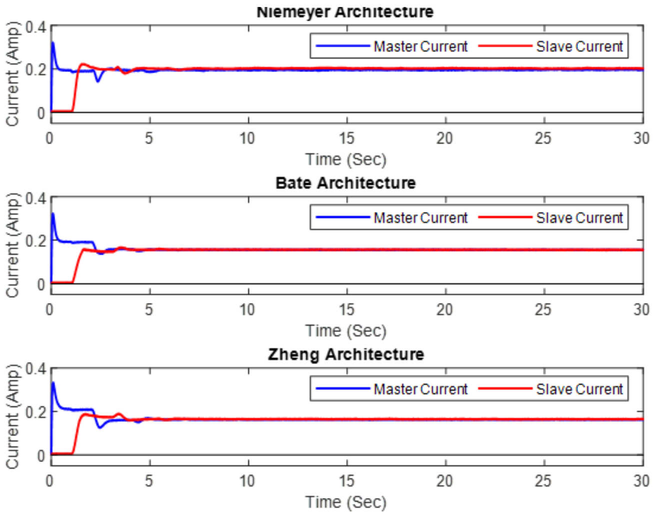


Fig. 11. Current response for a step signal and a low-pass filter in the communication channel.

V. ANALYSIS AND CONCLUSIONS

This work proved how different architectures based in the wave variable codification mitigate in different ways the effects of the time delay in the communication channel. Each scheme presented a diversity of special characteristics that are optimal for different applications where they can be used efficiently. Furthermore, differences between velocity, current, overshoot and stabilization time were identified with the purpose to obtain parameters that can facilitate the choice of the system to use. These differences might be due to the change in the equilibrium point of the closed loop system with different time delay compensation schemes. Individual conclusions for each scheme are discussed in this section.

The conducted experiments have demonstrated that all architectures successfully reduce the effects of a time delay in the communication channel. The values of the parameters proposed to evaluate the studied schemes in the experiments are presented in tables I and II.

TABLE I. VELOCITY PARAMETERS RESULTS

Scheme	**Robot**	**Velocity parameters**			
		Velocity (rad/s)	***Stabilization time (seconds)***	***Overshoot (rad/s)***	***PF***
Niemeyer	Master	240	9.5	-	50,695
	Slave	240	9.5	-	50,695
Niemeyer with filter	Master	245	10.7	-	60,327
	Slave	245	10.7	-	60,327
Bate	Master	88	3.8	206	76,530
	Slave	88	3.8	100	76,530
Bate with filter	Master	88	4.3	208	85,530
	Slave	88	4.3	92	85,530
Zheng	Master	101	4.3	266	107,280
	Slave	101	4.3	164	107,280
Zheng with filter	Master	101	5	270	120,202
	Slave	101	4.5	145	120,202

TABLE II. CURRENT PARAMETERS RESULTS

Scheme	**Robot**	**Velocity parameters**		
		Current (Amp)	***Overshoot (Amp)***	***Wave reflections***
Niemeyer	Master	0.2	0.34	Yes
	Slave	0.2	0.24	Yes
Niemeyer with filter	Master	0.19	0.32	Attenuated
	Slave	0.15	0.22	Attenuated
Bate	Master	0.15	0.32	Yes
	Slave	0.15	0.18	Yes
Bate with filter	Master	0.15	0.32	No
	Slave	0.15	0.16	No
Zheng	Master	0.15	0.34	Yes
	Slave	0.16	0.24	Yes
Zheng with filter	Master	0.16	0.33	No
	Slave	0.16	0.19	No

Niemeyer’s architecture: This scheme proves to have a good performance for a step reference signal. It has the advantage to provide a high velocity in both robots in comparison with the other two architectures. Nevertheless, wave reflections appear significantly in the system behavior, but they can be attenuated with the use of a low-pass filter. It has the weakness of having a long stabilization time and high overshoots in the current, also, if a low-pass filter is not implemented, distortions in the current could appear in the steady-state. Regarding the performance factor, is the architecture that provides a more accurate velocity tracking, because it has the smallest value in this parameter.

Bate's architecture: It has the advantage of being the architecture with the lower stabilization time. The master robot velocity presents a high overshoot before the slave robot begins the operation. In this scheme, the wave reflections are highly attenuated, achieving an optimal reduction with the use of a low-pass filter. One characteristic of this scheme is the reduction of the steady-state motor velocity in comparison with the Niemeyer's architecture.

Zheng's architecture: It demonstrates to have a small increase in the magnitude of the velocities in comparison to the results of the Bate's scheme. With the addition of a low - pass filter, an optimal performance is achieved, but with a small increase in the stabilization time. The overshoot that is present in the master's velocity increases the performance factor, which means that the velocity tracking is the most inaccurate of the three architectures.

With the results and conclusions explained in this paper, an architecture can be chosen depending on the final application of the system. In the opinion of the authors, if an accurate tracking of velocity and a high motor speed are required, the Niemeyer's architecture may be the best option for this purpose. In the case of a requirement of a faster stabilization in the response of the robots, the Bate's architecture is the optimal of the three schemes. Also, the implementation of a low-pass filter in any architecture is a good choice, if the tolerance to wave reflections is considered.

VI. REFERENCES

[1] Hokayem, P. F., & Spong, M. W. Bilateral teleoperation: An historicalsurvey. Automatica, 42(12), 2035–2057, 2006.

[2] Buzan, F. T., & Sheridan, T. B. A model-based predictive operator aidfor telemanipulators with time delay. Conference Proceedings., IEEE International Conference on Systems, Man and Cybernetics, 1989.

[3] C. Gómez-Rosas and R de J. Portillo-Veléz. Time delay compensationin a bilateral teleoperation system. Memorias del XXI Congreso Mexicano de Robótica, 2020.

[4] Da Sun, Fazel Naghdy, Haiping Du. (2014). Application of wave-variable control to bilateral teleoperation systems: A survey.

[5] Niemeyer G., Slotine J., (1991) Stable Adaptive Teleoperation. IEEE Journal of Oceanic Engineering, 16 (1), 152-162, 1991.

[6] Niemeyer G. Using Wave Variables In Time Delayed Force ReflectingTeleoperation, PhD. Thesis, MIT, 1996.

[7] Bate, L., Cook, D. Reducing Wave-Based Teleoperator Reflections for Unknown Environments, IEEE Transactions on Industrial Electronics 58(2), 392 - 397 2011.

[8] Zheng, M. Bilateral teleoperation with reducing wave-based reflections. Advances in manufacturing, 288-292. 2013.

Capítulo 5.

Aerial and educational robotics

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Active Fault Tolerant Control for a Quad-Rotor in case of a Rotor Failure

Romeo Falcón[†], Héctor Ríos[†,⋆] and Alejandro Dzul[†]

***Abstract*—In this paper an active fault tolerant control is developed for the stabilization problem of a Quad-Rotor system under the effects of a rotor failure. First, the healthy nonlinear model is presented. Then, the failure scenario is analyzed in order to identify how this affects the dynamics. It is assumed that the failure has been previously isolated. The fault tolerant control design is composed of PID and PD controllers allowing the yaw angular velocity to spin at a constant speed, around the vertical axis, and achieving a regulation task in the position (x, y, z). The closed-loop stability is formally proven. Numerical simulations show the performance of the proposed control strategy.**

***Index Terms*—Fault Tolerant Control, Quad-Rotor, Rotor Failure, Robust Control.**

I. INTRODUCTION

In the last decade the Unmanned Aerial Vehicles (UAVs) have attracted the attention of many research fields thanks to recent technological advances related to the size reduction and low cost of the used electronic devices. Among the UAVs, the Quad-Rotors remain as the most used vehicle due to flight flexibility to perform vertical take-off and landing tasks as well as the simplicity of operation.. Consequently, a wide variety of applications have appeared, *e.g.* security, agriculture, industrial supervision and military reconnaissance (see, for instance, [1] and [2]).

The growing demand for safety, reliability and acceptable level of performance in these tasks is a priority, and thus several interesting robust controllers have been designed (see, for instance, [3], [4] and [5]). However, such controllers are effective when the system works under rotors nominal conditions. In the presence of actuators faults, the control commands could not satisfy the performance level, operating in an unexpected scenario. Moreover, a total loss of one of their rotors can provoke a dangerous midair collision. Therefore, in order to increase safety and robustness, a fault tolerant control (FTC) must be designed in order to guarantee system stability and acceptable performance even in the presence of a rotor failure.

Regarding loss of efficiency in the rotors, in [6] the problem is tackled using robust properties in the design of the controllers based on sliding-mode control theory. An FTC that employs fault accommodation technique is proposed by [7] where the controller parameters are adapted to the parameters of the faulty plant. In [8] an adaptive estimation approach is exploited to identify the actuator fault upper bound and compensate it. Nevertheless, such strategies cannot be implemented when a rotor failure occurs.

In the context of rotor failures, in [9], after the complete loss of an actuator, a transformation of the Quad-Rotor system into a Tri-Rotor system is proposed. In this approach it is essential to redistribute the Quad-Rotor weight in order to shift its gravity center toward the rear rotor. On the other hand, in [10] and [11], a strategy to transform the Quad-Rotor system into a Bi-Rotor system is considered. Such an approach takes into account to turn off the propeller aligned on the same axis of the broken rotor, losing the pitch or roll angle, depending on the broken rotors, as well as the yaw angle, but allowing for an emergency landing. In [12], a link between the desired altitude and the yaw rotational speed is proposed. Once the yaw rate is stabilized, a cyclic reference is provided for the roll and pitch command angles in order to achieve the tracking objective. In [13], a multi-loop hybrid nonlinear controller is designed for the Quad-Rotor system with a single failure. Such a scheme is validated through experimental tests for high-speed tasks in the failure scenario, but lacks of a formal closed-loop stability proof. In a more recent study [14], an FTC is proposed for a Quad-Rotor considering a rotor failure and external disturbances. The proposed strategy is based on a nonsingular terminal sliding mode control, exploiting the robust properties provided by this controller but the yaw dynamics is totally ignored.

To sum up, to the best of our knowledge, the development of FTC against the total loss of a single rotor on Quad-Rotors is a current challenging topic and only few works have been reported in the literature. Motivated by the aforementioned difficulties, in this paper an FTC is proposed to deal with the effects of a rotor failure. The proposed strategy is composed of PID and PD controllers. Such a strategy allows the yaw angular velocity to spin at a constant speed around the vertical axis. In this sense, it is proved that the yaw angular velocity is related to the vehicle mass, moments of inertia, rotational drag and thrust coefficients. Moreover, the FTC allows to

[†] Tecnológico Nacional de México/I.T. La Laguna, División de Estudios de Posgrado e Investigación, C.P. 27000, Torreón Coahuila, México. Emails: `m.rfalconp@correo.itlalaguna.edu.mx` and `dzul@faraday.itlalaguna.edu.mx`, respectively.

[⋆] Cátedras CONACYT, C.P. 03940, Ciudad de México, México. Email: `hriosb@correo.itlalaguna.edu.mx`

achieve the regulation task in the three-dimensional plane without changing the gravity center and without shutting down healthy rotors. The closed-loop stability is formally proven. Numerical simulations validate the provided FTC strategy.

Notation: Denote $s(\cdot) := \sin(\cdot)$ and $c(\cdot) := \cos(\cdot)$; denote $\mathbb{R}_+ := \{x \in \mathbb{R} : x \geq 0\}$; for a Lebesgue measurable function $d : \mathbb{R}_+ \to \mathbb{R}^m$, define the norm $||d||_{[t_0,t_1)} = ess\sup_{t\in[t_0,t_1)}||d(t)||$, then $||d||_f = ||d||_{[t_f,+\infty)}$ and $||d||_\infty = ||d||_{[0,+\infty)}$; the set of $d(t)$ with the property $||d||_\infty < +\infty$ is denoted as $\mathcal{L}_\infty$. A sequence of integers $1, \ldots, n$ is denoted as $\overline{1,n}$.

II. PROBLEM STATEMENT

The simplified Quad-Rotor dynamics (see Fig. 1, and for modeling details see [15]), is given by

$$\dot{\xi}_1 = \xi_2, \tag{1a}$$
$$\dot{\xi}_2 = \frac{u}{m} g_\xi(\eta_1) - \Lambda_\xi \xi_2 - G, \tag{1b}$$
$$\dot{\eta}_1 = \eta_2, \tag{1c}$$
$$\dot{\eta}_2 = J^{-1}\tau + \Xi w_\eta(\eta_2) - \Lambda_\eta \eta_2, \tag{1d}$$

where $\xi_1 := (x, y, z)^T \in \mathbb{R}^3$, $\xi_2 := (\dot{x}, \dot{y}, \dot{z})^T \in \mathbb{R}^3$, $\eta_1 := (\phi, \theta, \psi) \in \mathbb{R}^3$ and $\eta_2 := (\dot{\phi}, \dot{\theta}, \dot{\psi})^T \in \mathbb{R}^3$. The variables x, $y \in \mathbb{R}$ are the coordinates in the horizontal plane, $z \in \mathbb{R}$ is the vertical position, while ϕ, θ and $\psi \in \mathbb{R}$ are the roll angle around the x-axis, the pitch angle around the y-axis, and the yaw angle around the z-axis, respectively. The term $u \in \mathbb{R}$ represents the main thrust and $m \in \mathbb{R}_+$ the mass of the Quad-Rotor. The matrices $\Lambda_\xi := \text{diag}(a_x, a_y, a_z) \in \mathbb{R}^{3\times3}$, $\Lambda_\eta := \text{diag}(a_\phi, a_\theta, a_\psi) \in \mathbb{R}^{3\times3}$, $\Xi := \text{diag}(b_\phi, b_\theta, b_\psi) \in \mathbb{R}^{3\times3}$ and $J := \text{diag}(J_x, J_y, J_z) \in \mathbb{R}^{3\times3}$ are given by the aerodynamic damping coefficients $a_x, a_y, a_z \in \mathbb{R}$, the rotational resistance moment coefficients $a_\phi, a_\theta, a_\psi \in \mathbb{R}$, and the inertial coefficients $b_\phi := (J_y - J_z)/J_x$, $b_\theta := (J_z - J_x)/J_y$ and $b_\psi := (J_x - J_y)/J_z$, where J_x, J_y and $J_z \in \mathbb{R}$ are the moments of inertia along x, y and z axis, respectively. The terms $G := (0, 0, g)^T \in \mathbb{R}^3$ and $\tau = (\tau_\phi, \tau_\theta, \tau_\psi)^T \in \mathbb{R}^3$ are the gravity vector and the angular moment vector, with $g \in \mathbb{R}$ as the gravitational acceleration, while τ_ϕ, τ_θ and $\tau_\psi \in \mathbb{R}$ represent the roll, pitch and yaw moments, respectively. The functions $g_\xi : \mathbb{R}^3 \to \mathbb{R}^3$ and $w_\eta : \mathbb{R}^3 \to \mathbb{R}^3$ are given as $g_\xi(\eta_1, \psi) := (\ c\phi s\theta c\psi + s\phi s\psi, c\phi s\theta s\psi - s\phi c\psi, c\phi c\theta\ )^T$ and $w_\eta(\eta_2) = (\dot{\theta}\dot{\psi}, \dot{\phi}\dot{\psi}, \dot{\phi}\dot{\theta})^T$.

The "+" configuration of the Quad-Rotor is considered in this work. Thus, the relation between the control inputs u, τ_ϕ, τ_θ, τ_ψ and the thrusts T_i, generated by the ith propeller, is given by

$$\begin{bmatrix} u \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 0 & L & -L \\ L & -L & 0 & 0 \\ K_\tau & K_\tau & -K_\tau & -K_\tau \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}, \tag{2}$$

where the constant L represents the distance between the motors and the center of mass of the aircraft while K_τ represents the thrust coefficient.

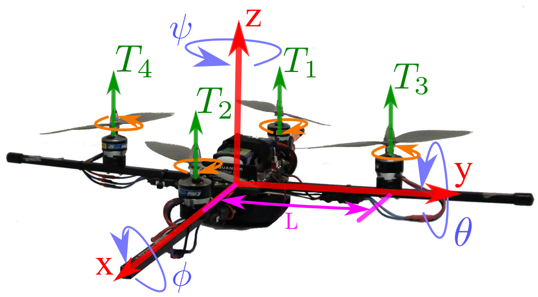


Figure 1. Schematic representation of the Quad-Rotor.

The actual thrust in the ith propeller is modeled as follows

$$\bar{T}_i(t) = \gamma(t)\, T_i(t), \tag{3}$$

where $\gamma(t) \in \mathbb{R}$ represents a switching failure variable given as

$$\gamma(t) = \begin{cases} 1, & \text{if } t < t_f, \\ 0, & \text{if } t \geq t_f, \end{cases}$$

where t_f is the time of failure occurrence. A propeller structural damage or an unexpected change in the rotor physical parameters would provoke a rotor failure.

The aim of this paper is to design an active FTC, taking into account the failure of a single rotor, in order to achieve a regulation task in the desired position (x, y, z). It is assumed that the failure has already been isolated. Such a fault diagnosis problem has been properly addressed in the literature (see, for instance, [16] and [17]).

Before proceeding with the failure scenario analysis, the following assumption is introduced.

Assumption 1. *The failure occurs once the Quad-Rotor has reached the hover task.*

III. FAILURE SCENARIO ANALYSIS

In the scenario of a rotor failure, relation (2) ceases to be bijective. In case of a failure on T_1, the relation between the control inputs and the remaining thrusts is given by

$$\begin{bmatrix} u \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & L & -L \\ -L & 0 & 0 \\ K_\tau & -K_\tau & -K_\tau \end{bmatrix} \begin{bmatrix} T_2 \\ T_3 \\ T_4 \end{bmatrix}. \tag{4}$$

From the previous relation, it is easy to see that the control input τ_ψ is now linearly dependent on u and τ_θ, *i.e.*

$$\tau_\psi(t) = K_\tau \left(-u(t) - 2\frac{\tau_\theta(t)}{L} \right), \forall\, t \geq t_{f1}, \tag{5}$$

where t_{f1} represents the failure time of the rotor 1. Similar expressions can be obtained by considering a failure on the rotors 2, 3 or 4, respectively, as

$$\tau_\psi(t) = K_\tau \left(-u(t) + 2\frac{\tau_\theta(t)}{L} \right), \ \forall\, t \geq t_{f2}, \tag{6a}$$

$$\tau_\psi(t) = K_\tau \left(u(t) + 2\frac{\tau_\phi(t)}{L} \right), \quad \forall\, t \geq t_{f3}, \tag{6b}$$

$$\tau_\psi(t) = K_\tau \left(u(t) - 2\frac{\tau_\phi(t)}{L} \right), \quad \forall\, t \geq t_{f4}, \tag{6c}$$

for the corresponding failure time. Therefore, the FTC design is carried out taking into account such a linear dependence caused by a rotor failure.

IV. ACTIVE FAULT TOLERANT CONTROL DESIGN

Define the tracking error vectors as

$$\begin{aligned} e_\xi &:= (e_x, e_y, e_z)^T = \xi_1 - \xi_d, \\ \varepsilon_\xi &:= (\varepsilon_x, \varepsilon_y, \varepsilon_z)^T = \xi_2, \\ e_\eta &:= (e_\phi, e_\theta)^T = T\eta_1 - \eta_d, \\ \varepsilon_\eta &:= (\varepsilon_\phi, \varepsilon_\theta)^T = T\eta_2, \end{aligned}$$

with

$$T := \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \end{array} \right],$$

and where $\xi_d := (x_d, y_d, z_d)^T \in \mathbb{R}^3$ and $\eta_d := (\phi_\star, \theta_\star)^T \in \mathbb{R}^2$ are the desired position and attitude vectors, respectively; with $\phi_\star$ and $\theta_\star$ being some reference signals to be designed.

Then, the tracking error dynamics is given as

$$\dot{e}_\xi = \varepsilon_\xi, \tag{7a}$$

$$\dot{\varepsilon}_\xi = \frac{u}{m} g_\xi(\eta_1) - \Lambda_\xi \xi_2 - G, \tag{7b}$$

$$\dot{e}_\eta = \varepsilon_\eta, \tag{7c}$$

$$\dot{\varepsilon}_\eta = T(J^{-1}\tau + \Xi w_\eta(\eta_2) - \Lambda_\eta \eta_2). \tag{7d}$$

Due to the under-actuated nature of the Quad-Rotor, one cannot control all the positions and angles independently. Hence, the references signals $\phi_\star$ and $\theta_\star$, as well as the control input u, must be properly designed in order to achieve a desired position ξ_d. To do this, a virtual control $\nu := (\nu_x, \nu_y, \nu_z)^T \in \mathbb{R}^3$ is introduced in the position error dynamics, *i.e.*

$$\dot{e}_\xi = \varepsilon_\xi, \tag{8a}$$

$$\dot{\varepsilon}_\xi = \nu + w_\xi(\eta_1, u, \nu) - \Lambda_\xi \xi_2, \tag{8b}$$

$$\dot{e}_\eta = \varepsilon_\eta, \tag{8c}$$

$$\dot{\varepsilon}_\eta = T(J^{-1}\tau + \Xi w_\eta(\eta_2) - \Lambda_\eta \eta_2), \tag{8d}$$

where the disturbance term $w_\xi(\eta_1, u, \nu) = (w_x, w_y, w_z) := \frac{u}{m} g_\xi(\eta_1) - G - \nu$. Thus, the virtual control ν may be chosen as follows

$$\nu_x = \frac{u}{m}(c\phi_\star s\theta_\star c\psi + s\phi_\star s\psi), \tag{9a}$$

$$\nu_y = \frac{u}{m}(c\phi_\star s\theta_\star s\psi - s\phi_\star c\psi), \tag{9b}$$

$$\nu_z = \frac{u}{m} c\phi_\star c\theta_\star - g. \tag{9c}$$

The main thrust u, as well as the references signals $\phi_\star$ and $\theta_\star$, are calculated from (9), *i.e.*

$$u = m\sqrt{\nu_x^2 + \nu_y^2 + (\nu_z + g)^2}, \tag{10a}$$

$$\phi_\star = \arcsin\left[u^{-1} \left(\nu_x s\psi - \nu_y c\psi \right) \right], \tag{10b}$$

$$\theta_\star = \arctan[(\nu_z + g)^{-1}(\nu_x c\psi + \nu_y s\psi)]. \tag{10c}$$

Therefore, the objective now is to design the virtual controller ν and the angular moments τ_ϕ and τ_θ such that the tracking error vector $e := (e_\xi, \varepsilon_\xi, e_\eta, \varepsilon_\eta) \in \mathbb{R}^{10}$ converges to zero despite the complete loss of a rotor. In order to fulfill such an objective, it will be demonstrated that system (8a) and (8b) is Input-to-State Stable (ISS) with respect to w_ξ, it will be shown that system (8c) and (8d) is Uniformly Exponentially Stable (UES), then the behavior of the term w_ξ will be discussed, and finally, the yaw dynamics will be analyzed.

A. Position Control Design

Let us propose the following PID virtual controller ν as

$$\nu = K_{i\xi}\bar{e}_\xi + K_{p\xi}e_\xi + K_{d\xi}\varepsilon_\xi - \Lambda_\xi \xi_2, \tag{11}$$

where $K_{i\xi} := \operatorname{diag}(k_{x1}, k_{y1}, k_{z1}) \in \mathbb{R}^{3\times3}$, $K_{p\xi} := \operatorname{diag}(k_{x2}, k_{y2}, k_{z2}) \in \mathbb{R}^{3\times3}$, $K_{d\xi} := \operatorname{diag}(k_{x3}, k_{y3}, k_{z3}) \in \mathbb{R}^{3\times3}$ and $\bar{e}_\xi := (\bar{e}_x, \bar{e}_y, \bar{e}_z)^T \in \mathbb{R}^3$, with $\bar{e}_x := \int_0^t e_x(\tau)d\tau$, $\bar{e}_y := \int_0^t e_y(\tau)d\tau$ and $\bar{e}_z := \int_0^t e_z(\tau)d\tau$.

Then, the following result can be established.

Theorem 1. *[3] Let the control* (11) *be applied to system* (1). *Then, system* (8a) *and* (8b) *is ISS with respect to ω_ξ.*

The proof of Theorem 1 is given in [3].

B. Pitch and Roll Control Design

The position control loop can be seen as an external control loop while the attitude tracking control of the pitch and roll angles as an internal control loop. The attitude tracking error dynamics (8c) can be viewed as a block decoupled dynamics with two independent control inputs.

Let us propose the following pitch and roll controllers

$$\tau_\phi = J_x \bar{\tau}_\phi - b_\phi \dot{\theta}\dot{\psi} + a_\phi \dot{\phi}, \tag{12a}$$

$$\tau_\theta = J_y \bar{\tau}_\theta - b_\theta \dot{\phi}\dot{\psi} + a_\theta \dot{\theta}. \tag{12b}$$

It is worth mentioning that if Assumption 1 holds, then before the failure occurs and in hover conditions one obtain that $\phi^\star = \theta^\star = \dot{\phi} = \dot{\theta} = 0$.

Since the thrust T_i in the ith rotor is proportional to the square of the rotation speed Ω_i, *i.e.* $T_i = k_i\Omega_i^2$, with k_i as a positive constant, then $T_i \geq 0$. Therefore, after the complete loss of a single rotor, the angular moments $\tau_\theta = T_1 - T_2$ and $\tau_\phi = T_3 - T_4$ are severely affected.

In the occurrence of a failure in the first rotor, it follows that $\tau_\theta(t) = -T_2 \leq 0$, for all $t \geq t_{f1}$. Consequently, such a control needs to be negative. A failure in the second rotor requires a positive control since $\tau_\theta(t) = T_1 \geq 0$, for all

$t \geq t_{f2}$. In the same way, a failure in the third rotor requires a negative control since $\tau_\phi(t) = -T_4 \leq 0$, for all $t \geq t_{f3}$, while a failure in the fourth rotor requires a positive control since $\tau_\phi(t) = T_3 \geq 0$, for all $t \geq t_{f4}$. Therefore, such sign constraints must be taken into account for the attitude control design.

Let us design $\bar{\tau}_\phi$ and $\bar{\tau}_\theta$ as PD controllers, *i.e.*

$$\bar{\tau}_\phi = k_{\phi 1} e_\phi + k_{\phi 2} \varepsilon_\phi, \tag{13a}$$

$$\bar{\tau}_\theta = k_{\theta 1} e_\theta + k_{\theta 2} \varepsilon_\theta, \tag{13b}$$

where the gains of the previous controllers are designed by pole-placement to achieve a critically damped or overdamped behavior in order to avoid a change of sign on the control laws (12).

Then, the following result can be established.

Theorem 2. *Let the control* (12)-(13) *be applied to system* (1). *Then,* $(e_\eta, \varepsilon_\eta) = 0$ *is UES.*

Proof. Let us consider that the first actuator has a failure. Then, τ_θ must satisfy

$$\tau_\theta = J_y(k_{\theta 1} e_\theta + k_{\theta 2} \varepsilon_\theta) - b_\theta \dot{\phi}\dot{\psi} + a_\theta \dot{\theta} \leq 0.$$

Note that after the failure occurrence on T_1, a clock-wise moment is provoked by the imbalance between the rotors 1 and 2 (please, see Fig. 1); and hence, a negative angle θ and a negative angular velocity $\dot{\theta}$ are induced, *i.e.* $\theta(t_{f1}) < 0$ and $\dot{\theta}(t_{f1}) < 0$. Then, since in hover flight $\theta^\star = \dot{\phi} = 0$, in order to fulfill the constraint $\tau_\theta \leq 0$, one just needs to properly design $\bar{\tau}_\theta \leq 0$, *i.e.* the gains $k_{\theta 1}$ and $k_{\theta 2}$ such that a negative critically damped or overdamped behavior is ensured.

Thus, the closed-loop dynamics for θ considering (12)-(13), can be written as

$$\dot{\Theta} = (A + BK_\theta)\Theta, \tag{14}$$

$$A = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, B = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

where $\Theta := (e_\theta, \varepsilon_\theta)^T \in \mathbb{R}^2$ and $K_\theta = (k_{\theta 1}, k_{\theta 2}) \in \mathbb{R}^2$. Due to the fact that the pair (A, B) is controllable, there always exists a matrix K_θ such that $A + BK_\theta$ is Hurwitz. To design the matrix K_θ, a pole-placement technique can be used in order to have a critical damped or overdamped behavior. Then, one has that $(e_\theta, \varepsilon_\theta) = 0$ is UES.

The same procedure is followed to ensure that $(e_\phi, \varepsilon_\phi) = 0$ is UES when the control input (12a)-(13a) is applied. Therefore, it is concluded that $(e_\eta, \varepsilon_\eta) = 0$ is UES.

A similar procedure can be followed for guaranteeing the exponential stability of $(e_\eta, \varepsilon_\eta) = 0$ when some other rotor has a failure. This concludes the proof. ☐

C. Virtual Control Disturbance Term Behavior

According to Theorem 1, it is given that the dynamics for e_ξ and ε_ξ are ISS with respect to w_ξ. Now, let us recall that such a disturbance term can be written as

$$w_\xi(\eta_1, u, \nu) = \frac{u}{m}\left[\begin{pmatrix} c\phi s\theta c\psi + s\phi s\psi \\ c\phi s\theta s\psi - s\phi c\psi \\ c\phi c\theta \end{pmatrix} - \begin{pmatrix} c\phi_\star s\theta_\star c\psi + s\phi_\star s\psi \\ c\phi_\star s\theta_\star s\psi - s\phi_\star c\psi \\ c\phi_\star c\theta_\star \end{pmatrix}\right].$$

Such a function is Lipschitz in η_1 and continuous in u, then it follows that

$$\|w_\xi\|_\infty \leq L \|e_\eta\|_\infty,$$

for all $\eta_1, \nu \in \mathbb{R}^3$ and $u \in \mathbb{R}$, for some positive $L > 0$. Then, as a result of Theorem 2, the disturbance term w_ξ vanishes when $e_\eta \to 0$, *i.e.* $\lim_{t\to\infty} w_\xi(t) = 0$. Moreover, the previous statement implies that $(e_\xi, \varepsilon_\xi) = 0$ will be UES.

D. Yaw Dynamics

The yaw dynamics, which is not possible to control due to the loss of one actuator and its linear dependence on u and τ_ϕ, or τ_θ according to the damaged rotor; is given by

$$\ddot{\psi} = \frac{\tau_\psi}{J_z} + b_\psi \dot{\phi}\dot{\theta} - a_\psi \dot{\psi}. \tag{15}$$

Considering the loss of the first actuator, (15) can be rewritten as

$$\ddot{\psi} = -\frac{K_\tau}{J_z}\left(u + 2\frac{\tau_\theta}{L}\right) + b_\psi \dot{\phi}\dot{\theta} - a_\psi \dot{\psi}, \tag{16}$$

and by substituting u, given in (10a), one obtains

$$\ddot{\psi} = -\frac{K_\tau}{J_z}\left(m\sqrt{\nu_x^2 + \nu_y^2 + (\nu_z + g)^2} + 2\frac{\tau_\theta}{L}\right) + b_\psi \dot{\phi}\dot{\theta} - a_\psi \dot{\psi}, \tag{17}$$

then, based on the statements given by Theorem 1 and 2, it follows that the tracking error vector $e = 0$ is UES. Thus, at steady state, one has that $\nu_x = \nu_y = \nu_z = \tau_\theta = 0$ as well as $\dot{\theta} = \dot{\phi} = 0$; and hence, (17) is written as

$$\ddot{\psi} = -\frac{K_\tau m g}{J_z} - a_\psi \dot{\psi}.$$

The solution of the previous differential equation is given by

$$\dot{\psi}(t) = \dot{\psi}(t_f) e^{-a_\psi(t - t_f)} - \frac{K_\tau m g}{J_z a_\psi}\left(1 - e^{-a_\psi(t - t_f)}\right),$$

for all $t \geq t_f$. Therefore, it is clear that

$$\lim_{t\to\infty} \dot{\psi}(t) = -\frac{K_\tau m g}{J_z a_\psi}. \tag{18}$$

Thus, after the occurrence of a single rotor failure, and applying the the proposed active FTC, the yaw angular velocity spin at a constant speed around the vertical axis and it achieves the regulation task in the desired position ξ_d.

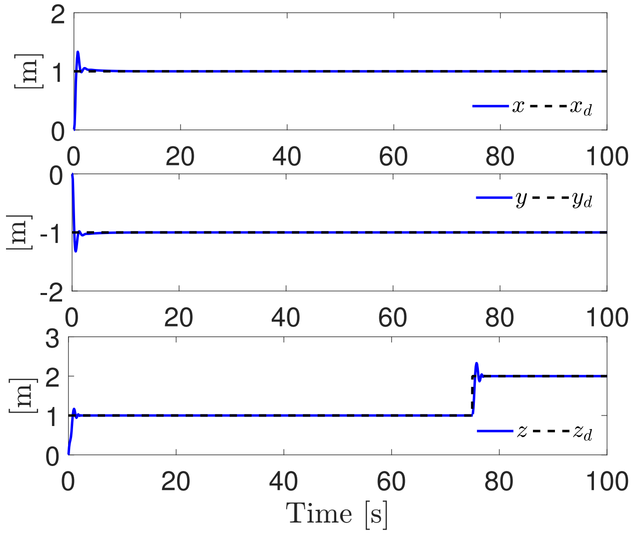


Figure 2. Trajectories of the Quad-Rotor Position.

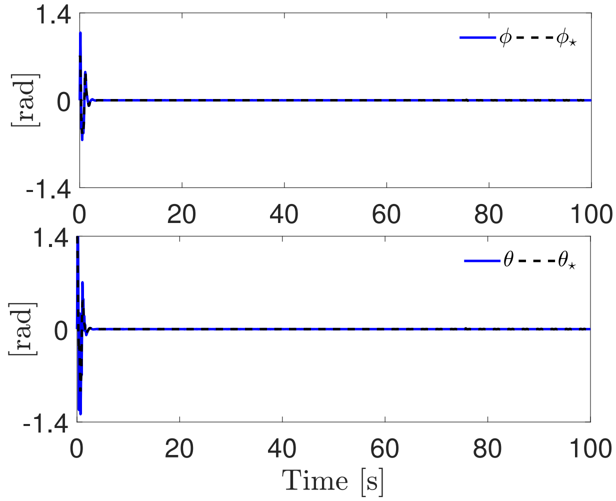


Figure 3. Trajectories of the Quad-Rotor Roll-Pitch Orientation.

V. NUMERICAL SIMULATIONS

In order to show the performance and effectiveness of the proposed FTC strategy, numerical simulations using MATLAB/SIMULINK® is presented. For this purpose, the Quad-Rotor model parameters are taken of Quanser's Qball2: $m = 1.79$ [kg] , $g = 9.81$ [m/s^2], $J_x = J_y = 0.03$ [Ns^2/rad], $J_z = 0.04$ [Ns^2/rad], $a_x = a_y = a_z = 0.1$ [Ns/m], $a_\phi = a_\theta = a_\psi = 0.3$ [Ns/rad], $L = 0.2$ [m] and $K_\tau = 0.0057$ [Nm/rad^2]. The failure of the first rotor is considered at $t_f = 50$[s].

The desired regulation point is given as $x_d = 1$, $y_d = -1$ and $z_d = 1$ for $t < 75$, while $z_d = 2$ for $t \geq 75$. The initial conditions are $\xi_1(0) = \xi_2(0) = \eta_1(0) = \eta_2(0) = 0$. The simulations have been implemented with the Euler's integration method with a sampling method equal to 0.001[s].

The regulation task, in the three-dimensional plane, is illustrated in Fig. 2 where, despite the failure of a single actuator, the Quad-Rotor remains the desired hover point. Furthermore, after the failure, there is a change in the altitude reference signal showing that the regulation task is insensitive to the failure in this coordinate, and then allowing landing actions. The roll and pitch signals, as well as their reference signals given by the expressions (10b) and (10c), are depicted in Fig. 3.

The yaw behavior, whose rotational moment τ_ψ is linearly dependent to u and τ_θ after the occurrence of the failure in the first rotor, is shown in Figure 4. The yaw angle ψ, after the occurrence of the failure, begins to spin in the clockwise direction around the z-axis. The angular velocity $\dot{\psi}$, governed by (18) is given by

$$\lim_{t\to\infty} \dot{\psi}(t) = -\frac{K_\tau m g}{J_z a_\psi} = -8.34,$$

which is bounded and reaches that value despite the reference change in the z coordinate. Yaw acceleration $\ddot{\psi}$ shows, after failure occurs, that it is abruptly accelerated by the difference in angular moment $\tau_\psi = K_\tau(T_2 - T_3 - T_4)$ and finally slows down by the rotational resistance term. Due to the existing linear dependence between u and τ_ψ, at the moment in which the reference signal z_d change, the dynamic behavior of ψ is also affected. This effect is seen more clearly in the second 75.

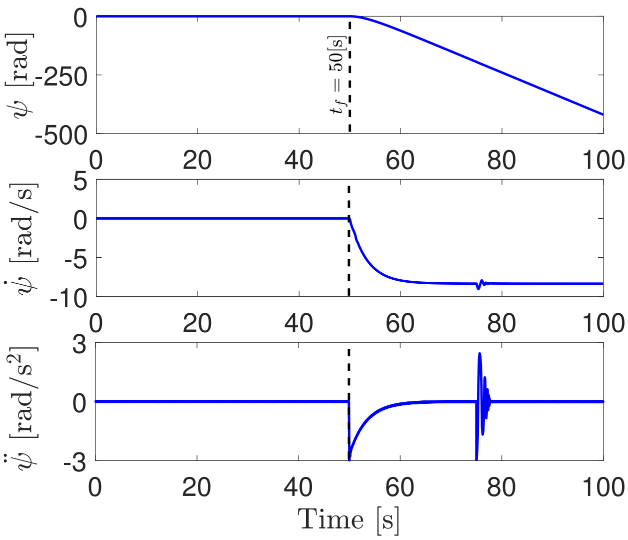


Figure 4. Trajectories of the Quad-Rotor Yaw.

Finally, the control signals, as well as the thrusts in each rotor, are shown in Figures 5 and 6, respectively. Again, after the occurrence of the failure, the yaw angular momentum takes the value given in (5). In order to maintain balance in the pitch motion, when the failure occurs, the FTC makes the thrust of rotor 2 near to zero, while the thrust on rotors 3 and 4 are increased to compensate the missing thrusts of the rotors 1 and 2. At the moment in which the reference

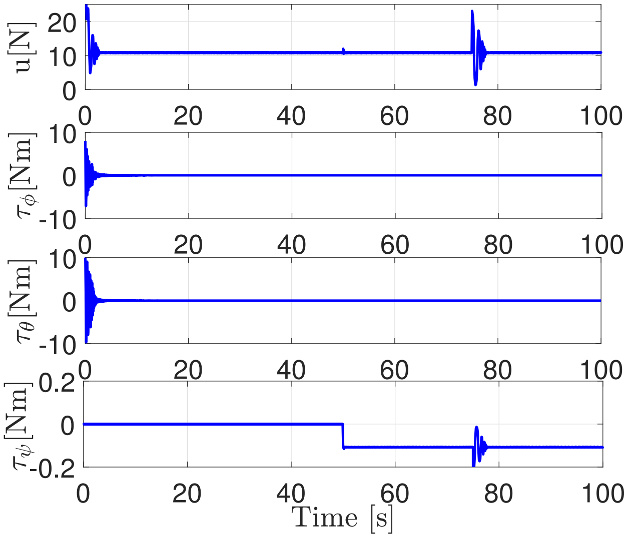


Figure 5. Quad-Rotor Control Signals.

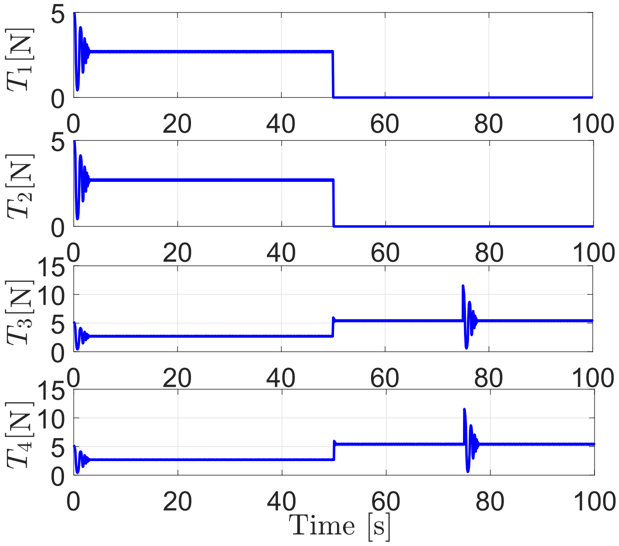


Figure 6. Rotor Thrusts.

signal z_d change, the control inputs and rotor thrusts behave abruptly in order to achieve the regulation task.

VI. CONCLUSIONS

This work contributes with the design of an active FTC for a Quad-Rotor under the effects of a rotor failure. The proposed strategy is composed of PID and PD controllers. Such a strategy allows the vehicle to spin at a constant speed around the vertical axis and achieving a regulation task on a desired position (x, y, z). The linear dependence in the control signals after the failure is analyzed. The closed-loop stability is formally proven. Simulations results validate the proposed strategy. The disturbance case and the tracking problem under the effects of a rotor failure are in the scope of future work.

ACKNOWLEDGMENT

This work was supported in part by the SEP-CONACYT-ECOS-ANUIES Project 315597. The authors gratefully acknowledge the financial support from TecNM projects, Catedras CONACYT CVU 270504 project 922 and CONACYT CVU 785635.

REFERENCES

[1] I. Palunko, P. Cruz, and R. Fierro, “Agile load transportation,” *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 69–79, 2012.

[2] T. Özaslan, S. Shen, Y. Mulgaonkar, N. Michael, and V. Kumar, “Inspection of penstocks and featureless tunnel-like environments using micro UAVs,” in *Field and Service Robotics*, ser. Springer Tracts in Advanced Robotics, L. Mejias, P. Corke, and J. Roberts, Eds. Switzerland: Springer International Publishing, 2015, vol. 105, pp. 123–136.

[3] H. Ríos, R. Falcón, O. González, and A. Dzul, “Continuous sliding-modes control strategies for quad-rotor robust tracking: Real-time application,” *IEEE Transactions on Industrial Electronics*, vol. 66, pp. 1264 – 1272, 2018.

[4] H. Wang, X. Ye, Y. Tian, G. Zheng, and N. Christov, “Model-free-based terminal SMC of quadrotor attitude and position,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 52, no. 5, pp. 2519–2528, 2016.

[5] B. Zhao, B. Xian, Y. Zhang, and X. Zhang, “Nonlinear robust adaptive tracking control of a quadrotor uav via immersion and invariance methodology,” *IEEE Transactions on Industrial Electronics*, vol. 62, no. 5, pp. 2891–2902, 2015.

[6] A.-R. Merheb, H. Noura, and F. Bateman, “Design of passive fault-tolerant controllers of a quadrotor based on sliding mode theory,” *International Journal of Applied Mathematics and Computer Science*, vol. 25, no. 3, pp. 561–576, 2015.

[7] R. C. Avram, X. Zhang, and J. Muse, “Quadrotor actuator fault diagnosis and accommodation using nonlinear adaptive estimators,” *IEEE Transactions on Control Systems Technology*, vol. 25, no. 6, pp. 2219–2226, 2017.

[8] B. Xiao, Q. Hu, and Y. Zhang, “Adaptive sliding mode fault tolerant attitude tracking control for flexible spacecraft under actuator saturation,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 6, pp. 1605–1612, 2012.

[9] A.-R. Merheb, H. Noura, and F. Bateman, “Emergency control of ar drone quadrotor uav suffering a total loss of one rotor,” *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 2, pp. 961–971, 2017.

[10] V. Lippiello, F. Ruggiero, and D. Serra, “Emergency landing for a quadrotor in case of a propeller failure: A PID based approach,” in *12th IEEE International Symposium on Safety, Security and Rescue Robotics*, 2014, pp. 1–7.

[11] V. Lippiello, F. Ruggiero, and D. Serra, “Emergency landing for a quadrotor in case of a propeller failure: A backstepping approach,” in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2014, pp. 4782–4788.

[12] A. Lanzon, A. Freddi, and S. Longhi, “Flight control of a quadrotor vehicle subsequent to a rotor failure,” *Journal of Guidance Control and Dynamics*, vol. 37, pp. 580–591, 2014.

[13] S. Sun, L. Sijbers, X. Wang, and C. De Visser, “High-speed flight of quadrotor despite loss of single rotor,” *IEEE Robotics and Automation Letters*, vol. 3, pp. 3201–3207, 2018.

[14] Z. Hou, P. Lu, and Z. Tu, “Nonsingular terminal sliding mode control for a quadrotor uav with a total rotor failure,” *Aerospace Science and Technology*, vol. 98 105716, pp. 1–18, 2020.

[15] L. R. García-Carrillo, A. Dzul-López, R. Lozano, and C. Pégard, *Quad Rotorcraft Control*, ser. Advances in Indutrial Control. London: Springer-Verlag London Heidelberg New York Dordrecht, 2013.

[16] M. H. Amoozgar, A. Chamseddine, and Y. Zhang, “Experimental test of a two-stage Kalman filter for actuator fault detection and diagnosis of an unmanned quadrotor helicopter,” *Journal of Intelligent & Robotic Systems*, vol. 70, no. 1-4, pp. 107–117, 2013.

[17] Z. Cen, H. Noura, T. B. Susilo, and Y. Al Younes, “Robust fault diagnosis for Quadrotor UAVs using adaptive Thau observer,” *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1-4, pp. 573–588, 2014.

Fault Diagnosis and Estimation for a Quadrotor UAV

1st J.J. Avilés-Espinoza
Facultad de Ingeniería Eléctrica
UMSNH
Morelia, México
juan.espinoza@umich.mx

2nd J. Anzurez-Marin
Div. de Estudios de Posgrado
Fac. de Ing. Eléctrica-UMSNH
Morelia, México
juan.anzurez@umich.mx

3rd F.R. López-Estrada
Turix-Dynamics Diagnosis and Control Group
Tecnológico Nacional de México
Tuxtla Gutiérrez, Mexico
frlopez@ittg.edu.mx

***Abstract*—This paper presents a Fault Detection and Estimation (FDE) scheme for a quadrotor UAV gyroscopic sensor, using a Takagi-Sugeno (TS) model for the rotational dynamics of the vehicle, with an Unknown Input Observer (UIO). The proposed scheme is solved with usage of linear matrix inequalities (LMIs) theory, and validated with nonlinear simulations.**

I. INTRODUCTION

The need to increase safety in systems has made Fault Diagnosis and Isolation (FDI) a major component in today's control theory. Therefore being able to distinguish whether a system is working in nominal conditions or a faulty condition will make the difference between a successful control design and implementation, with a lower probability of an accident occurring, or the opposite, see [1], [2], [3], [4].

In particular, the design of FDI schemes to a quadrotor Unmanned Aerial Vehicle (UAV) is not just an interesting problem in the scientific community due to system complexity, but also for the society itself. Quadrotors, popularly known as drones, have reached an amazing popularity due to their versatility and immense applications such as photography, filmmaking, infrastructure inspection, delivery, communications, agriculture and recreational uses, achieving a peak in sales according to Juniper's [5], [6]. Assuring their correct performance is then a must for drone manufacturing companies.

Due to reasons aforementioned many different approaches for FDI in a quadrotor, both for actuators and sensors, have been reported in the literature, most of them combined with a fault tolerant control (FTC) scheme, e.g. nonlinear observers, see [7],[8], Kalman filtering, see [9], [10], [11], sliding mode observers as in [12], [13], [14], Takagi Sugeno (TS) convex observers, and Linear Parameter-Varying (LPV) observers, see [15], [16], [17].

In this paper, a fault diagnosis, and estimation (FDE) scheme for a quadrotor under gyroscopic sensor additive faults is designed. Firstly, the quadrotor model is presented, from which a TS model is obtained considering rotational dynamics only. From this, a descriptor representation scheme is extended, and thus a TS convex Unknown Input Observer (UIO) is derived for the descriptor model. Fault diagnosis is achieved by residual generation, and estimation is achieved due to the descriptor scheme observer.

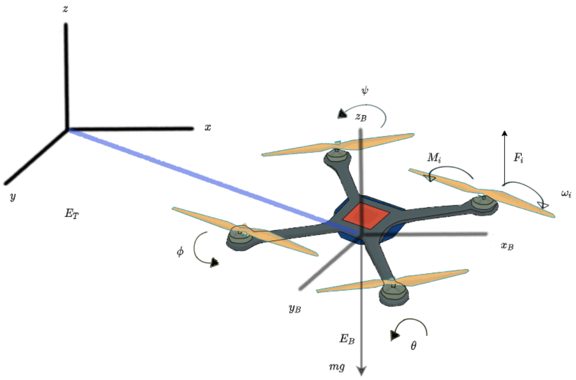


Fig. 1. Quadrotor UAV and reference frames.

II. QUADROTOR MODEL

In this section the quadrotor model is presented. The model considered is the one developed by [18] via the application of Newton-Euler formulation, giving the state-space representation (1), considering the state vector as:

$$X = [x_p, y_p, z_p, v_x, v_y, v_z, \phi, \theta, \psi, \rho, q, r_z]$$

where x, y, z are the positions, and v_x, v_y, v_z their correspondent velocities referred to the fixed inertial frame E_T, while the Euler angles are given as ϕ, θ, ψ, with their correspondent angular velocities ρ, q, r_z referred to the body frame E_B as shown in figure 1.

$$
\begin{aligned}
\dot{x_p} &= v_x \\
\dot{y_p} &= v_y \\
\dot{z_p} &= v_z \\
\dot{v_x} &= -(\cos\psi\sin\theta\cos\phi + \sin\psi\sin\phi)\frac{u_1}{m} \\
\dot{v_y} &= -(\sin\psi\sin\theta\cos\phi + \cos\psi\sin\phi)\frac{u_1}{m} \\
\dot{v_z} &= g - (\cos\theta\cos\phi)\frac{u_1}{m} \\
\dot{\phi} &= \rho + r_z(\tan\theta\cos\phi) + q(\tan\theta\sin\phi) \\
\dot{\theta} &= q\cos\phi - r_z\sin\phi \\
\dot{\psi} &= r_z(\sec\theta\cos\phi) + q(\sec\theta\sin\phi) \\
\dot{\rho} &= (\frac{I_y - I_z}{I_x})qr_z + \frac{u_2}{I_x} \\
\dot{q} &= (\frac{I_z - I_x}{I_y})pr_z + \frac{u_3}{I_y} \\
\dot{r_z} &= (\frac{I_x - I_y}{I_z})pq + \frac{u_4}{I_z}
\end{aligned}
\tag{1}
$$

The inputs u represent the forces and torques generated by the motors' speed, which are controlled with a PWM signal u_p as in [11], and are related by (2).

$$
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ l & -l & 0 & 0 \\ 0 & 0 & l & -l \\ K_\psi & K_\psi & -K_\psi & -K_\psi \end{bmatrix} \begin{bmatrix} K_m u_{p1} \\ K_m u_{p2} \\ K_m u_{p3} \\ K_m u_{p4} \end{bmatrix} \tag{2}
$$

III. TAKAGI-SUGENO CONVEX MODEL

In order to develop a FDI scheme for gyroscopic sensors, the rotational dynamics are considered decoupled from the inputs u_{pi}, and independent from the translational dynamics, hence the new state vector for the rotational dynamics is given as $x_{ts} = [\phi, \theta, \psi, \rho, q, r_z]^T$. To develop a TS system for the rotational dynamics, the quadrotor is considered near hover condition, simplifying its design; via state factorization from the rotational dynamics and by defining the scheduling variables as $z_1 = q, z_2 = \rho$, subsystem (3) is obtained.

$$
\begin{aligned}
\dot{x_{ts}}(t) = &\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_1 z_1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_2 z_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_3 z_2 & 0 \end{bmatrix} x_{ts}(t) \\
&+ \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{1}{I_x} & 0 & 0 \\ 0 & 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & 0 & \frac{1}{I_z} \end{bmatrix} u(t)
\end{aligned}
\tag{3}
$$

where:

$$
\begin{aligned}
a_1 &= (\frac{I_y - I_z}{I_z}) \\
a_2 &= (\frac{I_z - I_x}{I_z}) \\
a_3 &= (\frac{I_x - I_y}{I_z})
\end{aligned}
$$

In order to design a TS model with the nonlinear sector approach, the scheduling variables determine the number of models to use, which for the case of $p = 2$ variables, a total of $m = 2^p = 4$ models are obtained. These variables are bounded as $z_1, z_2 \in [-22 \ 22]$ rad/s. Then the scheduling functions are given by (4).

$$
\begin{aligned}
FP_{z1_{min}} &= \frac{22 - z_1}{22 - 0} = \frac{1}{2} - \frac{z_1}{44} \\
FP_{z1_{max}} &= 1 - FP_{z1_{min}} = \frac{1}{2} + \frac{z_1}{44} \\
FP_{z2_{min}} &= \frac{22 - z_2}{22 - 0} = \frac{1}{2} - \frac{z_2}{44} \\
FP_{z2_{max}} &= 1 - FP_{z2_{min}} = \frac{1}{2} + \frac{z_2}{44}
\end{aligned}
\tag{4}
$$

Using (4) the following weighting functions for the TS model can be obtained:

$$
\begin{aligned}
h_1 &= FP_{z1_{min}} * FP_{z2_{min}} \\
h_2 &= FP_{z1_{min}} * FP_{z2_{max}} \\
h_3 &= FP_{z1_{max}} * FP_{z2_{min}} \\
h_4 &= FP_{z1_{max}} * FP_{z2_{max}}
\end{aligned}
\tag{5}
$$

The system parameters considered are given in table I

TABLE I
SYSTEM PARAMETERS

Parameter	Value	Unit
K_m	120	N
K_ψ	4	Nm
l	0.2	m
I_x	0.03	Kgm^2
I_y	0.03	Kgm^2
I_z	0.04	Kgm^2
m	1.42	Kg
g	9.81	m/s^2

The state, input and output matrices A_i, B, C are given by:

$$
\begin{aligned}
A_1 &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 5.5 \\ 0 & 0 & 0 & 0 & 0 & -5.5 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
A_2 &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 5.5 \\ 0 & 0 & 0 & 0 & 0 & 5.5 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
A_3 &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -5.5 \\ 0 & 0 & 0 & 0 & 0 & -5.5 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
A_4 &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -5.5 \\ 0 & 0 & 0 & 0 & 0 & 5.5 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
B &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 800 & -800 & 0 & 0 \\ 0 & 0 & 800 & -800 \\ 12000 & 12000 & -12000 & -12000 \end{bmatrix} \\
C &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}
\end{aligned}
\tag{6}
$$

All the subsystems satisfy the observability condition. In order to regulate the quadrotor, a linear state feedback is designed, considering a linear model of (1) with actuator dynamics, obtained by a Taylor-Series expansion. The derivation is not presented since it is not the focus of this work.

IV. FAULT DIAGNOSIS AND ESTIMATION SCHEME

For FDE in gyroscopic sensors, considering the TS model obtained in previous section, a UIO convex observer is designed, extending the state vector as a descriptor system, [19]. The block diagram in Fig 2 describes the proposed scheme in general form.

A. Descriptor Scheme Representation

In order to design a UIO for sensor fault estimation, we consider the TS faulty model:

$$
\begin{aligned}
\dot{\tilde{x}} &= \sum_{i=1}^{4} (h_i \tilde{A}_i \tilde{x}) + \tilde{B} u_p \\
y &= \tilde{C} \tilde{x} + f
\end{aligned}
\tag{7}
$$

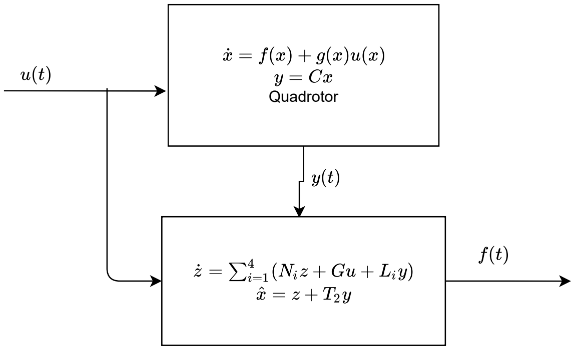


Fig. 2. FDE proposed scheme.

where f represents an additive fault vector. The model (7) can be written as (8)

$$
\begin{aligned}
E\dot{x} &= \sum_{i=1}^{4} (h_i A_i x) + Bu + B_f f \\
y &= Cx
\end{aligned}
\tag{8}
$$

considering the matrices:

$$
\begin{aligned}
E &= \begin{bmatrix} I & 0 \\ 0 & 0 \end{bmatrix} \quad A_i = \begin{bmatrix} \tilde{A}_i & 0 \\ 0 & -I_p \end{bmatrix} \\
B &= \begin{bmatrix} \tilde{B} \\ 0 \end{bmatrix} \quad C = \begin{bmatrix} \tilde{C} & I_p \end{bmatrix} \quad Bf = \begin{bmatrix} 0 \\ I_p \end{bmatrix}
\end{aligned}
\tag{9}
$$

This representation becomes a DAE, and has the extended state $x = [\tilde{x} \ f]^T$, for which a UIO convex observer given by (10) is designed.

$$
\begin{aligned}
\dot{z} &= \sum_{i=1}^{4} [h_i (N_i z + L_i y)] + Gu \\
\hat{x} &= z + T_2 y
\end{aligned}
\tag{10}
$$

Note that since the TS model (7) is linear respect to the inputs and outputs, also the scheduling variables are measurable, then the observer design is simplified. The error dynamics between the system (7) and observer (10) is given by (11).

$$
\dot{e} = N_i e + T_1 B_f f \tag{11}
$$

For (11) to be asymptotically stable, the conditions given by the following Theorem must hold.

Theorem 1. *Consider the TS convex system given by* (7)*, with the pair* (A_i, C) *observable. The error dynamics* (11) *is asymptotically stable, if there is* $P = P^T > 0$ *and* $Q_i > 0$ *such that the following holds:*

$$
\begin{bmatrix} He(PT_1 A - Q_i C) & PT_1 B_f & C^T \\ [PT_1 B_f]^T & -\beta^2 I & 0 \\ C & 0 & -I \end{bmatrix} < 0 \tag{12}
$$

Proof. The estimation error between the system (7) and the observer (10) is $e = x - \hat{x}$, by expansion:

$$
\begin{aligned}
e &= x - \hat{x} = x - (z + T_2 y) = \\
&= x - (z + T_2 C x) = (I - T_2 C)x - z
\end{aligned}
\tag{13}
$$

To simplify notation, the scheduling variable is omitted from the following equations. In order to consider the error dynamics the relation $I - T_2C = T_1E$ is defined, thus making:

$$\begin{aligned}\dot{e} &= T_1E\dot{x} - \dot{z} = T_1E\dot{x} - (Nz + Gu + Ly)\\ &= T_1(A_ix + Bu + B_ff) - N_iz - Gu - L_iCx\\ &= (T_1A_i - L_iC)x + (T_1B - G)u + T_1B_ff - N_i(\hat{x} - T_2Cx)\\ &= (T_1A_i - LC_i + N_iT_2C)x + (T_1B - G)u + T_1B_ff - N_i\hat{x}\end{aligned}$$

Noticing that $\hat{x} = x - e$, and by making $T_1B = G$:

$$\begin{aligned}\dot{e} &= (T_1A_i - L_iC + N_iT_2C)x + T_1B_ff - N_i(x - e)\\ &= (T_1A_i - L_iC - N_i[I - T_2C])x + T_1B_ff + N_ie\end{aligned}$$

Defining $L_i - N_iT_2 = K_i$, and $N_i = T_1A_i - K_iC_i$ it leaves:

$$\dot{e} = N_ie + T_1B_ff \tag{14}$$

Considering the Lyapunov error function:

$$\dot{V}(e) = e^TP\dot{e} + \dot{e}^TPe \tag{15}$$

Expanding the terms:

$$\begin{aligned}\dot{V}(e) &= e^TP([T_1A_i - K_iC]e + T_1B_ff)\\ &\quad + [(T_1A_i - K_iC)e + T_1B_ff]^TPe\\ &= e^T(P[T_1A_i - K_iC] + [T_1A_i - K_iC]^TP)e\\ &\quad + e^T(PT_1B_f)f + f^T(T_1B_f)^TPe\end{aligned} \tag{16}$$

Considering the $\mathcal{H}_\infty$ functional $J = V\dot{(}e) + r^Tr \leq \beta^2f^Tf$, and defining the residual $r = Ce$ the minimization is given by:

$$\begin{aligned}&e^T(P[T_1A_i - K_iC] + [T_1A_i - K_iC]^TP)e\\ &+ e^T(PT_1B_f)f + f^T(T_1B_f)^TPe + e^TC^TCe + f^Tf\end{aligned} \tag{17}$$

Defining $Q_i = PK_i$ and grouping:

$$\begin{bmatrix}e^T\\ f^T\end{bmatrix}\begin{bmatrix}He(PT_1A_i - Q_iC) + C^TC & PT_1B_f\\ [PT_1B_f]^T & -\beta^2I\end{bmatrix}\begin{bmatrix}e\\ f\end{bmatrix} \tag{18}$$

Finally applying Schur complement (12) is obtained, thus completing the proof. □

The design of the scheme can be summarized in the following steps:

1) Matrix computations $T_1 \& T_2$ with $[T_1\,T_2] = ([E \;\; C]^T)^\dagger$, and $[E \;\; C]^T$ is full rank.
2) $T_1B = G$, $Q_i = PK_i$.
3) LMI (12) resolution.
4) Matrix gain computations $K_i = P^{-1}Q_i$ y $N_i = T_1A_i - K_iC$.
5) Matrix gain computation: $L_i = K_i + N_iT_2$.

V. SIMULATION RESULTS

The designed convex UIO (10) was implemented by solving (12) using Yalmip Toolbox, [20], calculating the respective gains for its design. Due to space limitations the numerical results for matrices gains are omitted. Two scenarios are considered, a fault free case, and a multiple fault case. Simulation is done in Matlab.

A. *Fault free case*

Firstly, a fault free scenario is considered. For this case the non linear model is regulated to hover position, at 2 meters in the z_p axis as shown in Fig.3, the respective control input is shown in Fig. 4.

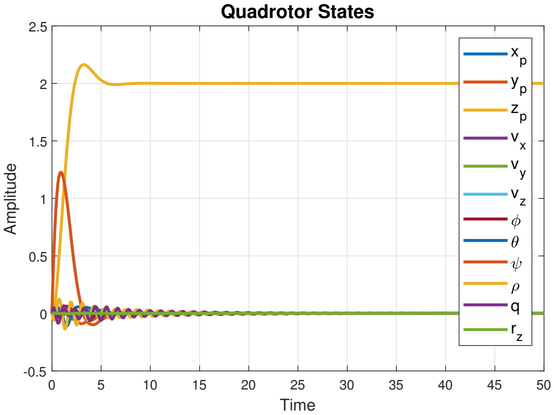


Fig. 3. Quadrotor states.

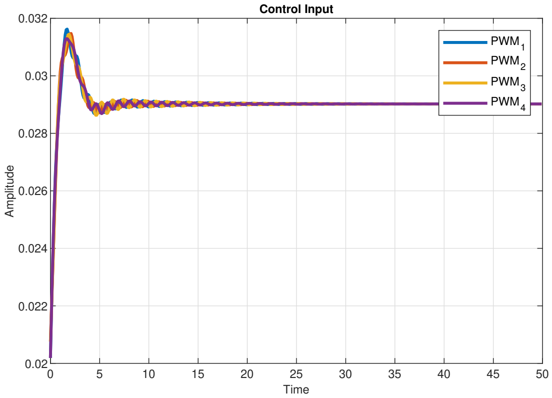


Fig. 4. Control inputs for the system.

In order to validate the TS model for the rotational dynamics a comparison was made with the nonlinear rotational dynamics, shown in Fig. 5, and the modeling error is presented in Fig. 6 then, the convex observer is validated both with the membership functions shown in Fig 7, and the estimation of faults in Fig. 8.

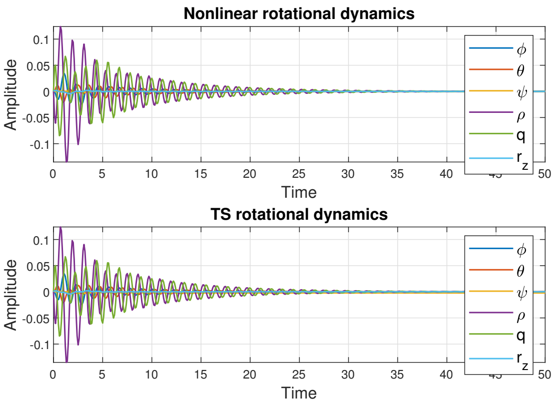


Fig. 5. Comparison between nonlinear system and TS model.

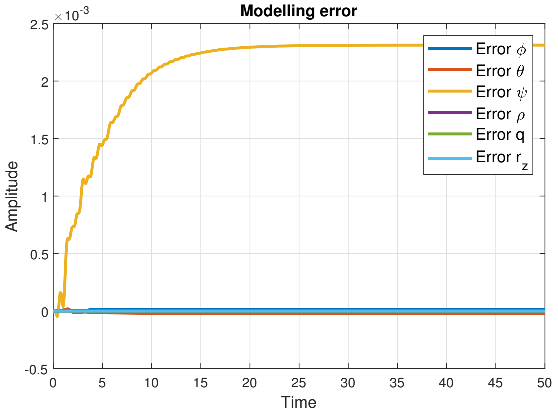


Fig. 6. Error between nonlinear model and TS model.

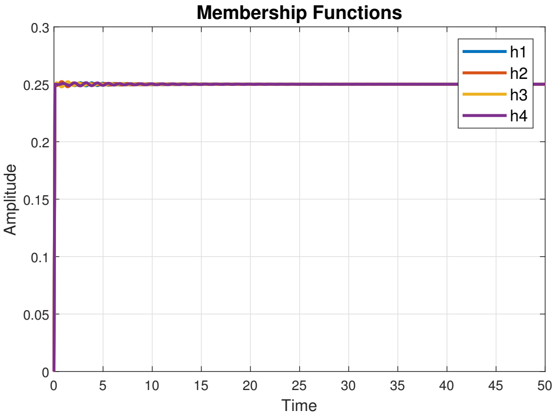


Fig. 7. TS membership functions.

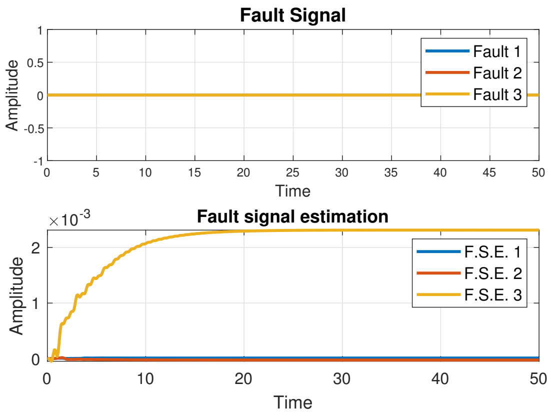


Fig. 8. Fault signals and fault free estimations.

As shown in Fig. 8 there is a minimum difference between the fault signals and their estimation in this scenario, this difference can be neglected, since it corresponds to the modeling error shown in Fig. 6, rendering the effectiveness of the proposed scheme satisfactory.

B. Multiple falt case

For this case, the quadrotor is also regulated to a hover position; in order to prove the effectiveness of the proposed observer scheme under fault, different faults in each sensor were applied at subsequent times. First, a 0.1 rad bias fault is applied to roll angle sensor from $10s$ to $20s$, this fault can be interpreted as a sensor disconnection. At $40s$ a sine fault signal of 0.1 rad and 0.16 Hz is applied to pitch angle sensor, representing a fault produced by external interference, and finally at $70s$ an incipient fault is applied to yaw angle sensor, interpreted as a deviation from the real value. Fault signals and estimations are shown in Fig. 9, and as seen, faults are correctly estimated despite their nature. The error is consistent with the one presented in Fig. 8. Due to the scope of this paper, it is considered that the faults do not affect the control scheme.

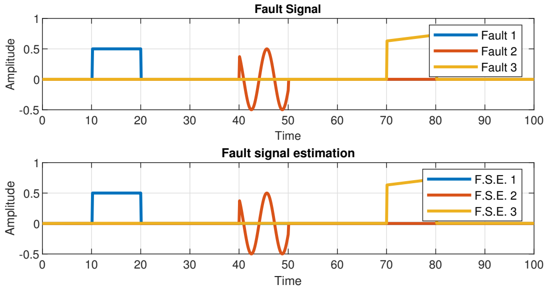


Fig. 9. Fault signals and estimations.

VI. CONCLUSION

In this paper, a fault estimation scheme for a quadrotor UAV is proposed using a UIO observer based on a TS convex model considering the rotational dynamics only in hover mode, and the simplified model of actuators. The proposed scheme was proven successful in a fault free scenario, and in presence of faults, despite its nature. This results from the fact that the descriptor scheme extends the state of the system and includes the fault signals in an algebraic manner, regardless of their dynamic behavior. Future work considers the application of the presented scheme to an active FTC control for the quadrotor.

REFERENCES

[1] M. Blanke, C. Frei, F. Kraus, R. Patton, and M. Staroswiecki, “What is fault-tolerant control,” *IFAC Proceedings Volumes*, vol. 33, 06 2000.

[2] H. Noura, D. Theilliol, J.-C. Ponsart, and A. Chamseddine, *Fault-Tolerant Control Systems: Design and Practical Applications*, 08 2009.

[3] J. Chen and R. Patton, *Robust Model-Based Fault Diagnosis for Dynamic Systems*, ser. The International Series on Asian Studies in Computer and Information Science. Springer US, 2012. [Online]. Available: https://books.google.com.mx/books?id=_wvrBwAAQBAJ

[4] M. Blanke, M. Kinnaert, J. Lunze, and M. Staroswiecki, *Diagnosis and Fault Tolerant Control*, 01 2016.

[5] J. Research. (2015, Nov) Consumer drone sales to reach almost 4m this year, rising to 16m annually by 2020. [Online]. Available: https://www.juniperresearch.com/press/consumer-drone-sales-to-reach-almost-4m-this-year

[6] B. Galkin, “L&rs spotlight: Consumer and commercial drones - how a technological revolution is impacting irish society,” *Oireachtas Library & Research Service 2021*, 2021.

[7] C. Berbra, S. Lesecq, and J. Martinez, “A multi-observer switching strategy for fault-tolerant control of a quadrotor helicopter,” in *2008 16th Mediterranean Conference on Control and Automation*, 2008, pp. 1094–1099.

[8] Y. Al Younes, H. Noura, A. Rabhi, A. El Hajjaji, and N. Al Hussien, “Sensor fault detection and isolation in the quadrotor vehicle using nonlinear identity observer approach,” in *2013 Conference on Control and Fault-Tolerant Systems (SysTol)*, 2013, pp. 486–491.

[9] H. A. Izadi, Y. Zhang, and B. W. Gordon, “Fault tolerant model predictive control of quad-rotor helicopters with actuator fault estimation,” *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 6343–6348, 2011, 18th IFAC World Congress. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1474667016446227

[10] P. Lu, E.-J. Van Kampen, and B. Yu, “Actuator fault detection and diagnosis for quadrotors,” in *IMAV 2014: International Micro Air Vehicle Conference and Competition 2014, Delft, The Netherlands, August 12-15, 2014*. Citeseer, 2014.

[11] Y. Zhong, Y. Zhang, W. Zhang, J. Zuo, and H. Zhan, “Robust actuator fault detection and diagnosis for a quadrotor uav with external disturbances,” *IEEE Access*, vol. 6, pp. 48 169–48 180, 2018.

[12] T. Li, Y. Zhang, and B. W. Gordon, “Passive and active nonlinear fault-tolerant control of a quadrotor unmanned aerial vehicle based on the sliding mode control technique,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 227, no. 1, pp. 12–23, 2013. [Online]. Available: https://doi.org/10.1177/0959651812455293

[13] A.-R. Merheb, H. Noura, and F. Bateman, “Active fault tolerant control of quadrotor uav using sliding mode control,” in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2014, pp. 156–166.

[14] R. C. Avram, X. Zhang, and J. Muse, “Quadrotor sensor fault diagnosis with experimental results,” *Journal of Intelligent & Robotic Systems*, vol. 86, no. 1, pp. 115–137, Apr 2017. [Online]. Available: https://doi.org/10.1007/s10846-016-0425-1

[15] D. Rotondo, F. Nejjari, and V. Puig, “Robust quasi-lpv model reference ftc of a quadrotor uav subject to actuator faults,” *International Journal of Applied Mathematics and Computer Science*, vol. 25, p. 2015, 03 2015.

[16] F. R. López-Estrada, J.-C. Ponsart, D. Theilliol, Y. Zhang, and C.-M. Astorga-Zaragoza, “Lpv model-based tracking control and robust sensor fault diagnosis for a quadrotor uav,” *Journal of Intelligent & Robotic Systems*, vol. 84, no. 1, pp. 163–177, Dec 2016. [Online]. Available: https://doi.org/10.1007/s10846-015-0295-y

[17] J. A. Guzmán-Rabasa, F. R. López-Estrada, B. M. González-Contreras, G. Valencia-Palomo, M. Chadli, and M. Pérez-Patricio, “Actuator fault detection and isolation on a quadrotor unmanned aerial vehicle modeled as a linear parameter-varying system,” *Measurement and Control*, vol. 52, no. 9-10, pp. 1228–1239, 2019. [Online]. Available: https://doi.org/10.1177/0020294018824764

[18] V. Mistler, A. Benallegue, and N. M’Sirdi, “Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback,” in *Proceedings 10th IEEE International Workshop on Robot and Human Interactive Communication. ROMAN 2001 (Cat. No.01TH8591)*, 2001, pp. 586–593.

[19] G.-R. Duan, *Equivalence and Solutions of Descriptor Linear Systems*. New York, NY: Springer New York, 2010, pp. 35–55. [Online]. Available: https://doi.org/10.1007/978-1-4419-6397-0_2

[20] J. Lofberg, “Yalmip : a toolbox for modeling and optimization in matlab,” in *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, 2004, pp. 284–289.

TecNM-Instituto Tecnológico de Tijuana y Asociación Mexicana de Robótica e Industria AC
XIII COMRob 2021, comrob.tijuana.tecnm.mx, ISBN: 978-607-99446-1-2
27 – 29 de Octubre, 2021,
Tijuana, Baja California, México

Passivity-Based Control Strategy for a Tricopter

Eduardo Delfín
TecNM/CENIDET
Cuernavaca, Morelos
Email: m20ce075@cenidet.tecnm.mx

Carlos D. Garcia Beltran
TecNM/CENIDET
Cuernavaca, Morelos
Email: carlos.gb@cenidet.tecnm.mx

Maria E. Guerrero Sanchez
Instituto Tecnologico de
Hermosillo, Sonora
Email: maria.guerrero@cenidet.edu.mx

***Abstract*—This paper presents a mathematical model and a passivity-based control strategy for a tricopter-type unmanned aerial vehicle with a tilt rotor arm. The mathematical model is based and represented on the Euler-Lagrange formulation and finally, a Hamiltonian formulation is derived. A passivity-based control by interconnection and damping assignment (IDA-PBC) is designed to control all states of a tricopter vehicle. The effectiveness of the proposed model and the designed controller are demonstrated by numerical simulations.**

Keywords: Passivity-based control by interconnection and damping assignment, unmanned aerial vehicle, port controlled hamiltonian.

1. Introducción

En la actualidad, las aplicaciones de los vehículos aéreos no tripulados (UAVs, por sus siglas en inglés) se han ido expandiendo de manera muy rápida en una gran variedad de industrias, gracias a las diversas tareas que pueden realizar estos vehículos. Los drones brindan ventajas sofisticadas en comparación con cualquier otro sistema de navegación, como lo es el monitoreo preciso de aquellas áreas que son de difícil acceso para el ser humano, rastreo de actividades ilegales, observación de incendios forestales, vigilancia de cultivos de grandes granjas agrícolas, etc.

Al depender de su configuración mecánica, los UAVs pueden utilizar rotores (uno, dos, tres o múltiples rotores) o utilizar alas flexibles [1]. Dentro de la categoría que hace uso de rotores se encuentra el UAV de tres motores comunmente llamado tricóptero, dicho sistema es objeto de estudio de este trabajo. El modelado matemático y control de un tricóptero es menos estudiado en la literatura en comparación con los UAVs del tipo cuadrotor y de ala fija, sin embargo, se han propuesto diferentes leyes de control para estos vehículos, por ejemplo, en [2] se utilizó la estrategia de control back-stepping para diseñar el controlador de actitud de vuelo del UAV, además se analiza la estructura mecánica y las tres posibles configuraciones del rotor del tricóptero. En [3] se presenta un enfoque de control predictivo, como método de control activo tolerante a fallas, aplicado a un UAV tricóptero coaxial de seis motores en presencia de una falla del actuador respetando las restricciones reales. Mientras que [4] introduce el diseño de un controlador proporcional, integral, derivativo para un tricóptero con alas fijas. Además, en [5] se desarrolla el modelo matemático y un controlador predictivo para un sistema de tres motores.

Por otro lado, en nuevas investigaciones donde se aplica el control (IDA-PBC) a vehículos aéreos no tripulados, se encuentran por ejemplo, en [6] donde se aborda el problema de control de UAVs considerados totalmente actuados en un marco geométrico hamiltoniano, el UAV se modela como un cuerpo rígido flotante y se obtiene un controlador unificado de movimiento e impedancia mediante la técnica de control basada en pasividad. En [7] se representa a un cuadrotor como un sistema hamiltoniano y se propone un controlador solamente para la rotación del vehículo. Además, en [8] se muestran resultados sobre el modelado y control de un vehículo aéreo no tripulado tipo cuadrotor que transporta una carga suspendida por un cable utilizando la metodología IDA-PBC. Mientras que en [9] se lleva cabo un análisis de la dinámica y un diseño de un controlador para un cuadrotor manipulador en un plano arbitrario a través de la metodología IDA-PBC. En [10] se introduce el diseño de una estrategia IDA-PBC para la regulación de un vehículo submarino subactuado, en el cual se consideran seis grados de libertad y cuatro propulsores como actuadores. También, en [11] se presenta el desarrollo y la implementación práctica de un algoritmo de control basado en pasividad para estabilizar un vehículo aéreo no tripulado descrito con cuaterniones unitarios.

Debido a que la estrategia IDA-PBC se ha empleado con éxito en los UAVs tipo cuadrotor, en este trabajo se ha propuesto utilizar la técnica de control no lineal IDA-PBC para el vehículo tricóptero, debido a que dentro de las multiples propiedades con las que cuentan estos vehículos la pasividad en concreto no ha sido investigada exaustivamente. Por tanto este trabajo tiene como objetivo aplicar una estrategia de control no lineal, que aproveche la propiedad de pasividad de un sistema tricóptero, con el objetivo de resolver el problema de regulación, seguimiento de trayectoria y de rotación producida en el eje yaw por la forma asimétrica que tienen estos vehículos al contar únicamente con tres motores.

Este artículo presenta la siguiente estructura: en la Sección 2, se describe el modelo matemático en las representacion Euler-Lagrange de un tricóptero. En la Sección 3, se desarrolla el control por asignación de interconexión y de amortiguamiento para el caso de estudio. En la Sección 4, se muestran las simulaciones numéricas. Finalmente, en la Sección 5, se dan las conclusiones.

2. Modelo Matemático

Se presenta una descripción del modelo matemático de un vehículo tricóptero. La Figura 1 muestra el diagrama de cuerpo libre del vehículo. Este sistema es subactuado, debido a que cuenta con seis grados de libertad y cuatro entradas de control.
La notación es usada de acuerdo a la Figura 1, una referencia inercial de coordenadas $\Gamma = [X, Y, Z]$, se considera con posiciones fijas en la tierra, mientras que $B = [X_b, Y_b, Z_b]$ son las coordenadas fijas al cuerpo, las cuales coinciden con el centro de masa del tricóptero.

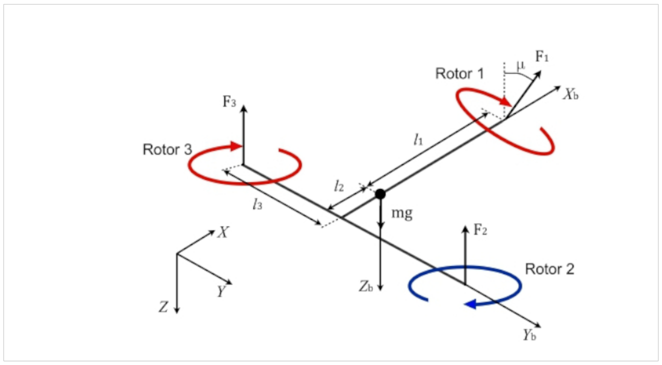


Figure 1. Representación esquemática del tricóptero

La configuración de variables son $q = [\xi, \eta]^T \in \Re^6$, donde $\xi = [x, y, z]^T \in \Re^3$ indican la posición del centro de masa del tricóptero relativo a la referencia inercial Γ, $\eta = [\phi, \theta, \psi] \in \Re^3$ son los ángulos de Euler: roll, pitch y yaw, respectivamente. l_1, l_2 y l_3 son las distancias entre los motores con respecto al centro de masa. Por último, F_1, F_2 y F_3 representan las fuerzas de empuje proporcionadas por cada uno de los motores.
La entrada de control se define como $u = [F \;\; \tau]^T \in \Re^4$, donde $F = F_1 + F_2 + F_3$ es la magnitud de la fuerza total de empuje, $\tau = [\tau_\phi \;\; \tau_\theta \;\; \tau_\psi]$, $\tau_\phi = L_3(F_3 - F_2)$, $\tau_\theta = l_1 F_1 Cos(\mu) - l_2(F_2 + F_3)$, $\tau_\psi = l_1 F_1 sin(\mu) - M_1 cos(\mu) + M_2 - M_3 = l_1 K_F \Omega_1^2 sin(\mu) - K_M \Omega_1^2 cos(\mu) + K_M \Omega_2^2 - K_M \Omega_3^2$.

Ω_i representa la velocidad angular del i-ésimo motor, mientras que μ representa el ángulo de inclinación del motor batiente del tricóptero, por otra parte las constantes K_F y K_M son la fuerza aerodinámica y la constante de momento, respectivamente.

Mientras que M_i y F_i suponiendo que el cambio de altitud es limitado y, por lo tanto, la densidad del aire es constante, el momento y la fuerza aerodinámica producida por el i-ésimo rotor usando la fuerza aerodinámica y las constantes de momento están dados por:

$$F_i = K_F \Omega_i^2 \tag{1}$$

$$M_i = K_M \Omega_i^2 \tag{2}$$

La matriz de rotación del marco del cuerpo al inercial, usando la notación corta $S\phi = \sin\phi$ y $C\phi = \cos\phi$ se expresa como:

$$R = \begin{bmatrix} C\psi C\theta & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ S\psi C\theta & S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & C\theta S\phi & C\theta C\phi \end{bmatrix}$$

2.1. Ecuaciones Euler-Lagrange

La representación matemática siguiente resulta a través de la formulación Euler-Lagrange. Donde la energía cinética y la potencial se presentan con el objetivo de formular el lagrangiano del sistema. La función de la energía cinética total $K(q, \dot{q})$ del tricóptero, resultante del movimiento de traslación y rotación se denota como:

$$K = \frac{1}{2}\dot{\xi}^T P \dot{\xi} + \frac{1}{2}\dot{\eta}^T I \dot{\eta} \tag{3}$$

donde $I \in \Re^{3x3}$ es la matriz de inercia respecto a B, la cual funge como la matriz inercial para la energía cinética rotacional completa del tricóptero, se considera que el vehículo trabaja cerca de $\eta = 0$ y se supone diagonal:

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{4}$$

Mientras que I_{xx}, I_{yy}, I_{zz} representan los momentos de inercia del tricóptero y $P \in \Re^{3x3}$ es la matriz diagonal de la masa del tricoptero.

La función de energía potencial del sistema total $P_o(q)$ es

$$P_o(q) = -mgz \tag{5}$$

Usando (3) y (5), el lagrangiano resulta de la siguiente manera:

$$L = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2 + \dot{z}^2 +) + \frac{1}{2}(\dot{\phi}^2 I_{xx} + \dot{\theta}^2 I_{yy} + \dot{\psi}^2 I_{zz}) + mgz \tag{6}$$

Se obtienen las ecuaciones que describen la dinamica total del sistema aplicando la formulación Euler-Lagrange, que se expresan en forma matricial como:

$$M\ddot{q} + G = b(q)u \tag{7}$$

donde $M \in \Re^{6x6}$ es la matriz inercial, la cual es simétrica y definida positiva, $G(q) \in \Re^6$ es el vector gravitacional y $b(q) \in \Re^{6x4}$ es la matriz de entradas del sistema $u \in \Re^4$, esta matriz no es invertible, debido a que el sistema es subactuado. Dichas matrices son:

$$M = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} \end{bmatrix}$$

$$G = \begin{bmatrix} 0 & 0 & mg & 0 & 0 & 0 \end{bmatrix}^T$$

$$b = \begin{bmatrix} C\psi S\theta C\phi + S\psi S\phi & 0 & 0 & 0 \\ S\psi S\theta C\phi - C\psi S\phi & 0 & 0 & 0 \\ C\theta C\phi & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

3. ESTRATEGIA DE CONTROL

La estrategia de control IDA-PBC se implementa al modelo con energía total

$$H_{am}(q,p) = \frac{1}{2}p^T M^{-1}(q)p + P_o(q) \tag{8}$$

donde $q \in \Re^3$, $p \in \Re^3$, son las posiciones y los momentos generalizados, respectivamente y $H_{am}(q,p)$ es el Hamiltoniano. Las ecuaciones de movimiento se pueden denotar como [12]

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \begin{bmatrix} \nabla_q H_{am} \\ \nabla_p H_{am} \end{bmatrix} + \begin{bmatrix} 0 \\ b(q) \end{bmatrix} u \tag{9}$$

donde $I_n \in \Re_{3x3}$ es una matriz identidad, $\nabla_q H_{am} = \frac{\partial H_{am}}{\partial q}$ y $\nabla_p H_{am} = \frac{\partial H_{am}}{\partial p}$.
La función de energía deseada se propone al sistema como:

$$H_{amd}(q,p) = \frac{1}{2}p^T M_d^{-1}(q)p + P_{od}(q) \tag{10}$$

donde $M_d = M_d^T > 0$ representan la matriz inercial deseada en lazo cerrado y P_{od} la función de energía potencial. Es necesario que la función P_{od} tenga un mínimo en q^*, esto es

$$q^* = \arg\min P_{od}(q) \tag{11}$$

La entrada de control en el enfoque IDA-PBC, es descompuesta en dos términos

$$u = u_e s(q,p) + u_{di}(q,p) \tag{12}$$

Siendo u_{es} el moldeo de energía y u_{di} la inyección de amortiguamiento. La dinámica hamiltoniana controlada por puerto deseada tiene la siguiente forma [13]

$$\begin{bmatrix} \dot{q} \\ \dot{p} \end{bmatrix} = \begin{bmatrix} J_d(q,p) & -R_d(q,p) \end{bmatrix} \begin{bmatrix} \nabla_q H_{am} \\ \nabla_p H_{am} \end{bmatrix} \tag{13}$$

donde

$$J_d = -J_d^T = \begin{bmatrix} 0 & M^{-1}M_d \\ -M^{-1}M_d & J_2(q,p) \end{bmatrix} \tag{14}$$

$$R_d = -R_d^T = \begin{bmatrix} 0 & 0 \\ 0 & bK_v b^T \end{bmatrix} \geq 0 \tag{15}$$

J_2 es una matriz antisimétrica y $K_v = K_v^T > 0$ debe cumplirse.

La inyección de amortiguamiento se expresa como

$$u_{di}(q,p) = -K_v b^T \nabla_p H_{amd} \tag{16}$$

Para obtener el moldeo de energía u_{es} del controlador, se sustituye (12) y (16) en (9) y se iguala con (13) resultando

$$\begin{gathered} \begin{bmatrix} 0 & I_n \\ -I_n & 0 \end{bmatrix} \begin{bmatrix} \nabla_q H_{am} \\ \nabla_p H_{am} \end{bmatrix} + \begin{bmatrix} 0 \\ b(q) \end{bmatrix} u_{es} = \\ \begin{bmatrix} J_d(q,p) & -R_d(q,p) \end{bmatrix} \begin{bmatrix} \nabla_q H_{am} \\ \nabla_p H_{am} \end{bmatrix} \end{gathered} \tag{17}$$

El primer renglón de las ecuaciones se satisface a simple vista. En el segundo, el conjunto de ecuaciones diferenciales parciales dan el término del moldeo de la energía dado por

$$u_{es} = b^{\perp}(\nabla_q H_{am} - M_d M^{-1} \nabla_q H_{amd} + J_2 M_d^{-1} p) \tag{18}$$

La Ecuación (17) puede separarse en las siguientes expresiones:

$$b^{\perp} \nabla_q (p^T M^{-1} p) - M_d M^{-1} \nabla_q (p^T M_d^{-1} p) + 2J_2 M^{-1} p = 0 \tag{19}$$

$$b^{\perp} \nabla_q P_o - M_d M^{-1} \nabla_q P_{od} = 0 \tag{20}$$

Siendo $b^{\perp}$ el anulador por la izquierda de rango completo de $b(q)$.

En el moldeo de energía, se observa que la matriz M es independiendte de q, por lo que se puede tomar $J_2 = 0$ y M_d como una matriz constante, es decir,

$$M_d = \begin{bmatrix} m & 0 & 0 & 0 & a_5 & 0 \\ 0 & m & 0 & a_4 & 0 & 0 \\ 0 & 0 & m & 0 & 0 & 0 \\ 0 & a_4 & 0 & I_{xx} & 0 & 0 \\ a_5 & 0 & 0 & 0 & I_{yy} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} \end{bmatrix}$$

Entonces el controlador se reduce a

$$u_{es} = (b^T b)^{-1} b^T (\nabla_q H_{am} - M_d M^- 1 \nabla_q H_{amd}) \tag{21}$$

con $b^{\perp} = [(\frac{s\phi c\psi - s\psi s\theta c\phi}{c\phi c\psi s\theta + s\psi s\phi} - c\theta c\phi), \ (c\theta c\phi), \ (c\phi c\psi s\theta + s\psi s\phi)]$, la ecuación diferencial potencial se obtiene como:

$$\begin{gathered} (\frac{s\phi c\psi - s\psi s\theta c\phi}{c\phi c\psi s\theta + s\psi s\phi} - c\theta c\phi)(-\frac{\partial P_{od}}{\partial x} - \frac{a_5 \partial P_{od}}{I_{yy} \partial \theta}) + \\ (c\theta c\phi)(-\frac{\partial P_{od}}{\partial y} - \frac{a_4 \partial P_{od}}{I_{xx} \partial \theta}) + \\ (c\phi c\psi s\theta + s\psi s\phi)(-mg - \frac{\partial P_{od}}{\partial z}) = 0 \end{gathered}$$

La cual se resuelve obteniendo la energía potencial deseada como

$$P_{od} = -mgz + \Phi$$

donde Φ es una función diferenciable de $\left((\theta - \theta_d) - \frac{a_5}{I_{yy}}(x - x_d)\right)$, $\left(\phi - \phi_d) - \frac{a_4}{I_{xx}}(y - y_d)\right)$

y $\left((z-z_d)-\frac{I_{xx}}{a_4}\frac{(S\theta C\psi)(\phi-\phi_d)-S\psi Ln(C\phi)}{C\theta}\right)$. La obtención de Φ depende de la condición dada en (11). Para esto la condición necesaria, $\nabla_q P_{od}(q_*) = 0$ y la condición suficiente $\nabla_q^2 P_{od}(q_*) > 0$ se mantendrá si el Hessiano de $\Phi(x,y,z,\phi,\theta,\psi)$ en q_* es positivo. En este caso, se elige $\Phi(x,y,z,\phi,\theta,\psi)$ como una función cuadrática tal que:

$$P_{od} = -mgz + \frac{1}{2}Kpx\left((\theta-\theta_d)-\frac{a_5}{I_{yy}}(x-x_d)\right)^2$$
$$+\frac{1}{2}Kpy\left((\phi-\phi_d)-\frac{a_4}{I_{xx}}(y-y_d)\right)^2$$
$$+\frac{1}{2}Kpz\left((z-z_d)-\frac{I_{xx}}{a_4}\frac{(S\theta C\psi)(\phi-\phi_d)-C\psi Ln(C\phi)}{C\theta}\right)^2$$

En la cual $(x_d, y_d, z_d, \phi_d, \theta_d)$ representan la configuración en equilibrio y Kpx, Kpy, Kpz son las ganancias del controlador las cuales se usan como términos de sintonización. La ley de control, se determina al obtener el moldeo de energía u_{es} de (18), tomando la forma de:

$$u_{es} = \begin{bmatrix} Kpz\gamma(\frac{a_5 I_{xx}}{I_{yy}a_4}\omega\rho \ +\xi-1) \\ Kpy\frac{a_4^2}{m_I xx}\beta - Kpy\beta + Kpz\frac{I_{xx}}{a_4}\varsigma\gamma \\ Kpx\frac{a_5^2}{mI_{yy}}\alpha - Kpx\alpha + Kpz\frac{I_{xx}}{a_4}\rho\gamma \\ -Kpz\frac{I_{xx}}{a_4}\sigma\gamma \end{bmatrix} \qquad (22)$$

donde

$$\alpha = \left((\theta-\theta_d)-\frac{a5}{I_{yy}}(x-x_d)\right)$$

$$\beta = \left((\phi-\phi_d)-\frac{a4}{I_{xx}}(y-y_d)\right)$$

$$\gamma = \left((z-z_d)-\frac{I_{xx}}{a4}\frac{(S\theta C\psi)(\phi-\phi_d)-S\psi Ln(C\phi)}{C\theta}\right)$$

$$\xi = \left(\frac{(S\psi S\theta C\phi - C\psi S\phi)(C\psi S\theta C\phi + S\psi S\phi)}{C\theta C\phi}\right)$$

$$\rho = \left(\frac{S\theta((S\theta C\psi)(\phi-\phi_d)-S\psi Ln(C\phi))}{C^2\theta} - C\psi(\phi-\phi_d)\right)$$

$$\sigma = \left(\frac{(S\theta S\psi)(\phi-\phi_d)+C\psi Ln(C\phi)}{C\theta}\right)$$

$$\varkappa = (C\psi S\theta C\phi + S\psi S\phi)$$

$$\varsigma = \left(\frac{(C\psi S\theta C\phi + S\psi S\phi)}{C\theta C\phi}\right)$$

La inyección de amortiguamiento u_{di} se determina de (20).

$$u_{di} = K_v\begin{bmatrix} rI_{yy}\dot{X} - fa_5\dot{\theta} + na_4\dot{\phi} - vI_{xx}\dot{Y} + \dot{Z} \\ va_4\dot{Y} - nmp\dot{h}i \\ fm\dot{\theta} - ra_5\dot{X} \\ \frac{\dot{\psi}}{I_{yy}} \end{bmatrix} \qquad (23)$$

donde

$$r = \frac{m}{mI_{yy}-a_5^2}, \qquad f = \frac{I_{yy}}{mI_{yy}-a_5^2},$$

$$v = \frac{m}{a_4^2-mI_{xx}}, \qquad n = \frac{I_{xx}}{a_4^2-mI_{xx}},$$

4. RESULTADOS Y SIMULACIONES

La simulación tiene como objetivo mover el tricóptero desde una posición inicial a una posición deseada. Los parámetros del modelo se usaron cerca de los que tienen las plataformas de tricópteros reales. Dichos parámetros se muestran en la Tabla 1.Los parámetros de calibración del control se ajustaron manualmente hasta obtener las mejores respuestas. Las ganancias para el moldeo de energía u_{es} obtenidas fueron $kp_z = 70$. $kp_x = 100$, $kp_y = 700$. En las Figuras 2 y 3 se muestran los resultados de las posiciones translacionales y rotacionales del vehículo, con las condiciones de equilibrio deseadas como $z_d = -1.5$, $x_d = -5$, $y_d = 0$ y con un ángulo de inclinación en roll de $\phi_d = 11$ grados para contrarrestar los efectos de traslación en el eje Y debido a la asimetría de la rotación de los motores.

Parámetros	Descripción	Unidades
m	Masa del tricóptero	1.1 kg
I_{xx}	Momento de inercia eje X	0.0239 k/m^2
I_{yy}	Momento de inercia eje Y	0.01271 k/m^2
I_{zz}	Momento de inercia eje Z	0.01273 k/m^2
l_1	distancia 1	0.2483 m
l_2	distancia 2	0.1241 m
l_2	distancia 3	0.2150 m
K_F	Constante de fuerza aerodinámica	$1.970x10^{-6}$
K_M	Constante del momento aerodinámico	$2.880x10^{-7}$

Table 1. TABLA DE PARÁMETROS

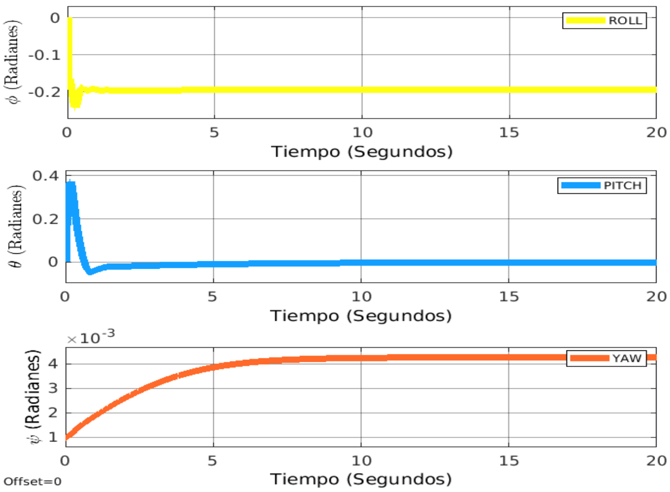


Figure 2. Dinámica de la rotación del tricóptero.

En la Figura 2 se observa como las 3 posiciones angulares son reguladas por el controlador, llegando rápidamente a las referencias deseadas, con tiempos de estabilización de 5 segundos para el pitch, 2 segundos para el roll y el ángulo yaw es estabilizado en un tiempo muy pequeño.

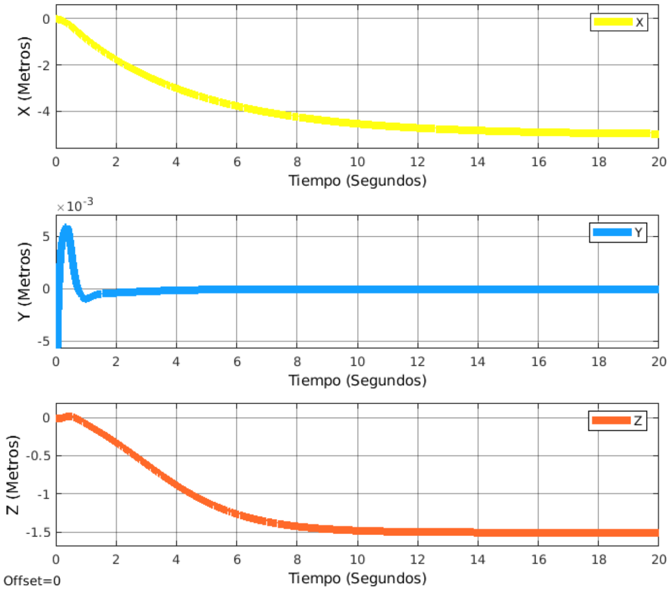


Figure 3. Dinámica de la posición del tricóptero.

En la Figura 3 se muestra la trayectoria que sigue el UAV para llegar a sus referencias deseadas para cada eje en particular, con tiempos de estabilizacion de aproximadamente 12 segundos para x y 15 segundos para z, teniendo respuestas suaves y sin sobreimpulsos, se puede analizar de igual manera de las graficas que las coordenadas longitudinales son negativas esto debido a la convención utilizada en el plano cartesiano, ya que el ascender y el avanzar son representados de dicha forma.

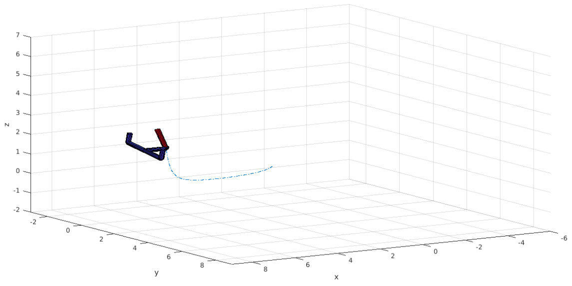


Figure 4. Representación vectorial del tricóptero.

La Figura 4 muestra la representacion tridimensional de la trayectoria recorrida por el tricóptero, en donde se observa una respuesta satisfactoria.

En la Figura 5 se observan las entradas de control, las cuales se mantienen estables durante el tiempo de simulación, sin presentar oscilaciones en sus respuestas.

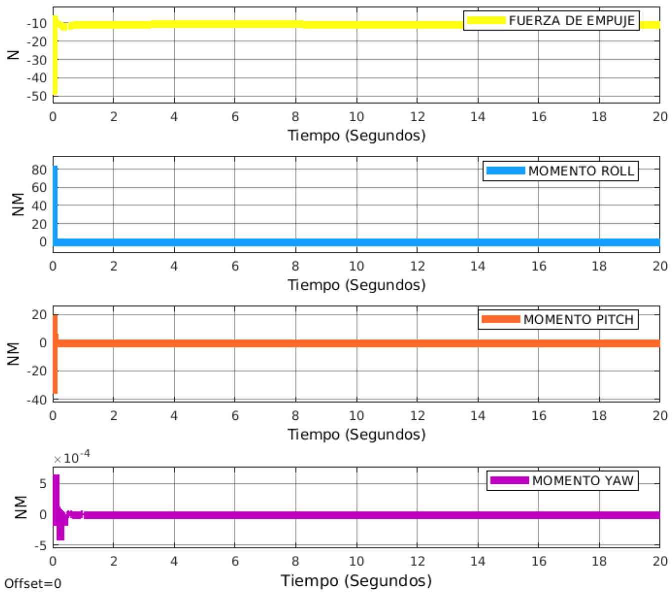


Figure 5. Entradas de control.

5. CONCLUSIONES

En este trabajo, se ha desarrollado un modelo matemático y un controlador no lineal basado en pasividad para un vehículo aéreo no tripulado tipo tricóptero. El modelo matemático se representó en la formulacion Euler-Lagrange. El algoritmo IDA-PBC tuvo un resultado satisfactorio en el control de todos los estados del sistema, teniendo un buen comportamiento en las regulaciones de las posiciones translacionales y rotacionales.

References

[1] P. Castillo, P. García, R. Lozano, and P. Albertos, “Modelado y estabilización de un helicóptero con cuatro rotores,” *Revista Iberoamericana de Automática e Informática industrial*, vol. 4, no. 1, 2009.

[2] Z. Song, K. Li, Z. Cai, Y. Wang, and N. Liu, “Modeling and maneuvering control for tricopter based on the back-stepping method,” in *2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, 2016, pp. 889–894.

[3] M. Mehndiratta and E. Kayacan, “Reconfigurable fault-tolerant nmpc for y6 coaxial tricopter with complete loss of one rotor,” in *2018 IEEE Conference on Control Technology and Applications (CCTA)*, 2018, pp. 774–780.

[4] A. Summers, “Modeling and Control of a Fixed Wing Tilt-Rotor Tri-Copter,” Ph.D. dissertation, University of Washington, Jan. 2017.

[5] A. Prach and E. Kayacan, “An mpc-based position controller for a tilt-rotor tricopter vtol uav,” *Optimal Control Applications and Methods*, vol. 39, no. 1, pp. 343–356, 2018.

[6] S. S. R. Rashad, F. Califano, “Interconnection and damping assignment passivity-based control of port-controlled hamiltonian systems,” *IEEE ROBOTICS AND AUTOMATION LETTERS*, vol. 4, no. 4, 2019.

[7] K. Hu, Y. Wu, and X.-M. Sun, “Attitude controller design for quadrotors based on the controlled hamiltonian system,” *2017 29th Chinese Control And Decision Conference (CCDC)*, pp. 5137–5141, 2017.

[8] M. E. Guerrero, D. A. Mercado, R. Lozano, and C. D. García, “Ida-pbc methodology for a quadrotor uav transporting a cable-suspended payload,” *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 470–476, 2015.

[9] J. Acosta, M. Sánchez, and A. Ollero, “Robust control of underactuated aerial manipulators via ida-pbc,” *53rd IEEE Conference on Decision and Control*, pp. 673–678, 2014.

[10] D. García, J. Sandoval, J. Gutiérrez–Jagüey, and E. Bugarin, “Control ida-pbc de un vehículo submarino subactuado,” *Revista Iberoamericana de Automática e Informática industrial*, vol. 15, no. 1, pp. 36–45, 2017.

[11] M. E. Guerrero-Sanchez, H. Abaunza, P. Castillo, R. Lozano, C. Garcia-Beltran, and A. Rodriguez-Palacios, “Passivity-based control for a micro air vehicle using unit quaternions,” *Applied Sciences*, vol. 7, no. 1, p. 13, 2017.

[12] B. M. y. G. E. R. Ortega, A. van der Schaft, “Interconnection and damping assignment passivity-based control of port-controlled hamiltonian systems,” *Automatica*, vol. 38, no. 4, pp. 585–596, 2002.

[13] R. Ortega, M. Spong, F. Gomez-Estern, and G. Blankenstein, “Stabilization of a class of underactuated mechanical systems via interconnection and damping assignment,” *IEEE Transactions on Automatic Control*, vol. 47, no. 8, pp. 1218–1233, 2002.

Motick Kids: Una herramienta orientada a la población infantil con deficiencias motoras

Pérez Silva Ricardo Bladimir
Ingeniería Biomédica
Instituto Tecnológico Superior de Pátzcuaro, México
persilricvla@gmail.com

Soria Medina Abraham Estéfano
Ingeniería Biomédica
Instituto Tecnológico Superior de Pátzcuaro, México
estefano01020@gmail.com

Velázquez Izguerra Joel
Ingeniería Biomédica
Instituto Tecnológico Superior de Pátzcuaro, México
joelvizguerra@hotmail.com

Peñaloza Mendoza Guillermo R.
Ingeniería Biomédica
Instituto Tecnológico Superior de Pátzcuaro, México
grey@itspa.edu.mx

Campos Hernández Alicia
Ingeniería Biomédica
Instituto Tecnológico Superior de Pátzcuaro, México
acampos@itspa.edu.mx

Castro Zenil Mario Salvador
Ingeniería Biomédica
Instituto Tecnológico Superior de Pátzcuaro, México
mcastro@itspa.edu.mx

Abstract— **En la actualidad, la aplicación de la robótica en apoyo a las personas con alguna discapacidad se encuentra en todas las áreas, sin embargo, en su mayoría se centran en suplir una función o miembro del cuerpo. El presente trabajo aborda el diseño y construcción de un dispositivo trazador controlado por joystick, llamado Motick Kids, el cual se encuentra dirigido a la población infantil que padece alguna malformación congénita en las extremidades superiores o alguna deficiencia motora, brindándoles una alternativa simple de reacondicionamiento, que les permita expresarse de forma escrita y pictórica. La construcción de este dispositivo se realiza a partir de un sistema de desplazamiento xyz, por medio de motores a pasos, en los ejes xy, mientras que el eje z cuenta con dos posiciones, las cuales son alcanzadas empleando un servomotor. El desplazamiento en los tres ejes es controlado a través de un joystick, que permite desplazar un marcador o un lápiz sobre una superficie para poder realizar los trazos.**

Keywords*— *Joystick, Malformación congénita, Mesa xyz.

I. INTRODUCCIÓN

De acuerdo con la Academia Americana de Pediatría [1], entre el 1% y 2% de los recién nacidos con defectos congénitos, un 10% nace con malformaciones de la mano. Dependiendo del tipo y el tamaño de la malformación, es posible que algunos bebés tengan pocos problemas para adaptarse y lograr un desempeño correcto durante su desarrollo.

Lamentablemente algunos pueden enfrentar diferentes desafíos a medida que crecen y aprenden, y al ser una pequeña población, no hay gran cantidad de opciones disponibles para ayudar a su correcto desarrollo. Algunas de las malformaciones, [2, 3] pueden apreciarse en la Fig. 1, estas pueden ser:

- Polidactilia: Pulgar adicional o dedos adicionales al lado del meñique.
- Sindactilia: Dos o más dedos no logran separarse cuando el bebé está en el útero.
- Simbraquidactilia: Dedos pequeños o faltantes.
- Mano zamba: Falta todo o parte de uno de los dos huesos largos que forman el antebrazo.
- Ectrodactilia: Ausencia de uno o más dedos del centro de la mano.

Por otra parte, de acuerdo con el Instituto Nacional de Trastornos Neurológicos y Accidentes Cerebrovasculares [4], uno de cada 200 niños que nacen en México presenta un tipo de autismo conocido como síndrome de Asperger, caracterizado por un mayor o menor impedimento en las habilidades del lenguaje y la comunicación, además de patrones repetitivos o restringidos de pensamiento y comportamiento.

Este síndrome se caracteriza por: intereses limitados o una preocupación inusual con un objeto en particular; peculiaridades en el habla y el lenguaje, como hablar de manera demasiado monótona; comportamiento social y emocionalmente inadecuado y la incapacidad de interactuar exitosamente con los demás.; y movimientos motores torpes y no coordinados.

El trabajo desarrollado se encuentra organizado de la siguiente manera: la sección 2 presenta el Estado del Arte, donde se abordan algunas tecnologías en pro de las personas con discapacidad motriz, en la sección 3 se presenta el Planteamiento del problema, mientras que la sección 4 presenta la Solución propuesta mencionando la justificación y estrategia a usar, la sección 5 presenta Diseño y Construcción de la herramienta a emplear. Por último, las secciones 6, 7 y 8 muestran la Operación, Resultados y Conclusiones obtenidas con la aplicación de la herramienta didáctica.

Fig. 1.- Casos de ectrodactilia y simbraquidactilia

II. ESTADO DEL ARTE

El desarrollo de tecnologías para el apoyo a personas con alguna discapacidad, ha tenido un crecimiento acelerado en la última década. En los trabajos de Roman Zahariev, et al, [5, 6] se describe el desarrollo de prototipos robóticos antropomórficos con inteligencia para su uso en servicio y asistencia a personas con discapacidades, particularmente en el proceso de educación especial de niños con necesidades.

En [7], se describen dos aplicaciones hápticas las cuales permiten tomar información perceptiva visual y háptica, cuyo objetivo es el reconocimiento de la escritura a mano y ayudar a los usuarios en una tarea de dibujo, actuando como una guía virtual.

Mientras tanto, la tecnología aplicada en accesorios para el apoyo a personas con malformaciones congénitas en las manos o con problemas de motricidad fina se engloban en las siguientes categorías:

- Cubiertos adaptados: Existen diversos cubiertos de distintos tamaños y formas adaptables, para diferentes malformaciones. Así mismo, existen diferentes objetos diseñados para colocarse sobre la mano para actividades específicas [8]. La Fig. 2 muestra ejemplo de este tipo de cubiertos.
- Juegos para motricidad: La psicomotricidad fina en niños se trabaja a través de juegos de pinzas, plastilina, pajitas, circuitos de bolas, como se muestra en la Fig., 3, y sencillas manualidades con las que, además de entrenar, pasarán divertidos a ratos [9].
- Objetos de la vida cotidiana: Actualmente existen diferentes objetos adaptables, que les permiten llevar a cabo tareas como abotonar una camisa, subir una cremallera o tocas un instrumento musical [8], como se observan en la Fig. 4.

Fig. 2.- Cubiertos adaptables. a).- Mango adaptable, b).- Curvos, c).- Encorvados, y d).- Con correa adaptable. Imagen obtenida de https://ortopediaencasa.com/

Fig. 3.- Juego para motricidad: Circuito con bolas. Imagen obtenida de https://kinuma.com/es/motricidad-habilidad/7828-lazo-de-motricidad-circuito-de-bolas-4013594599825.html

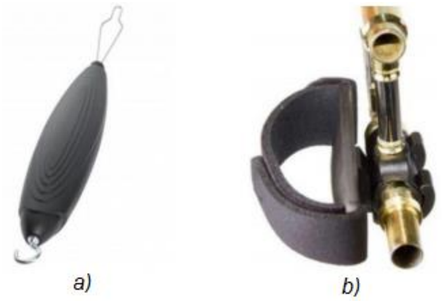


Fig. 4.- Objetos de la vida cotidiana. a).- Abrochador de botones y cremallera, b).- Adaptador para instrumento musical. [8]

III. PLANTEAMIENTO DEL PROBLEMA

Las malformaciones y la falta de motricidad fina pueden derivar en una afección para el desarrollo de la niñez, puesto que tendrán que convivir con otros niños, los cuales pueden dirigir burlas o críticas al aspecto físico del infante, lo que origina cuadros de depresión infantil. Las terapias suelen ser complicadas, pues existen pocos dispositivos que les permitan desarrollar la capacidad de escribir o dibujar, además, las terapias con juegos son dirigidas específicamente a niños menores de 6 años, ya que a mayor edad suele aburrirles y pierden el interés. Por lo anterior descrito, los problemas a resolver son:

a) Los prototipos existentes en el mercado que ayudan a los niños con malformaciones en sus manos son dirigidos en su mayoría como accesorios adaptativos y no cumplen la función de rehabilitación.

b) Los juegos existentes para apoyo con la motricidad en niños con malformaciones en sus manos son por etapas de crecimiento con una función especifica en la rehabilitación.

c) No existen prototipos en el mercado que permita la potencialización de la expresión escrita o gráfica, como medio de rehabilitación en malformaciones en manos

IV. SOLUCIÓN PROPUESTA

El desarrollo infantil de niños con discapacidad motriz o deformación de extremidades superiores, no implica en el cien por ciento de los casos de una infancia difícil, sin embargo, si presentan dificultades para realizar actividades debido a su malformación de la mano, pueden deprimirse y perder la motivación para seguir adelante [1].

De acuerdo con la directora del Centro Integral de Terapia del Arte [10], mediante diversos elementos de arte, como pueden ser imágenes o dibujos, se logra de manera terapéutica una expresión que permite la exploración y solución de problemas. Los beneficios que se obtienen a través de esto son la expresión de los sentimientos, focalización de problemas, desarrollo de habilidades creativas, promoción del crecimiento, mejorar la comunicación, entre otros. Una de las metas que buscan lograr es que el individuo aprenda a vivir en forma creativa, aplicando la creatividad en todas las áreas de su vida para crear una vida plena y sana. Por lo tanto, a través de la terapia del arte, existe la posibilidad de ayudar a los niños que presentan los problemas anteriormente mencionados al brindarles la oportunidad de la expresión de manera escrita.

Es por esto que se propone el diseño y construcción de un prototipo que le permitirá a niños con deformidades congénitas en la palma de la mano trazar líneas sobre una superficie para dibujar o escribir, mediante el control de un joystick, con las siguientes características:

- Movimiento en los 3 ejes: eje x y y para dibujar y z para levantar o bajar el lápiz o marcador.
- Superficie cambiable para diferentes actividades.
- Posibilidad de ajustar la velocidad de uso para que sirva como terapia para la motricidad fina.

V. DISEÑO Y CONSTRUCCIÓN

Para el diseño e implementación física del prototipo, se tomó como referencia una mesa cartesiana XYZ, [11], para determinar los elementos y formas de estructuras.

A. *Diseño y Construcción de la Estructura*

El diseño de la estructura física se llevó a cabo en el programa de Blender, aplicación de fácil uso y de licencia gratuita para realizar diseños.

La estructura tiene las siguientes características:

- Superficie plana de tamaño carta, al ser el tamaño más accesible y común en hojas de papel.
  - Dimensiones (x, y): 22.5 cm x 27 cm
  - Área de trabajo máxima (x, y): 18.5 cm x 23 cm
- Modelo prismático hueco para garantizar la fácil colocación sobre una superficie plana, y permitir ocultar el cableado electrónico.
  - Dimensiones: Largo 27 cm, Ancho, 22.5 cm y Alto 5 cm.
  - Espesor: 3 mm
- Espacio para intercambiar la superficie plana, para diferentes actividades.
  - Espesor máximo de superficie removible: 5 mm.
  - Materiales: Papel, Vidrio y Plástico

La Fig. 5 muestra el diseño de las piezas laterales del prisma, mientras que la Fig. 6 muestra los soportes de los ejes y los soportes de los motores.

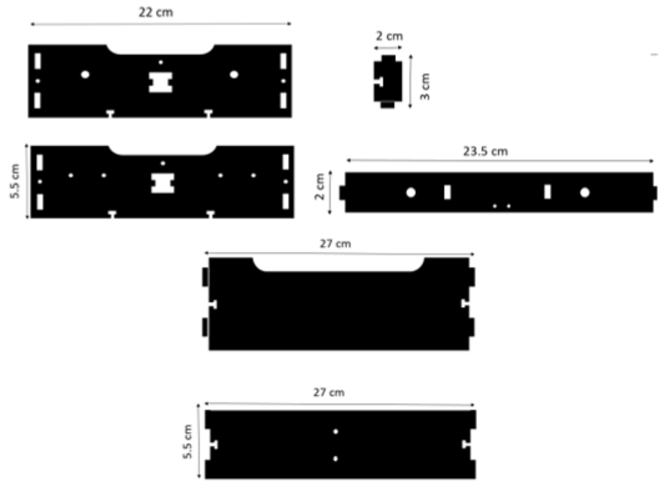


Fig. 5.- Muestras los laterales del dispositivo, soportes para los ejes y piezas que permiten el movimiento en los mismos.

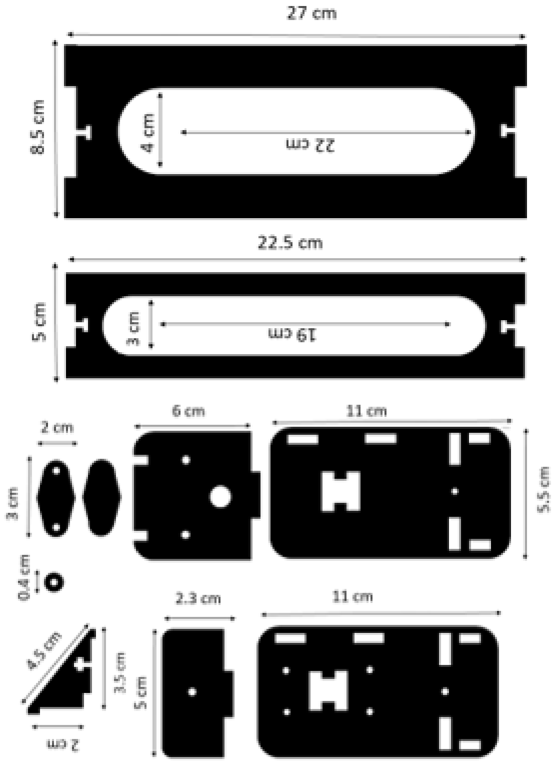


Fig. 6.- Soporte del eje X y eje Y, así como soportes para el motor.

Una vez terminado el diseño de las piezas que componen el prototipo se comenzó con el armado del mismo. Se comenzó por la base, colocando las cuatro piezas correspondientes a las misma, así como la pieza que ayuda a dar soporte para evitar que el prototipo se flexione, a continuación, se colocaron las guías para el eje X, se armó la pieza que permite el movimiento en dicho eje y sobre ella fueron colocadas tanto las piezas como las guías correspondientes al eje Y, todo lo anterior fue introducido entre las guías para el eje X; posteriormente, se ensamblo el soporte para el lápiz, así como el mecanismo que permite elevarlo en caso de que sea necesario, finalmente fueron colocados los tensores tanto para el eje X como para el eje Y, Fig. 7.

Para el movimiento en el plano de trabajo del prototipo se cuenta con los siguientes elementos:

- Varillas lisas: son el elemento que guía las piezas móviles de nuestro prototipo. Se emplean dos varillas de acero inoxidable de 8mm de diámetro por eje de movimiento horizontal (x) y vertical (y), para dar mayor soporte a la estructura.

- Correas: En los casos de los ejes x y y, se emplean correas de goma para transmitir el movimiento del motor al elemento móvil.

- Manivela: Para el movimiento z, comúnmente se emplean husillos para garantizar un movimiento controlado a lo largo del desplazamiento, sin embargo, en este prototipo se emplea un soporte tipo manivela directamente sobre un servomotor que garantiza las dos posiciones deseadas, subir y bajar el elemento trazador.

B. *Conexión Electrónica*

Considerando componentes de bajo costo y buen desempeño, pensando en largas jornadas de trabajo, se definieron los siguientes elementos:

- Motores paso a paso 28BYJ-48: voltaje de 5v; consumo de corriente de 55 mAh.
- Driver ULN2003: voltaje de operación de 5V; capacidad de corriente de 500mA; cuenta con 4 leds indicadores de las fases del motor paso a paso.
- Servomotores SG90: velocidad de operación de 0.12 segundos/60º sin carga; voltaje de 3.5V a 6V; par de torsión de 1.6 kg*cm a 4.8V.
- Joystick KY-023: voltaje de 2.7V a 5.5V; 2 potenciómetros de 10KΩ; pulsador central normalmente abierto.
- Arduino Uno: microcontrolador ATmega328P; voltaje de operación de 5V; corriente por pin de 40mA.
- Alimentación: Fuente de poder de 12 v.

Los motores 28BYJ-48 fueron elegidos debido a su bajo consumo de corriente, lo cual evita poner en riesgo a los niños que usen el dispositivo. El servomotor SG90 se encarga de elevar el elemento trazador, la selección de este elemento se hizo en base a reducir elementos, por lo que no requiere un controlador adicional, solo se requieren dos posiciones sin un alto índice de exactitud.

El diseño del joystick KY-023 permitirá en el fututo poder implementar adaptaciones de palancas que permitan al niño tener mejor agarre, facilitándole el poder realizar los trazos de manera más cómoda.

El diagrama de conexión eléctrica se puede observar en la Fig. 8, en este se muestra el uso de 3 señales digitales para cada motor a pasos que controlan los desplazamientos en los ejes X y Y, así mismo, se muestra la lectura de dos señales analógicas provenientes del Joystick que determina la dirección del movimiento a realizar. Por último, la señal de control de la posición del elemento final se obtiene directamente de una salida digital del Joystick hacia el servomotor.

VI. MODO DE OPERACIÓN

Dentro de la programación del microcontrolador, se realizó un mapeo de los valores que lee el joystick en diferentes posiciones, esto para adecuar la señal de este y poder mandar la señal de movimiento a los motores paso a paso. De esta manera, fue posible hacer que, dependiendo de la dirección de inclinación del joystick, se activara el motor paso a paso correspondiente a ese eje, y de acuerdo con el ángulo de inclinación, controlar la velocidad angular de los motores paso a paso entre los 0 y 40 mm/s.

Fig. 7.- Armado de la estructura física.

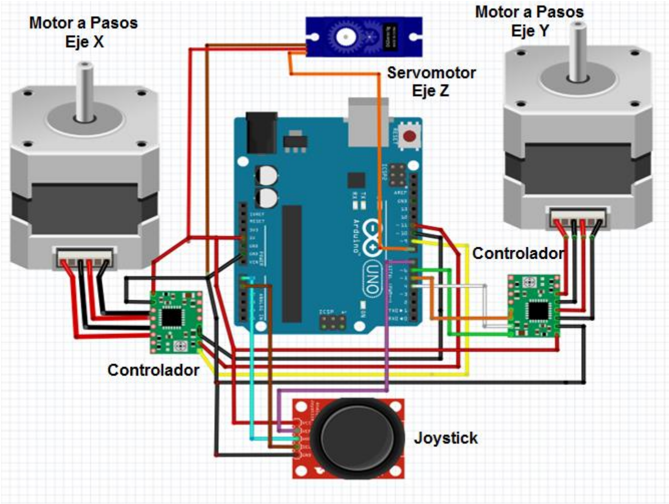


Fig. 8.- Diagrama de conexión eléctrica

Con un botón de dos estados que tiene el joystick se controla el eje z mediante el accionamiento del servomotor, el cual permite levantar o bajar 20mm el elemento trazador.

Hasta el momento, se cuenta con 2 modos de uso:

1. Seguimiento de patrones: Se emplea una superficie transparente que permite colocar un dibujo debajo de ella, y de esta manera otorgarle al usuario un patrón de líneas a seguir; gracias a esto el terapeuta podrá elaborar un registro mediante el cual se pueda observar y analizar el avance que lleva el niño en cuanto a su rehabilitación.
2. Recreativo: Consiste en colocar una hoja de papel sobre el área de trabajo para otorgarle al usuario la posibilidad de que se exprese libremente. Esto se puede utilizar como un medio de recreación mediante el cual el usuario pueda divertirse mientras adquiere la destreza necesaria para el uso de la herramienta

VII. RESULTADOS

El prototipo tiene un área de trabajo máxima de 18.5 cm x 23 cm (correspondiente a una hoja de papel tamaño carta con márgenes), lo que otorga mayor practicidad a la hora de escribir y dibujar; las velocidades de movimiento se encuentran dentro de un rango de 0 a 40mm/s, siendo establecida por el terapeuta encargado de la terapia del niño conforme vaya registrando sus avances. El prototipo final se muestra en la Fig. 9 y Fig. 10.

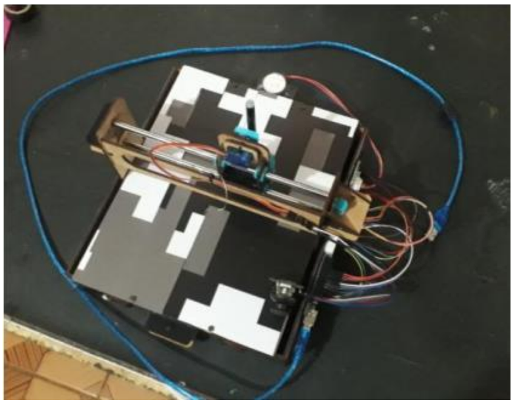


Fig. 9.- Prototipo Final de la herramienta Motick Kids.

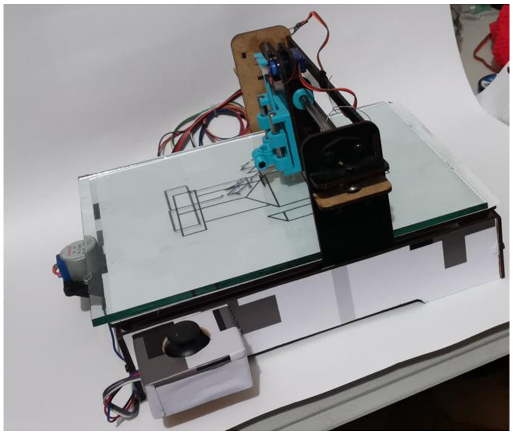


Fig. 10.- Superficie de cristal para seguimiento de patrones.

En la aplicación, el dispositivo fue capaz de ser manipulado mediante el joystick, estableciendo una interfaz máquina-usuario que permitía a los niños manipular la máquina x, y, z, y lograr los trazos que ellos deseaban.

PRUEBAS

Al estar en fase de prototipo, no se cuenta con un protocolo de pruebas fundamentado, sin embargo, al emplear el prototipo como un juguete orientado a la rehabilitación y como medio de expresión de los niños, se establecieron pautas de uso, con consentimiento de los padres y con apoyo de terapeutas.

De acuerdo a la experiencia obtenida mediante la realización del prototipo, así como el trabajo realizado con los terapeutas que atienden a los niños se llegó a la conclusión de no tener interacción directa con los niños para las pruebas, debido a que suelen ser muy sensibles con su espacio personal, por lo que se instruye al terapeuta sobre el modo de operación del dispositivo para que de esta manera él pueda instruir al niño para que comience a utilizar el dispositivo.

El prototipo de herramienta fue empleado con 8 niños de un Centro de Rehabilitación en la ciudad de Morelia, Michoacán, cuyas edades oscilaron entre los 4 y 6 años, durante un periodo de 30 días de uso a lo largo de 10 semanas, 3 sesiones por semana. En cada sesión de 30 minutos, se realizan 3 rondas de actividades de 30 minutos cada una, la primera actividad es replicar un dibujo sobre la superficie de cristal, la segunda practicar la escritura y la tercera para que el niño aplique libremente su creatividad.

Por motivos de privacidad y al no contar con un protocolo de pruebas, no nos es posible colocar nombres o fotografías de las personas y pruebas realizadas, hasta madurar el prototipo, pero la Tabla 1 muestra un resumen de los resultados cualitativos obtenidos por los terapeutas en comparación con sus pacientes que no emplean la herramienta.

Los resultados cuantitativos obtenidos a lo largo de la aplicación de la herramienta son mostrados en las Tablas 2 y 3, inherentes al tiempo de los trazos y a la destreza desarrollada por 8 niños que participaron en la prueba.

La Tabla 2 muestra los resultados obtenidos después de 15 sesiones de uso de la herramienta, se toma esta información, cuando los niños ya se familiarizaron con la herramienta y se encuentran con una buena motivación para seguirla empleando. Como se puede ver, la realización de la letra A en general es rápida, mientras que el trazo de un círculo disminuye en velocidad al tratar de hacerlo lo más preciso posible. La Tabla 3 muestra el porcentaje de niños que puede realizar los trazos indicados: una letra, seguir unos trazos específicos o escribir su nombre, a lo largo de las 30 sesiones de prueba.

Tabla 1.- Datos cualitativos recogidos por los terapeutas

Día	Motivación	Felicidad	Destreza
1	Regular	Sin cambios	Mala
3	Interés	Baja	Mala
5	Buena	Buena	Leve mejora
7	Buena	Buena	Leve mejora
14	Muy buena	Muy Buena	Regular
21	Excelente	Excelente	Buena
30	Excelente	Excelente	Muy buena

Tabla 2.- Tiempos de realización de trazos específicos.

Niño	Tiempo requerido para realizar la letra A	Tiempo requerido para realizar un cuadrado	Tiempo requerido para realizar un circulo
1	4 segundos	5 segundos	8 segundos
2	7 segundos	7 segundos	10 segundos
3	5 segundos	8 segundos	9 segundos
4	6 segundos	7 segundos	13 segundos
5	5 segundos	5 segundos	8 segundos
5	4 segundos	6 segundos	10 segundos
7	8 segundos	10 segundos	15 segundos
8	5 segundos	5 segundos	9 segundos

Tabla 3.- Porcentajes de avance en la realización de trazos

Día	Letras/Fig.	Seguir trazos	Escribir nombres
1	0%	12.5%	0%
3	12.5%	25%	0%
5	25%	50%	12.5%
7	37.5%	75%	12.5%
14	50%	87.5%	50%
21	75%	100%	75%
30	87.5%	100%	87.5%

De manera cualitativa se pudo observar mejora en el estado anímico de los niños con el uso de Motick Kids, después de su proceso de terapia, usaban el prototipo para realizar sus trazos y mostrárselos a su familia.

Los reportes del terapeuta mostraban cómo la actitud ante el proceso terapéutico había mejorado considerablemente. También se reportó mejoría considerable en su estado de ánimo y en el empeño que ponen en su tratamiento, mostrándose más sociables y con ganas de compartir el resultado de sus dibujos.

VIII. CONCLUSIONES Y TRABAJO FUTURO

La operación del prototipo fue exitosa al ser probado dentro de un Centro de Rehabilitación por un terapeuta que atiende a niños con malformaciones congénitas en manos, así mismo, se entrega un prototipo que cumple con las características de funcionamiento establecidas: movimiento fluido en el plano y un movimiento para levantar o bajar el lápiz o marcador, permisibilidad para cambio de superficie de hasta 5mm de espesor.

Así mismo, el prototipo apoya directamente a la rehabilitación para la motricidad fina, empleando la expresión escrita y gráfica, y de acuerdo a la experiencia de la prueba incrementa el estado de ánimo y motivación de los niños dentro de la rehabilitación.

Se logra con éxito, la implementación de la misma herramienta sin importar la etapa de crecimiento en la que se encuentre el niño.

Como trabajo futuro se plantea mejorar el mecanismo de movimiento de los ejes X y Y para permitir un desplazamiento más suave y con un margen más amplio de velocidades, para permitir una mayor destreza.

El terapeuta nos realizó diversas observaciones, entre ellas nos indicó que es adecuada la baja velocidad al comenzar a utilizar el dispositivo, esto para que el usuario pueda adaptarse más rápidamente a su uso, así mismo se recomendó dejar el joystick libre, no atado a la estructura, para que se pudiera mover. También nos recomendó crear diferentes adaptaciones en el joystick para que se pudiese acoplar a una mayor cantidad de usuarios.

Así mismo, se plantea la necesidad de integrar un medidor de fuerza que permita servir como instrumento de evaluación de la rehabilitación y como parámetro de medición de la habilidad adquirida

REFERENCIAS

[1] American Academy of Pediatrics, «healtychildren.org,» 5 Septiembre 2017. [En línea]. Available: https://www.healthychildren.org/Spanish/health-issues/conditions/Cleft-Craniofacial/Paginas/children-with-congenital-hand-anomalies-malformations.aspx.

[2] A. Pérez-González, M. R. Pérez-Docal y M. González-Martínez, «Experiencia con polidactilia preaxial en el Instituto Nacional de Pediatría,» Boletín Médico del Hospital Infantil de México, nº 65, pp. 27-31, Enero-Febrero 2008.

[3] F. Castellano-Fernández, S. Moraleda-Pérez y M. M. Moreno, «Resultados funcionales de la simbraquidactilia tratada quirúrgicamente,» Rehabilitación, vol. 44, pp. 177-179, 2010.

[4] National Institute of Neurological Disorders and Stroke, «nih.gov,» 21 Diciembre 2016. [En línea]. Available: https://espanol.ninds.nih.gov/trastornos/sindrome_de_Asperger.htm#AspergerSyndrome.

[5] Roman Zahariev, Nina Valchkova, Georgi Angelov, Yasen Paunski y Aleksander Krastev. 2019. Robots de Ayuda en Pedagogía y Rehabilitación: Sistemas Ciberfísicos para la Rehabilitación Pedagógica. En Actas de la 20ª Conferencia Internacional sobre Sistemas y Tecnologías Computacionales (CompSysTech '19). Association for Computing Machinery, Nueva York, NY, EE. UU., 1–7. DOI: https: //doi.org/10.1145/3345252.3345253

[6] Roman Zahariev, Nina Valchkova, and Hiroaki Wagatsuma. 2020. Service Robots for Special Education of Children with Disabilities: Robotized Systems for Social Applications. In Proceedings of the 21st International Conference on Computer Systems and Technologies '20 (CompSysTech '20). Association for Computing Machinery, New York, NY, USA, 300–306. DOI:https://doi.org/10.1145/3407982.3408023

[7] O. Portillo, C. A. Avizzano, M. Raspolli and M. Bergamasco, "Haptic desktop for assisted handwriting and drawing," ROMAN 2005. IEEE International Workshop on Robot and Human Interactive Communication, 2005., 2005, pp. 512-517, doi: 10.1109/ROMAN.2005.1513831.

[8] Esparza García, J. (2019). Diseño de material de apoyo para facilitar la autonomía de personas con malformaciones hipoplásicas de la mano APAU-MAHIMA. Universidad Pública de Navarro. [En línea] https://academica-e.unavarra.es/xmlui/bitstream/handle/2454/33824/TFE_Jerusalen_Esparza.pdf?sequence=1&isAllowed=y

[9] Landi Cochancela, S. P. (2017). Estrategías metodológicas lúdicas para mejorar la motricidad fina en niños y niñas de 3 a 4 años del centro de educación inicial Antonio Borrero Vega en el nivel inicial I B año lectivo 2016-2017. Universidad Politécnica Salesiana. https://dspace.ups.edu.ec/bitstream/123456789/14583/1/UPS-CT007164.pdf

[10] G. Nosovsky, «Centro Integral de Terapia del Arte,» 08 Junio 2017. [En línea]. Available: https://terapia-de-arte.com/ventajas-y-desventajas-de-la-terapia-de-arte/

[11] J. A. Ramírez M., J. C. Correa, y S. Flórez, «Diseño de la plataforma mecánica de una mesa cartesiana xyz automatizada», Vis. Electron., vol. 7, n.º 2, pp. 145–154, dic. 2013.