Artículo de Investigación. Revista Ciencia e Ingeniería. Vol. 43, No. 3 pp. 239-244, agosto-noviembre, 2022.
ISSN 1316-7081. ISSN Elect. 2244-8780 Universidad de los Andes (ULA)

Control de modo deslizante aplicado a un robot scara paralelo

Sliding mode control applied to a parallel scara robot

Morales, Christian[1]; Chávez, Danilo*[1]; Moya, Viviana[1]; Camacho, Oscar[2]
[1]Departamento de Automatización y Control Industrial, Escuela Politécnica Nacional, Quito-Ecuador.
[2]Colegio de Ciencias e Ingenierías. Universidad San Francisco de Quito, Quito-Ecuador.
*danilo.chavez@epn.edu.ec

DOI: https://doi.org/10.53766/CEI/2022.43.03.01

Abstract

The parallel Scara robot is one of the most basic structures of parallel robotics; it is a planar parallel robot with two degrees of freedom corresponding to the translation of the final effector in a plane. This article develops a position controller by SMC (Sliding Mode Control) for a parallel Scara; through a simulation developed in Unity 3D, the robot's performance results are obtained when working in an industrial environment filling boxes with spheres.

Keywords: Parallel Scara, SMC, Kinematic Model

Resumen

El robot paralelo Scara es una de las estructuras más básicas de la robótica paralela; es un robot plano paralelo con dos grados de libertad correspondientes a la traslación del efector final en un plano. Este artículo desarrolla un controlador de posición por SMC (Sliding Mode Control) para un Scara paralelo; a través de una simulación desarrollada en Unity 3D se obtienen los resultados de desempeño del robot al trabajar en un ambiente industrial llenando cajas con esferas.

Palabras clave: Scara paralelo, SMC, modelo cinemático

1 Introduction

The first step in designing a control system is obtaining a mathematical model of a physical system. In some cases, the model may be non-linear, distributed parameters, and/or in addition to high order. A high-order and complex model is not altogether useful; it complicates the design process and results in high-order controllers. For these reasons, it is sought to obtain a model as simple as possible, but at the same time reflect all the intrinsic characteristics of the physical system that are important for the problem being treated and leave the complete model for verification of the obtained controller. This model will incur a modeling error. Thus, the question is if the controller designed for the model obtained will work satisfactorily for the actual system. To answer this question, a control theory known as robust control has been developed since 1980 (Inthamoussou, 2011).

The parallel Scara robot is one of the most basic structures of parallel robotics. A common problem for this parallel robot is to work under uncertainty in the system parameters, either because there are errors in the modeling or because these parameters tend to change over time; the classical controllers cannot compensate for this type of error, and for such reasons is used robust control techniques.

SMC (Sliding Mode Control) can be an efficient tool for complex high-order control of non-linear plants operating under uncertain conditions; this explains the high level of research and publication activity in the area and the interest in practices in Sliding mode control over the last two decades.

2 Parallel Scara Robot

The parallel Scara is commonly used for pick and place action, composed of five bars joined together, forming a closed kinematic chain. The bar is attached to another at its end by joints; one of these bars is static and is considered the base of the robot, the two bars that are attached directly to the base must be attached to a motor; this allows the control of the movement of the robot, the end effector is located in the common joint between the two following bars (Campo et al., 2010).

2.1 Kinematic Model

Figure 1, shows the representation of a parallel Scara robot, where $\boldsymbol{l_1}, \boldsymbol{l_2}, \boldsymbol{l_3}, \boldsymbol{l_4}$ and $\boldsymbol{l_5}$ are the lengths of the links $\overline{\mathbf{AB}}$, $\overline{\mathbf{BC}}$, $\overline{\mathbf{CD}}$, $\overline{\mathbf{DE}}$ and $\overline{\mathbf{EA}}$ respectively, $\boldsymbol{\theta_1}, \boldsymbol{\theta_2}, \boldsymbol{\theta_3}$ and $\boldsymbol{\theta_4}$ are the rotation angles of each joint concerning the X axis the coordinate system origin of the robot is at point A, and the end effector is located at the point C(Campo et al., 2010).

The kinematic model of the parallel Scara robot is defined as follows:

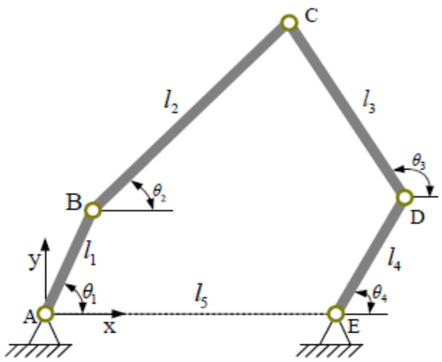


Fig. 1. Parallel Scara robot scheme

For the kinematic chain $\overline{ABC}$:

$$X = l_1 * cos(\theta_1) + l_2 * cos(\theta_2) \quad (1)$$

$$Y = l_1 * sin(\theta_1) + l_2 * sin(\theta_2) \quad (2)$$

For the kinematic chain $\overline{AEDC}$:

$$X = l_4 * cos(\theta_4) + l_3 * cos(\theta_3) + l_5 \quad (3)$$

$$Y = l_4 * sin(\theta_4) + l_4 * sin(\theta_4) \quad (4)$$

The Jacobian matrix is applied to the system to express the system in terms of inputs and outputs (Guamán y Lozada, 2016).

$$\dot{P} = J * \dot{\theta} \quad (5)$$

Where, $P = [x \quad y]^T$ is the position vector of the end effector, $\dot{P} = [\dot{x} \quad \dot{y}]^T$ is the linear velocity vector of the end effector, $\theta = [\theta_1 \quad \theta_2]^T$ is the joint angle vector and $\dot{\theta} = [\dot{\theta}_1 \quad \dot{\theta}_2]^T$ is joint velocity vector of the system. J is the Jacobian matrix defined by:

$$J = \frac{\dot{P}}{\dot{\theta}} = \frac{\partial P}{\partial \theta} = \begin{bmatrix} \frac{\partial X}{\partial \theta_1} & \frac{\partial X}{\partial \theta_2} \\ \frac{\partial Y}{\partial \theta_1} & \frac{\partial Y}{\partial \theta_2} \end{bmatrix} \quad (6)$$

Applying the Jacobian matrix to the system gives:

For the kinematic chain $\overline{ABC}$:

$$\begin{bmatrix} \dot{X}r \\ \dot{Y}r \end{bmatrix} = \begin{bmatrix} -l_1 * \sin(\theta_1) & -l_2 * \sin(\theta_2) \\ l_1 * \cos(\theta_1) & l_2 * \cos(\theta_2) \end{bmatrix} * \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad (7)$$

For the kinematic chain $\overline{AEDC}$:

$$\begin{bmatrix} \dot{X}r \\ \dot{Y}r \end{bmatrix} = \begin{bmatrix} -l_4 * \sin(\theta_4) & -l_3 * \sin(\theta_3) \\ l_4 * \cos(\theta_4) & l_3 * \cos(\theta_3) \end{bmatrix} * \begin{bmatrix} \dot{\theta}_4 \\ \dot{\theta}_3 \end{bmatrix} \quad (8)$$

The inverse kinematic model is required to design the position controller, and as explained above, the objective is to find the joint variables from a specific position. The inverse kinematic model of the system is expressed as follows:

$$\dot{\theta} = J^+ * \dot{P} \quad (9)$$

Where J^+ is the Pseudoinverse matrix and is defined by:

$$J^+ = J^T * (J * J^T)^{-1} \quad (10)$$

3 Position Controller Design

The position controller to be implemented in the parallel Scara is an SMC, which is a kind of controller of variable structure that presents robustness to the uncertainty of the parameters of the system.

The sliding surface to be used for the design of the controller is a Proportional Integral Derivative surface area and is defined as follows (García y Zambrano, 2004):

$$s(t) = \left(\frac{d}{dt} + \lambda\right)^n \int e(t)\, dt \quad (11)$$

Where e is the error and is defined as the difference between the desired output and the actual output.

$$e = P_d - P = \begin{bmatrix} X_d \\ Y_d \end{bmatrix} - \begin{bmatrix} X_r \\ Y_r \end{bmatrix} \quad (12)$$

The system under study is of the first-order reason why n = 1; for this reason, the derivative part of the surface is eliminated and is defined by:

$$s(t) = e(t) + \lambda \int e(t)\, dt \quad (13)$$

The next step is to derive the surface concerning time:

$$\dot{s}(t) = \dot{e}(t) + \lambda\, e(t) \quad (14)$$

Replacing (11) in (13)

$$\dot{s}(t) = (\dot{P}_d - \dot{P}) + \lambda\, e(t) \quad (15)$$

Replacing the kinematic model in the previous equation (15).

The sliding mode control law is defined by $U(t) = v_C(t) + v_D(t)$, it is divided into a continuous part and a discontinuous part. To proceed to find the continuous part, it must be considered that the slide surface has already been reached and the dynamics of the system are staying on it; the necessary condition for it is the time derivative of the slide surface is zero $\dot{s}(t) = 0$ and that the discontinuous part of the system does not act $v_D(t) = 0$ [5].

$$\mathbf{0} = (\dot{\boldsymbol{P}}_d - \dot{\boldsymbol{P}}) + \boldsymbol{\lambda}\, \boldsymbol{e(t)} \quad (17)$$

Clearing the continuous part of the controller and considering that the control law is $U(t) = \dot{P} = v_C(t)$.

$$\boldsymbol{v_C(t)} = \boldsymbol{J^+}(\dot{\boldsymbol{P}}_d + \boldsymbol{\lambda}\, \boldsymbol{e(t)}) \quad (18)$$

The discontinuous part is defined by:

$$u_D(t) = J^+\, [k_1\, sign(s(t))] \quad (19)$$

Where k_1 is a constant of adjustment, in this part, when using the sign function, the problem of chattering in the system is produced, so the following modification is made in the discontinuous part of the controller to reduce the chattering problem.

$$u_D(t) = \; J_e^T\, (J_e\, J_e^T)^{-1} \left[k_1\, \frac{s(t)}{|s(t)|+\delta}\right] \quad (20)$$

With the constant δ is possible to obtain a smooth curve and not have switched to high frequencies. Replacing (19) and (20) in the control law equation.

$$U(t) = \dot{\theta} = J^+\left(\dot{P}_d + \lambda\, e(t) + \left[k_1\, \frac{s(t)}{|s(t)|+\delta}\right]\right) \quad (21)$$

Figure 2 shows the control scheme of the system that is implemented.

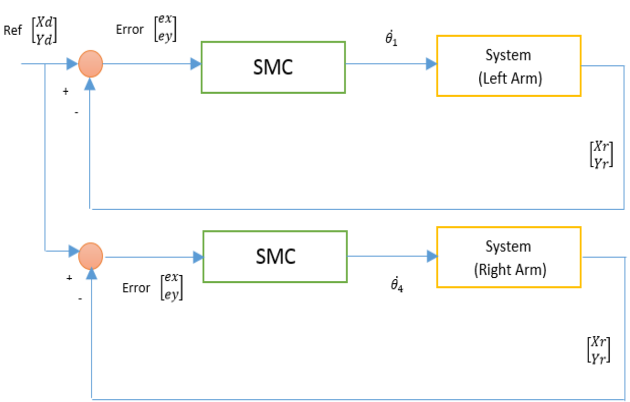


Fig. 2. Implemented scheme control

Where $[X_d\; Y_d]^T$ is the system reference or the desired position for the final effector, $[ex\; ey]^T$ is the error in "X" and "Y" of the system, $[X_r\; Y_r]^T$ is the current position of the final effector, $\dot{\theta}_1$ and $\dot{\theta}_4$ is the output of each controller, ie the speed of rotation of each motor that allows the robot to move.

As previously mentioned, a parallel robot is a closed kinematic chain formed by several independent serial kinematic chains; for this reason, it is necessary to implement two controllers, one for each motor.

To choose the value of the controller's parameters, we use the heuristic method from which the final parameters used in the parallel Scara controller are obtained.

To find the lowest ISE value, the maximum speed of rotation of the motor is 30 [rad / s], and Table 1 presents the controller parameters.

Table 1 SMC tuned parameters that present the lowest ISE

Controller	$\boldsymbol{\lambda}$	$\boldsymbol{K_i}$	$\boldsymbol{\delta}$
SMC	27	400	9

Figure 3 shows the output system when the controller has the parameters presented in Table 1.

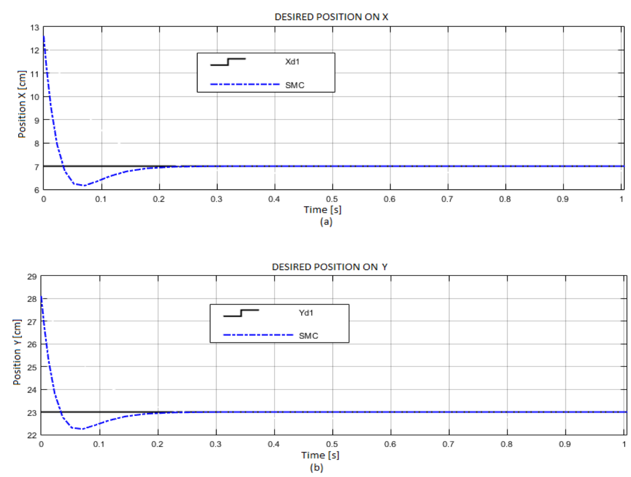


Fig. 3. System output: (a) desired position on X, (b) desired position on Y.

Figure 4 shows the control signal provided by the controller for both motors; with the first motor, the maximum speed is reached, and the settling time and the ISE value are shown in table 2.

The parameters with the lowest ISE value are not chosen directly because this implies that the angular velocities each motor must rotate are high. The system can leave the reachable working zone and take a configuration with a singularity.

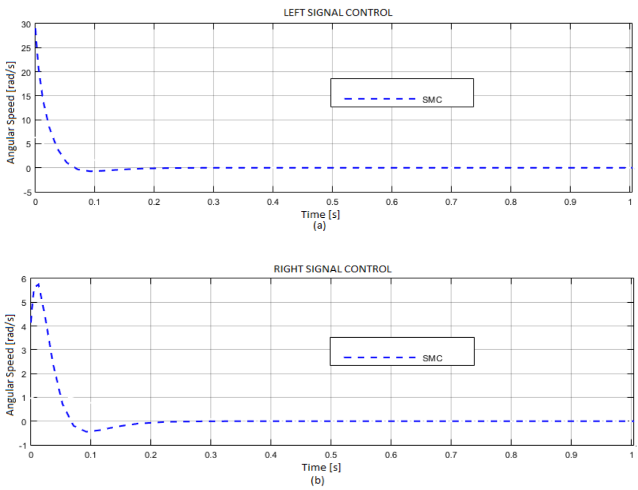


Fig. 4. Control signal provided by controllers (Speed in [rad / s] of right and left motor), (a) left system control signal, (b) right system control signal.

Table 2 Results obtained from Simulink

Controller	"X" Settling Time [s]	"Y" Settling Time [s]	"X" ISE	"Y" ISE
SMC (Left Arm)	0.226	0.226	0.274	0.213
SMC (Right Arm)	0.226	0.226	0.278	0.227

It is for this reason that the parameters of the controller must be modified by simulating the robot in 3D unity can be observed in a more real operation, where l takes into account the maximum speed of rotation of the motor, avoid adopting configurations that have singularity and the lower ISE, choosing the final adjustment parameters of the controller The lengths of the links that were used to carry out the present project are $l1 = l4 = 18[cm]$ $l2 = l3 = 16[cm]$ and $l5 = 25$. The adjustment parameters of the controller are shown in the following Table 3.

Table 3 SMC tuned parameters

Controller	$\boldsymbol{\lambda}$	$\boldsymbol{K_i}$	$\boldsymbol{\delta}$
SMC	4	12	0.4

The following Table 4 shows the settling time of the system and the ISE value.

Table 4 Results obtained from Simulink

Controller	"X" Settling time [s]	"Y" Settling Time [s]	"X" ISE	"Y" ISE
SMC (Left Arm)	1.91	1.81	2.6506	1.992
SMC (Right Arm)	1.91	1.81	2.6905	2.1392

4 Results obtained

To see the behavior of SMC by graphics, we proceed to simulate the system in Matlab Simulink. The results obtained by the simulation of the controllers are presented below.

Figure 5 shows the system's output, the position in both "x" and "y" of the end effector of the parallel Scara, and the change in time due to the controller's action. It can be seen that the controller by sliding modes presents overlap; however, this parameter present in the controller's behavior is not very relevant for the development of this project and can be reduced by modifying the controller constants if necessary. Still, as explained above, the tuning of both controllers was done taking into account the ISE criterion and through "trial and error" in the simulation environment developed in Unity 3D to find parameters that enable the industrial process to be carried out more quickly.

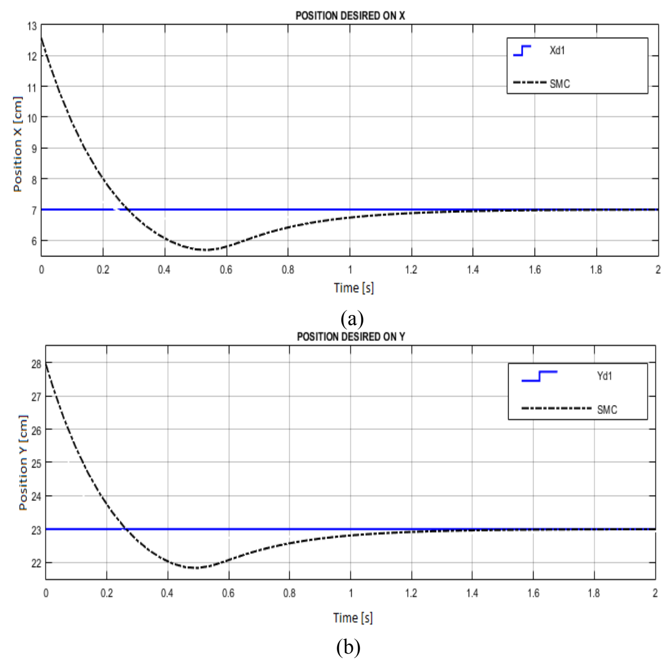
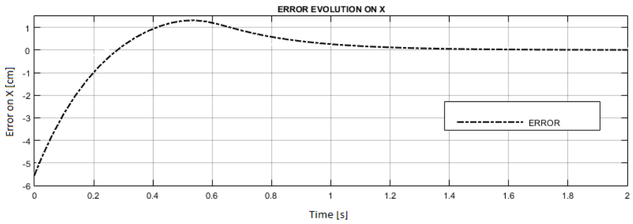


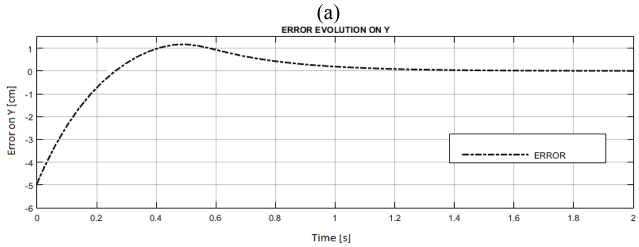
Fig. 5. Control signal provided by controllers (Speed in [rad / s] of right and left motor), (a) left system control signal, (b) right system control signal.

Figure 6 shows the evolution of the error for each serial kinematic chain that composes the parallel Scara; as we can see, the controllers achieve a steady-state error value equal to zero (e = 0), so it can be said that the controllers manage to complete the task satisfactorily.

As seen in Table 2, the time taken by the controller implemented to the right and left kinematic chain to reach the zero-error value is practically the same because both joined chains form a closed chain and reach the same point simultaneously.



(a)



(b)

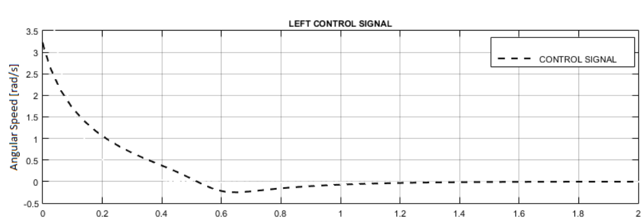
Fig. 6. Error evolution from X (a) and Y (b).

Figure 7 shows the control signal provided by the controller and the speed at which each motor of the parallel Scara must rotate. The rotation speed can be increased by modifying the controller's parameters. However, it should be prudent because if the rotation speed is very high, it can cause the robot to fall into some configuration that presents singularities.

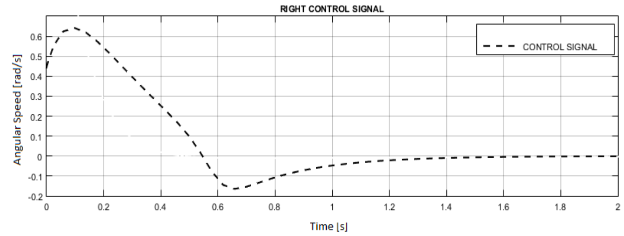
Figure 8 shows the simulation environment developed in Unity 3D in which the controller will be implemented to see the performance when the robot works in an industrial environment, filling boxes.

The industrial environment consists of filling different boxes; in this case, four boxes have holes for the spheres in different positions. Once the current one is filled, it moves to one side. At the same time, a new box is also moved to the robot's reachable workspace to fulfill the filling process again, thus giving the effect of a production line, where everything in the industrial process consists of continuously filling boxes.

Figure 9 shows the parallel Scara filling one kind of box implemented in the simulation. The time it takes for the robot to fill each box accumulates and is displayed in the window "console" whenever there is a change of box; the time that exists during the change of boxes is not taken into account due to It does not depend on the controller implemented in the robot.



(a)



(b)

Fig. 7. Error evolution from X (a) and Y (b). Control signal provided by controllers (Speed in [rad / s] of right and left motor), (a) left system control signal, (b) right system control signal.

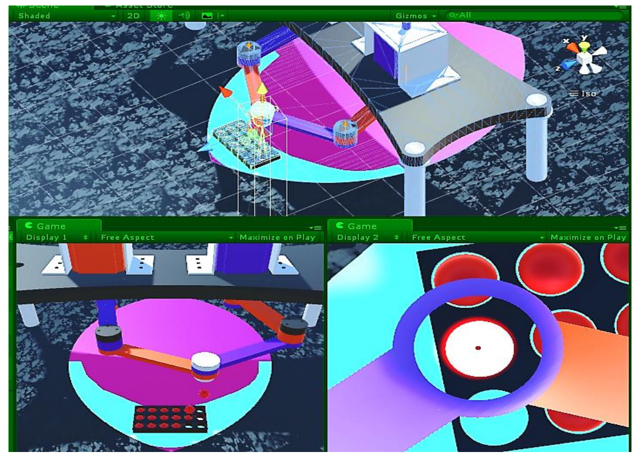


Fig. 8. Parallel Scara robot simulation environment.

Figure 9 shows the parallel Scara filling one kind of box implemented in the simulation. The time it takes for the robot to fill each box accumulates and is displayed in the window "console" whenever there is a change of box; the time that exists during the change of boxes is not taken into account due to It does not depend on the controller implemented in the robot.

Figure 10 shows a different kind of boxes used in the industrial process, where it can be seen the holes have different positions in each box.

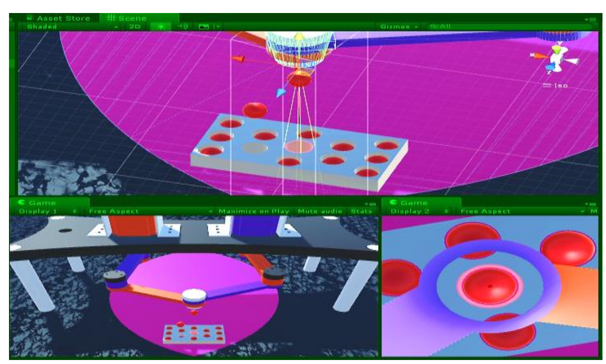


Fig. 9. Parallel Scara robot is filling the white box.

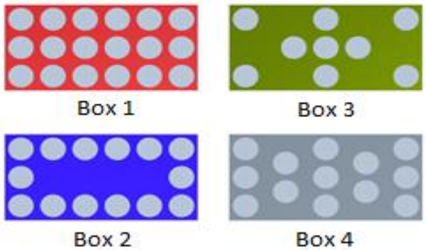


Fig. 10. Kind of boxes implemented on unity.

Table 5 shows the time taken by the controller to comply with the entire industrial process, that is, to fill the four boxes.

Table 5 Time to fill the boxes by the parallel Scara

Box	Time [s]
1	11.25
2	8.41
3	7.15
4	8.19

5 Conclusions

The simulation developed in Simulink showed that the SMC could comply with the control objective because the error in steady-state is cero, which means the desired position is reached, and this allows to fill all the boxes complying with the entire industrial process as demonstrated by simulation in Unity 3D.

The ISE value of the controller can be reduced by the controller setting parameters modifications; however, it is recommended to perform functional tests in the system depending on the task that the robot must perform since the action of the robot may be faster, but this also increases the possibility of the robot falling into a configuration that presents singularity.

References

Inthamoussou, F. (2011). Nociones de control Robusto. Málaga: Universidad Nacional de la Plata.

Vivas, O. (2005). Robótica Paralela: Aplicaciones Industriales, Modelado y Control. Popayán: Universidad del Cauca.

Ojeda, P. (2006). Modelación y Simulación Cinemática de un Robot Delta Planar tipo RR. México: Universidad Autónoma de México.

Campos, L., Bournbonnais, F., Bonev, I. y Bigras, P. (2010). Development of a five-bar parallel robot with large workspace. Quebec: École de technologie supérieure (ÉTS).

García, W. y Zambrano, D. (2004). Control predictivo por modo deslizante para robots manipuladores, Mérida: Universidad de los Andes.

Guamán, R. y Lozada, W. (2016). Control de posición y seguimiento de trayectorias de un manipulador móvil de 3 grados de libertad. Quito: Escuela Politécnica Nacional.

Jian, S., Rolland, L. y James, L. (2014). Five bar planar manipulator simulation and analysis by bond graph, Proceedings of the ASME 2014 International Mechanical Engineering Congress and Exposition. pp. 1-7. https://www.academia.edu/59673453/Five_Bar_Planar_Manipulator_Simulation_and_Analysis_by_Bond_Graph

Zi, B., Cao, J. y Zhu, Z. (2011). Dynamic Simulation of Hybrid-driven Planar Five-bar Parallel Mechanism Based on SimMechanics and Tracking Control. China: University of Mining and Technology.

Muñoz, E. y Gaviria. C. (2011). Control PID Multivariable y Modos Deslizantes de un Robot SCARA. Cauca: Grupo en Automática Industrial - Universidad del Cauca.

Recibido: *20 de febrero de 2022*

Aceptado: *15 de mayo de 2022*

Moracles, *Christian:* *holds a Electronic and Control Engineering degree from the Escuela Politécnica Nacional (Quito – Ecuador). Correo electrónico:Christian.morales@epn.edu.ec*

Chávez, Danilo: *holds an Electronic and Control Engineering degree from the Escuela Politécnica Nacional (Quito – Ecuador) and Ph.D from the Universidad Nacional de San Juan. His research interests are: control automatic systems and human-machine interaction.*
https://orcid.org/0000-0002-7529-9006

Moya, Viviana: *holds an Electronic and Control Engineering degree from the Escuela Politécnica Nacional (Quito – Ecuador) and Ph.D from the Universidad Nacional de San Juan. Her current research is on the teleoperation of robots, delayed systems, and control automatic systems. Correo electrónico: viviana.moya@epn.edu.ec*
https://orcid.org/0000-0002-6064-6925

Camacho, Oscar: *holds an Electic Engineering degree from the Universidad de Los Andes (Mérida-Venezuela). and Ph.D from Universidad del Sur de Florida, Tampa (USF). His research interests are: control automatic systems, robust control and smc. Correo electrónico: ocamacho@usfq.edu.ec*
https://orcid.org/0000-0001-8827-5938