Collaborative Control of Mobile Manipulator Robots through the *Hardware in the Loop Technique*

Luis F. Santo, Richard M. Tandalla and
Víctor H. Andaluz

Universidad de las Fuerzas Armadas ESPE, Sangolquí, Ecuador
`{lfsanto, rmtandalla, vhandaluz1}@espe.edu.ec`

Abstract. This article aims at designing and implementing the "Hardware in the Loop" (HIL) technique, to evaluate the collaborative control algorithm of two mobile manipulator robots to carry out tasks of movement and manipulation of objects in an industrial environment. The developed control structure is made up of a centralized control algorithm, developed in Matlab mathematical software, which is linked to the HIL system, which contains both the kinematic model and the dynamic model of each robotic system programmed on the Raspberry Pi. To analyze the optimal functioning of the proposed control algorithm, an immersive virtual reality scenario is designed and implemented using the UNITY3D graphic engine, which facilitates interaction with the user.

Keywords: Collaborative Control, Hardware in the Loop, Mobile Manipulator, Virtual Environment

1 Introduction

The industry 4.0 has opened several forms of automation that have as common objective the improvement of the productivity and the optimization of the work processes, reason why, many companies already have different collaborative robots that are designed to share with people with high degree of security in a work atmosphere [1], [2].

Currently, mobile manipulator robots refer to robots that consist of a manipulator arm on a moving platform [3], [4]. the combination of these robotic systems has several advantages because it complements the capacity of a fixed base manipulator arm with the freedom of movement offered by the mobile platform with wheels [5]. Mobile manipulators allow for complex tasks that require both locomotion and handling capabilities [6], [7]. The mobile manipulator robots have multiple applications in different areas of the industry, for example, in construction companies, in mining, as assistance to people, among others [8]. In addition, a mobile manipulator robot to adapt to an industrial environment, must meet several features such as having the ability to work with people without any risk, be autonomous, easy to configure and install and work in compliance with the requirements of the industry [9].

Systems generally made up of two or more mobile manipulator robots that fulfill a common objective are called collaborative robots [10] which allows for multi-tasking operations [11], allowing standard controllers to cooperate with each other, to perform complex tasks that cannot be performed by a single robot [12], The control schemes of the collaborative robots are mainly based on: *i) centralized architecture* in which the central computer generates the control actions to achieve the secondary projections [13]; *ii) decentralized architecture,* in which all components of the robotic system consist of the proprietary processing unit which develops both kinematic and dynamic control [13].

The implementation of control algorithms for mobile manipulator robots in the development of collaborative tasks presents a high complexity, because the robots are not always physically available due to the high cost of acquiring each one [14]. Therefore, we suggest the implementation of didactic modules that use the HIL technique [15] as a low cost alternative for the implementation of the proposed algorithms, in which it will be possible to simulate a control environment for the robot, which will analyze the stability of the control algorithm implemented [16], to validate trends in control errors. Avoiding the cost, risk and time associated with physical testing today. [17]

This article aims to develop a system that uses the technique Hardware in the Loop, for collaborative control between two mobile manipulator robots, for which it is considered a 3D virtual environment where the displacement of the robots is simulated. To do this, from Matlab mathematical software control actions are sent through a wireless channel to Raspberry Pi modules, which are programmed mathematical models for each robot, to evaluate the control algorithms and present them in the 3D virtual interface.

This document is developed in 7 sections. In section two it presents the structure of the implemented HIL system. Section three shows the development of the virtual environment. The section four presents the modeling both kinematic and dynamic of the Mobile Manipulator Robot. Section five describes the Collaborative Control algorithm. Section six includes results obtained. Finally, section 7 contains the conclusions of the implemented system.

2 System Structure

The HIL technique is a simulation of the system or process required in real time, in which real signals from the controller are connected to a test system that uses a computer as a virtual representation of the plant model, with the objective of validating the controller without the cost and time associated with current physical testing.

For the proposed system, its structure consists of the implementation of the "Hardware in the Loop" (HIL) technique, for the collaborative control of two mobile manipulator robots to execute tasks of moving and manipulating a common object. Fig. 1 describes the structure of the system based on HIL.

The proposed structure consists of three main stages. In the first stage, the collaborative control algorithm consists of three layers: *a) The Offline Planning layer*. is in charge of determining the initial conditions for the system and generating the required path of the common object to be manipulated by the robots, *b) The layer of Planning Online.* aims to restore the references at any time, so that the layer of formation reacts appropriately to the environment, *c) The layer of control formation.* It is in charge to generate the signals of control that are sent to each robot manipulator mobile that works as a team to realize the established trajectory by the planning layers.

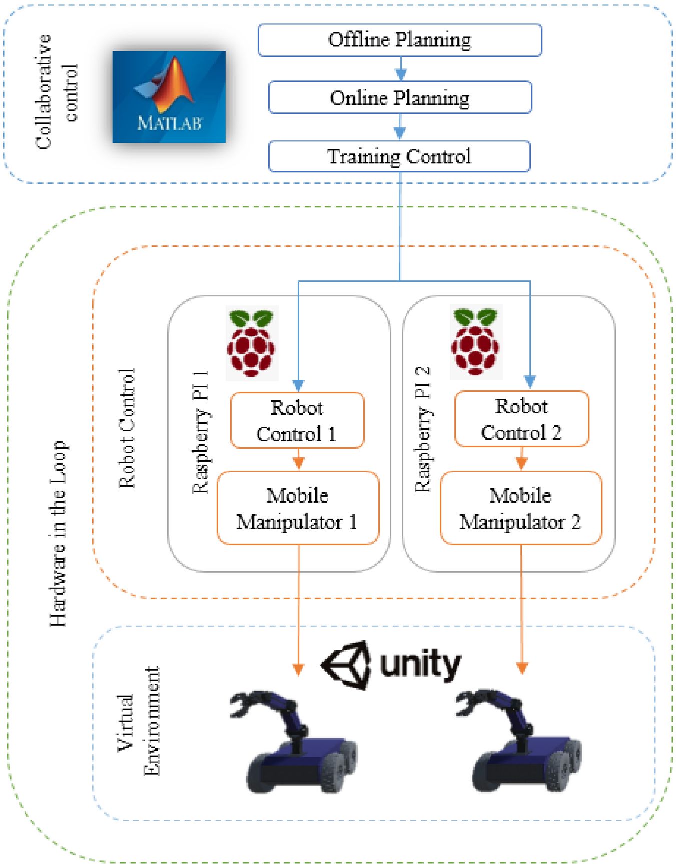

Fig. 1. System Structure.

The signals of the collaborative control algorithm for the robots are calculated by the mathematical software Matlab, which are transmitted via a wireless communication channel to each mobile manipulator robot of the HIL system.

The second stage presents the *Robot Control*, which has two mobile manipulator robots, each consists of a 3DOF robotic arm on a mobile platform type unicycle, whose mathematical models, kinematics and dynamics, are integrated into a Raspberry PI to simulate the dynamic behavior of the mobile manipulator robot in real-time, without the need for the physical mechanism.

In stage three, in the Unity3D graphic engine is designed and implemented a *virtual environment* for immersion and interaction of the system in real time, which contains the structure of the mobile manipulator robots, designed by means of CAD software considering their respective mechanical characteristics, which emulate the movement of the robots. The communication between Matlab mathematical software and Unity 3D virtual environment is based on the communication protocol DLL (dynamic link library), through shared memories.

3 Virtual Environment

Fig. 2 presents the proposed scheme for the implementation of the virtual environment designed in the UNITY3D software. The aim of the virtual system is to simulate the environment; the programming of SCRIPTS that allow the control of each object present in the environment; external complements that allow to improve the animation of the scenes.

The simulated system incorporates 3D objects that allow the creation of a virtual reality environment. This environment is developed considering all the real physical properties, taking into account that it is an industrial environment. The environment contains two mobile manipulator robots, each consisting of a 3DOF robotic arm on a mobile platform, designed to execute collaborative tasks that require the ability to manipulate and move objects. The virtual environment presents each mobile manipulator robot, developed in a CAD design software, considering the mechanical and physical characteristics of the whole robotic system. In the same way, the 3D design of the scenario where the animation is developed is presented; in this case, the environment is an industrial environment (a construction), where the robots will execute tasks of manipulation and displacement of the mechanical pieces (beams, rods, etc.) belonging to the environment.

In the execution of the system, the virtual environment is linked to input devices such as the Oculus, which uses the device library (SDK), which allows visual and auditory user interaction during the execution of the established task. It also links up with Matlab mathematical software through the use of shared memories (SM) based on dynamic link libraries (DLLs), which allow information to be shared between programs. The use of shared memories allows the control actions, both position and rotation, to be entered into the robotic system.

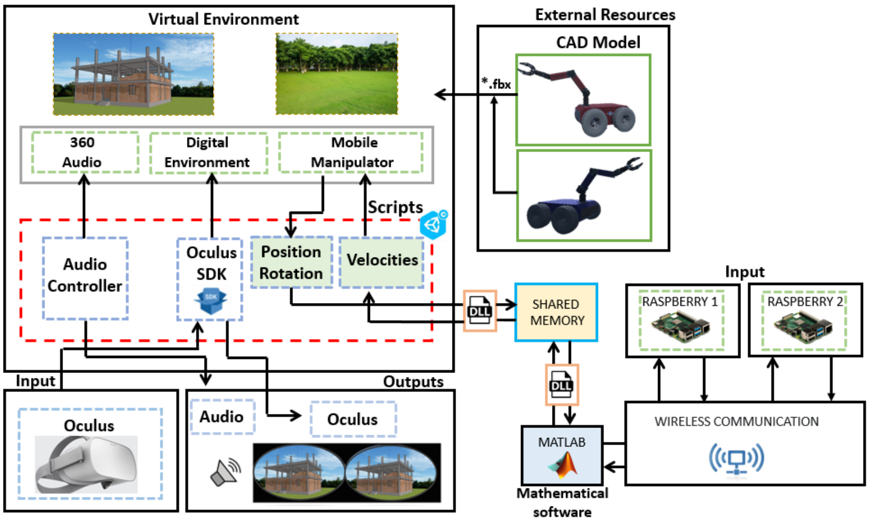

Fig. 2. Structure of the virtual environment

4 Mobile Manipulator Robots

The vector $\mathbf{q}$ of n coordinates, define the mobile manipulator robot, so that $\mathbf{q} = [q_1 \quad q_2 \quad q_3 \quad \dots \quad q_n]$ where, $\mathbf{q}_a$ and $\mathbf{q}_p$ represent the standardized coordinates of both the manipulator and the mobile platform respectively. It is obtained $n = n_a + n_p$, where n_a refers to the magnitudes of the standardized spaces of the manipulator, in the same way n_p of the mobile platform.

The vector $\mathbf{q}$ expresses the configuration of the working area of the robot; expressed as N. The ubication for the point of interest of the robot refers to the vector m-dimensional $\mathbf{h} = [h_1 \quad h_2 \quad h_3 \quad \dots \quad h_m]$ which represents both the position and the orientation of the point of interest corresponding to the robot in the plane $\Re$. the grouping of all locations forms the operating area of the robot, expressed as M.

4.1 *Kinematic Model*

It presents the ubication of the point of interest (operating end) $\mathbf{h}(t)$ depending on both the position of the platform as well as the configuration of the manipulator [18].

$$f : \mathrm{N}_a \quad x \quad \mathrm{M}_p \rightarrow \mathrm{M}$$

so that

$$(\mathbf{q}_a \quad , \quad \mathbf{q}_p) \Rightarrow \mathbf{h} = f(\mathbf{q}_a \quad , \quad \mathbf{q}_p)$$

in which, N_a corresponds to the operating area of the manipulator, in the same way M_p of the mobile platform.

By means of the derivative of the position of the point of interest taking into account those derived from the location of the platform and the configuration of the manipulator, the instantaneous kinematic model of the robotic system is obtained [18],

$$\dot{\mathbf{h}}(t) = \frac{\partial f}{\partial \mathbf{q}}(\mathbf{q}_a, \mathbf{q}_p)\mathbf{v}(t)$$

in which, $\dot{\mathbf{h}} = [\dot{h}_1 \quad \dot{h}_2 \quad \dot{h}_3 \quad \dots \quad \dot{h}_m]$ expresses the velocity at the point of interest, $\mathbf{v} = [v_1 \quad v_2 \quad v_3 \quad \dots \quad v_m] = [v_p \quad v_a]$ expresses the mobility control of the robotic system. The dimension is denoted by $\Delta_n = \Delta_{np} + \Delta_{na}$, in which Δ_{np} expresses the magnitudes of the mobility control corresponding to the platform and Δ_{na} to the magnitudes of the manipulator. Following the analysis of the above statements, it is possible to express the velocity at the operational end, as:

$$\dot{\mathbf{h}}(t) = \mathbf{\Gamma}(\mathbf{q})\mathbf{v}(t) \qquad (1)$$

in which, $\mathbf{\Gamma}(\mathbf{q})$ defines the Jacobian matrix which represents the linear mapping of $\mathbf{v}(t)$ and $\dot{\mathbf{h}}(t)$, which corresponds to the velocity vector of the robotic system and the operating end respectively.

4.2 *Dynamic Model*

The mathematical model that refers to the dynamics of the robotic system, is obtained by means of the Euler-Lagrange technique, from the difference of the kinetic energy K, and the potential energy P [19].

$$L = K - P \qquad (2)$$

$$f_i(t) = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}_i}\right) - \frac{\partial L}{\partial \mathbf{q}_i} \qquad (3)$$

Applying the Euler-Lagrange method, the forces generated in the robot are defined

$$\mathbf{f}(t) = \mathbf{g}(\mathbf{q}) + \mathbf{C}(\mathbf{q}\ ,\ \dot{\mathbf{q}})\mathbf{v}(t) + \mathbf{M}(\mathbf{q})\dot{\mathbf{v}}(t) \qquad (4)$$

in which, $\mathbf{q}$ is the main vector of the coordinate system for the robotic system, $\mathbf{v}$ represents the velocity vector of the robot, $\mathbf{M}$ expresses the inertial matrix for the system, $\mathbf{C}$ represents the matrix of centrifugal and centripetal forces, and $\mathbf{g}(\mathbf{q})$ defines the gravity [19] [20].

The reference velocities that act as control signals of the system, considering the dynamics of the robot, is expressed in the following way:

$$\mathbf{v}_{\mathbf{ref}}(t) = \bar{\mathbf{g}}(\mathbf{q}) + \bar{\mathbf{C}}(\mathbf{q}\ ,\ \dot{\mathbf{q}})\mathbf{v}(t) + \bar{\mathbf{M}}(\mathbf{q})\dot{\mathbf{v}}(t) \qquad (5)$$

5 Collaborative Control

The collaborative control starts from the formation of two mobile manipulator robots in which the main interest is based on the center point of the distance between the manipulators having a projection of position and orientation. This projection is shown in Fig. 3, in which the two mobile manipulators $\mathbf{h}_1$ and $\mathbf{h}_2$ execute the collaborative task.

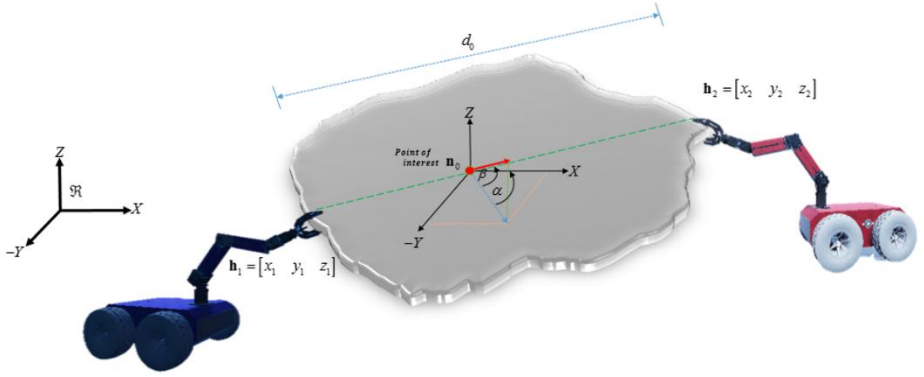

Fig. 3. Collaborative Control of Mobile Manipulators

Between the two robots h1, h2 there is the projection in which the point n0 is obtained $\boldsymbol{\eta}_0 = [\mathbf{p}_0 \quad \mathbf{s}_0]$, where you have to $\mathbf{p}_0 = [x_0 \quad y_0 \quad z_0]$, which establishes the center point of the distance that separates the robots in coordinates from the reference system $\Re$; y $\mathbf{s}_0 = [d_0 \quad \alpha_0 \quad \beta_0]$ where d_0 is in units of length and represents the distance between the manipulators; while α_0 and β_0 are angles representing the orientation formed with the axes of the reference system $\Re$.

The projection allows to differentiate the position and orientation that is desired (given by the established task) and the orientation and position of the collaborative work to be done at that time $\boldsymbol{\eta}_0$, to later generate new velocities and make corrections of the errors in the mobile manipulators, for which we have the following expressions with which the projections are made [7].

$$\mathbf{p}_0 = \tfrac{1}{2}\left[(x_2 + x_1) \quad (y_2 + y_1) \quad (z_2 + z_1) \right] \tag{6}$$

$$\mathbf{s}_0 = \begin{bmatrix} \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 + z_1)^2} \\ \tan^{-1}\left(\dfrac{z_2 - z_1}{x_2 - x_1} \right) \\ \tan^{-1}\left(\dfrac{y_2 - y_1}{x_2 - x_1} \right) \end{bmatrix} \tag{7}$$

where $[x_1 \quad y_1 \quad z_1], [x_2 \quad y_2 \quad z_2]$ refers to the position components of the reference system $\Re$, of the point of interest of the first and second mobile manipulator respectively.

Taking into account the forward and backward temporal derivation of the kinematic transformations, the relation between the temporal variations is obtained $\dot{\mathbf{h}}(t)$, $\dot{\boldsymbol{\eta}}(t)$ which are represented by $\mathbf{J}$, defined as Jacobian matrix, the same that is described by

$$\dot{\boldsymbol{\eta}}(t) = \mathbf{J}(\mathbf{h})\dot{\mathbf{h}}(t) \tag{8}$$

otherwise it is defined by

$$\dot{\mathbf{h}}(t) = \mathbf{J}^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}(t)\,. \tag{9}$$

5.1 *Formation Controller*

The collaborative control stage in Fig. 1 designates the shape parameters and the definite position parameters for the task $\boldsymbol{\eta}_d = \begin{bmatrix}\mathbf{p}_d & \mathbf{s}_d\end{bmatrix}$ and desired variations $\dot{\boldsymbol{\eta}}_d = \begin{bmatrix}\dot{\mathbf{p}}_d & \dot{\mathbf{s}}_d\end{bmatrix}$ by means of $\tilde{\boldsymbol{\eta}}_d = \boldsymbol{\eta}_d - \boldsymbol{\eta}$ the formation error is determined, deriving the equation as a function of time $\dot{\tilde{\boldsymbol{\eta}}}_d = \dot{\boldsymbol{\eta}}_d - \dot{\boldsymbol{\eta}}$. Now, by defining $\tilde{\boldsymbol{\eta}}_d$ with null or zero value and considering as the control objective of the system, to verify which system is stable, a check is implemented with Lyapunov's method. Defining the function as $\mathbf{V}(\tilde{\boldsymbol{\eta}}) = \frac{1}{2}\tilde{\boldsymbol{\eta}}^T\tilde{\boldsymbol{\eta}} > 0$. Taking into consideration the first derivative $\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}}) = \frac{1}{2}\tilde{\boldsymbol{\eta}}^T\dot{\tilde{\boldsymbol{\eta}}}$ is replacing in $\dot{\tilde{\boldsymbol{\eta}}}_d = \dot{\tilde{\boldsymbol{\eta}}}_d - \dot{\tilde{\boldsymbol{\eta}}}$ taking into consideration $\dot{\boldsymbol{\eta}} = \mathbf{J}\dot{\mathbf{h}}$ is obtained $\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}}) = \tilde{\boldsymbol{\eta}}^T\dot{\tilde{\boldsymbol{\eta}}} = \tilde{\mathbf{n}}^T\left(\dot{\tilde{\boldsymbol{\eta}}}_d - \mathbf{J}\dot{\mathbf{h}}\right)$ consequently, the law of control is established by

$$\dot{\mathbf{h}}(t) = \mathbf{J}^{-1}\left(\dot{\boldsymbol{\eta}}_d + \mathbf{K}\tanh(\tilde{\boldsymbol{\eta}})\right) = \mathbf{J}^{-1}\dot{\boldsymbol{\eta}} \tag{10}$$

where, $\mathbf{K}$ expresses the diagonal matrix with positive gain. By deriving (10) with respect to time it is determined that

$$\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}}) = -\tilde{\boldsymbol{\eta}}^T\mathbf{K}\tanh(\tilde{\boldsymbol{\eta}}) < 0 \tag{11}$$

so it is established that the equilibrium point is asymptotically stable when, $\tilde{\boldsymbol{\eta}}(t) \to 0$ asymptotically.

5.2 *Control i-th Robot*

This section presents the decentralized control to be implemented in the i-th mobile manipulator robot, for which are considered as inputs the velocities emitted by the centralized training control, where the desired positions are obtained $\mathbf{h}_d = \begin{bmatrix} h_{xd} & h_{yd} & h_{zd} \end{bmatrix}$ with a variation $\dot{\mathbf{h}}_d = \begin{bmatrix} \dot{h}_{xd} & \dot{h}_{yd} & \dot{h}_{zd} \end{bmatrix}$ so that the control error is obtained by $\tilde{\mathbf{h}}(t) = \mathbf{h}_d(t) - \mathbf{h}(t)$ taking into account the derivation in time $\dot{\tilde{\mathbf{h}}} = \dot{\mathbf{h}}_d - \dot{\mathbf{h}}$. In order to demonstrate the stability of the system, it is proposed to use a controller based on Lyapunov. In which you have a candidate function that is defined as $\mathbf{V}(\tilde{\mathbf{h}}) = \frac{1}{2}\tilde{\mathbf{h}}^T\tilde{\mathbf{h}} > 0$. Taking into account the first derivative and replacing $\dot{\tilde{\mathbf{h}}} = \dot{\mathbf{h}}_d - \dot{\mathbf{h}}$ and considering that $\dot{\mathbf{h}} = \mathbf{\Gamma}\mathbf{v}$, where $\mathbf{\Gamma}$ represents the Jacobine matrix and $\mathbf{v}$ represents the control actions; with what is obtained $\dot{\mathbf{V}}(\tilde{\mathbf{h}}) = \tilde{\mathbf{h}}^T(\dot{\mathbf{h}}_d - \mathbf{\Gamma}\mathbf{v})$Thus, the law of control for the i-th mobile manipulator is defined as,

$$\mathbf{v} = \mathbf{\Gamma}^{-1}\left(\dot{\mathbf{h}}_d + \mathbf{G}\tanh(\tilde{\mathbf{h}})\right) \tag{12}$$

in which $\mathbf{G}$ represents the matrix that has positive diagonal gain. By deriving (12) with respect to time it is determined that

$$\dot{\mathbf{V}}(\tilde{\mathbf{h}}) = \tilde{\mathbf{h}}^T\mathbf{G}\tanh(\tilde{\mathbf{h}}) < 0 \tag{13}$$

which leads to say that the equilibrium point is asymptotically stable, therefore the error position the point of interest $\tilde{\mathbf{h}}(t) \to 0$ asymptotically $t \to \infty$

6 Experimental Results

In this part of the article you can see the results of the development and implementation of the HIL technique focused on the teaching process. The proposed system consists of two Raspberry Pi development cards were considered, in which the kinematic and dynamic models corresponding to each mobile manipulator robot obtained in (1) and (5). Each Raspberry Pi device is connected through a wireless communication channel to the Central Control Unit developed in Matlab mathematical software which is in charge of executing the collaborative control law. By using the DLL communication protocol, the communication between Matlab software and the Unity 3D development environment is carried out, where the user has the ability to interact and observe the movement that develops the robotic system to execute the various tasks of collaboration.

Fig. 4 shows the digitized environment which consists of an environment that simulates an industrial environment (Construction) in which two mobile manipulators are used as means of transporting heavy loads.

Fig. 4. Virtual Mobile Manipulator Environment

The evaluation was conducted using two mobile manipulators with a collaborative controller, for the transport of a bar with a defined trajectory, the simulation progresses in time, where the correction of errors made by the controller is checked. The manipulators acquire a formation pattern to make the trajectory and also keep the axis of the manipulated object in the center of the distance separating the manipulators.

Fig. 5, the results obtained from the experiment are presented, showing the desired task and the task performed, so that the controller responds optimally and independently.

ROBOT ONE
Q1 : -0,01 [rad/s]
Q2 : 0,15 [rad/s]
Q3 : -0,01 [rad/s]
phi : -0,01 [rad]
u : 0,2 [m/s]
w : 0,07 [rad/s]

TRAYECTORIA DESEADA 🟦
TRAYECTORIA OBTENIDA 🟩

ROBOT TWO
Q1 : 0,11 [rad/s]
Q2 : 0,16 [rad/s]
Q3 : -0,04 [rad/s]
phi : 0,01 [rad]
u : 0,22 [m/s]
w : 0,07 [rad/s]

Fig. 5. Trajectory described by the mobile manipulators.

The errors in the speeds generated by the controller for each mobile manipulator are presented in the curves in Figs. 6. and 7.

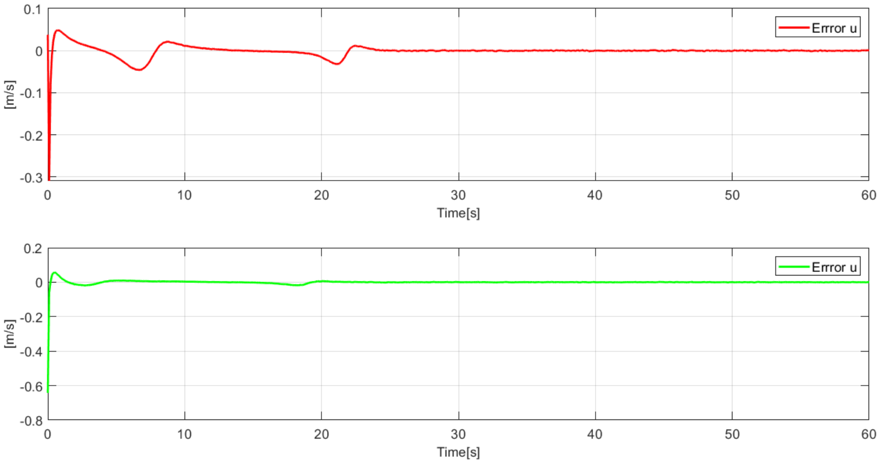

Fig. 6. Linear velocities errors of the Mobile Manipulators.

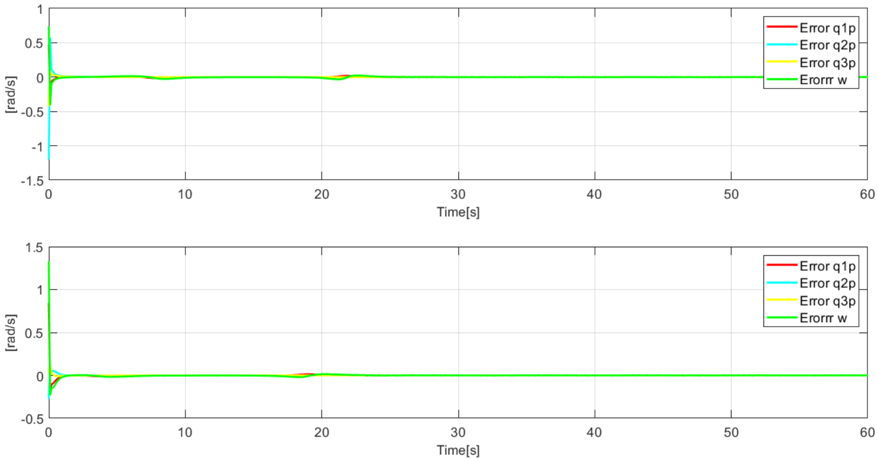

Fig. 7. Errors in the angular velocities of the Mobile Manipulators.

The errors corrected by the corrective action of the controller are presented in the curves in Figs. 8 and 9.

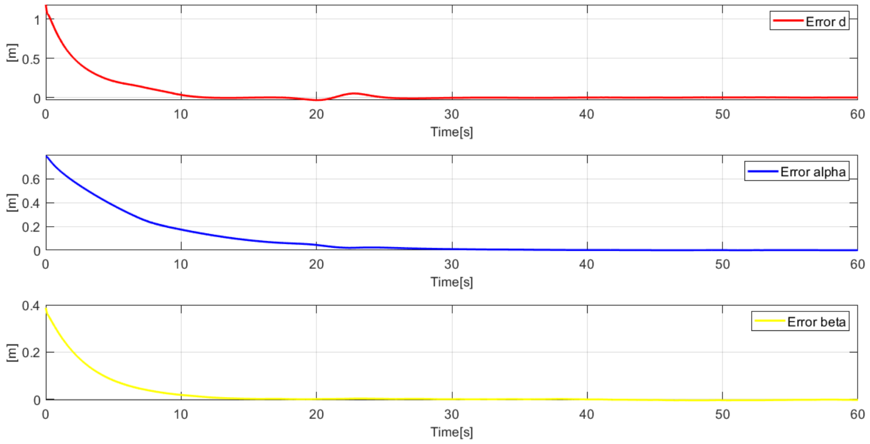

Fig. 8. Distance and orientation errors.

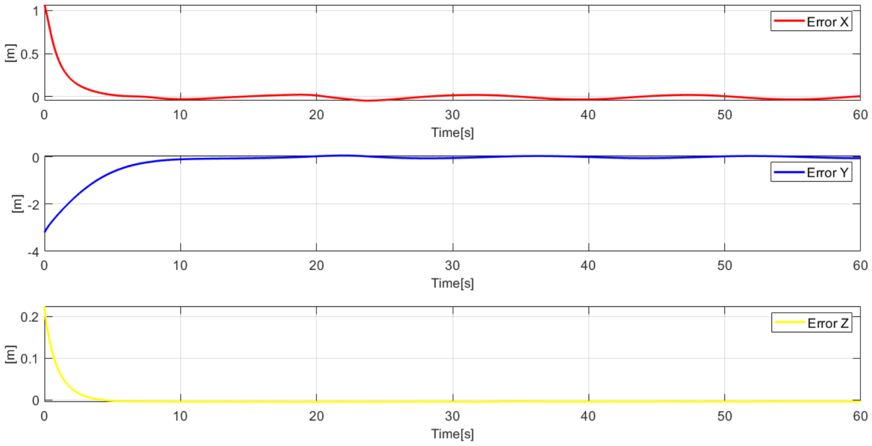

Fig. 9. Positioning errors.

7 Concluciones

This work presents the design of the Hardware in the Loop technique developed in a virtual environment, for the collaborative control between two robotic systems, in order to carry out tasks of moving and manipulating objects in an industrial environment. For the development of the control algorithms is composed of two stages, a kinematic cascade controller, which executes the path of the manipulated object; also a dynamic controller that acts as a compensator for errors generated by the dynamic effects of the robotic system. With the tests developed through the use of virtual system it was possible to verify and evaluate the control algorithm implemented, managing to demonstrate the validity of the control algorithm for the fulfillment of the established task, as presented in Fig. 8 and 9, the errors of position, distance and orientation of the manipulated object, tend satisfactorily to zero. In developing this project, the flexibility obtained by using an HIL system becomes evident, since it allows the creation of efficient systems with good performance at a much lower cost than the usual in today's industry.

ACKNOWLEDGMENT. The authors would like to thank the Coorporación Ecuatoriana para el Desarrollo de la Investigación y Academia- CEDIA for their contribution in innovation, through the CEPRA projects, especially the project CEPRA-XIV-2020-10 *"Multi-user immersive technologies aimed at synergistic teaching-learning systems";* also, the Universidad de las Fuerzas Armadas ESPE and the Research Group in ARSI, for the support for the development of this work.

References

1. D. Zuehlke, « Industry 4.0: More than a Technological Revolution, » Revista CEA, vol. 10, nº 5, 2019.
2. A. Sharkawy, C. Papakonstantinou, V. Papakostopoulos, V. Moulianitis y N. Aspragathos « Task Location for High Performance Human-Robot Collaboration, » J Intell Robot Syst, vol. 100, pp. 183–202, 2020.
3. J. Chen, S. Kai, « Cooperative transportation control of multiple mobile manipulators through distributed optimization, » Sci. China Inf. Sci., vol. 61, 2018.
4. M. Soltanpour, S. Zaare, M. Haghgoo y M. Moattari, « Free-Chattering Fuzzy Sliding Mode Control of Robot Manipulators with Joints Flexibility in Presence of Matched and Mismatched Uncertainties in Model Dynamic and Actuators, » J Intell Robot Syst, vol. 100, nº 51, pp. 47-69, 2020.
5. J. Acosta, G. Gonzáles, V. Andaluz y J. Garrido, « Multirobot Heterogeneous Control Considering Secondary Objectives, » Sensors, vol. 19, nº 20, 2019.
6. D. Ramos, L. Almeida y U. Moreno, « Integrated Robotic and Network Simulation Method, » Sensors, vol. 19, nº 20, 2019.
7. E. Bonilla, J. Rodriguez, J. Acosa y V. Andaluz, « Teaching and Learning Virtual Strategy for the Navigation of Multiple-UAV, » 15th Iberian Conference on Information Systems and Technologies (CISTI), 2020.
8. K. Herrera, J. Rocha, F. Silva y V. Andaluz, « Training Systems for Control of Mobile Manipulator Robots in Augmented Reality, » 15th Iberian Conference on Information Systems and Technologies, pp. 1-7, 2020.
9. M. Molina y J. Ortiz, « Coordinated and Cooperative Control of Heterogeneous Mobile Manipulators » International Conference on Social Robotics, vol. 11357, 2018.
10. M. Galicki, « Tracking the Kinematically Optimal Trajectories by Mobile Manipulators, » Journal of Intelligent & Robotic Systems, vol. 93, nº 51, pp. 635-648, 2019.
11. J. Varela, J. Buele, J. Jadan y V. Andaluz, « Teaching STEM Competencies Through an Educational Mobile Robot, » International Conference on Human-Computer Interaction, vol. 12206, 2020.

12. P. Leica, K. Rivera, S. Muela, D. Chávez, G. Andaluz y V. H. Andaluz, « Consensus Algorithms for Bidirectional Teleoperation of Aerial Manipulator Robots in an Environment with Obstacles, » IEEE Fourth Ecuador Technical Chapters Meeting (ETCM), pp. 1-6, 2019.
13. Z. Salcic, U. Atmojo, H. Park, A. Chen y K. Wang, « Designing Dynamic and Collaborative Automation and Robotics Software Systems, » IEEE Transactions on Industrial Informatics, vol. 15, nº 1, pp. 540-549, 2019.
14. P. Leica, J. Balseca, D. Cabascango, D. Chávez, G. Andaluz y V. H. Andaluz, « Controller Based on Null Space and Sliding Mode (NSB-SMC) for Bidirectional Teleoperation of Mobile Robots Formation in an Environment with Obstacles, » IEEE Fourth Ecuador Technical Chapters Meeting (ETCM), pp. 1-6, 2019.
15. G. Li, D. Zhang, Y. Xin, S. Jiang, W. Wang y J. Du, « Design of MMC hardware-in-the-loop platform and controller test scheme, » CPSS Transactions on Power Electronics and Applications, vol. 4, nº 2, pp. 143-151, 2019.
16. Y. Chen y D. J. Braun, « Hardware-in-the-Loop Iterative Optimal Feedback Control Without Model-Based Future Prediction, » IEEE Transactions on Robotics, vol. 35, nº 6, pp. 1419-1434, 2019.
17. H. Nguyen, G. Yang, A. Nielsen y P. Jensen, « Hardware- and Software-in-the-Loop Simulation for Parameterizing the Model and Control of Synchronous Condensers, » IEEE Transactions on Sustainable Energy, vol. 10, nº 3, pp. 1593-1602, 2019.
18. J. Varela, V. Andaluz y F. Chicaiza, « Modelling and Control of a Mobile Manipulator for Trajectory Tracking, » 2018 International Conference on Information Systems and Computer Science (INCISCOS), pp. 69-74, 2018.
19. V. Andaluz, J. Varela, F. Chicaiza, W. Quevedo, B. Ruales, « Teleoperation-of-a-mobile-manipulator-with-feedback-forces-for-evasion-of-obstacles, » Iberian Journal of Information Systems and Technologies, pp. 291–304, 2019.
20. J. Varela, F. Chicaiza, V. Andaluz, « Dynamics of a Unicycle-Type Wheeled Mobile Manipulator Robot, » Advances in Emerging Trends and Technologies, pp.24-33, 2020.