Adaptive control of a mobile robot for cargo transportation in industrial environments

Guerrero Vásquez, Fidel Napoleón y Proaño Jarrin, Karen Abigail

Departamento de Eléctrica y Electrónica

Carrera de Ingeniería en Electrónica e Instrumentación.

Artículo académico, previo a la obtención del título de Ingeniero en Electrónica e Instrumentación

Ing. Naranjo Hidalgo, César Naranjo

26 de julio del 2022

Latacunga

Adaptive Control of a Mobile Robot for Cargo Transportation in Industrial Environments

Fidel N. Guerrero, Karen A. Proaño, Cesar A. Naranjo and Víctor H. Andaluz

Universidad de las Fuerzas Armadas, Sangolquí-Ecuador

{fnguerrero,kaproano3,canaranjo,vhandaluz1}@espe.edu.ec

Abstract. This work focuses on the proposal of a cascade control scheme between a kinematic controller and an adaptive dynamic compensator. By using the Hardware in the Loop (HIL) technique, which allows the connection between hardware that simulates a real system with a computer that emits control signals; in this case, the hardware simulates the behavior of a robotic system with unicycle traction developed in a virtual reality (VR) environment for a teaching-learning process. To represent the behavior of this robotic platform, kinematic and dynamic models are found.; in the case of the dynamic model, a robotic platform with unicycle traction is built to estimate the dynamic parameters experimentally and validate the dynamic model obtained. In turn, this constructed robotic platform allows comparing the behavior of the controllers with those implemented in the HIL technique. The research demonstrates the favorable behavior of the controller cascading a proposed trajectory and changing the dynamics of the unicycle robot with different loads applied as in an industrial environment. The objective is to replace the use of physical platforms for the evaluation of new control algorithm proposals, reducing costs and even being focused on educational environments where the acquisition of physical robotic platforms is avoided.

Keywords: Unicycle Robot, Virtual Reality, Education, Hardware in the Loop.

1 Introduction

Currently, technological advances in the industrial field are focused on the fourth industrial revolution known as industry 4.0 in terms of the benefits to be gained from process automation: increased productivity and quality; detection of faults with electronic systems; reduction in downtime in production lines [1]. Manufacturing systems in Industry 4.0 are composed of sensors, intelligent machines and intelligent robots, so it is necessary to manage, store and index data, together with technologies that allow the grouping and interconnection of devices through an Internet network giving way to the use of IoT platforms (Internet of Things), which interact with humans in order to provide cyber-physical services such as object recognition for detection of object failures in production lines [2][3][4] .

Another booming technology today is Virtual Reality, which seeks to represent as much as possible real environments such as: the development of interactive systems for the simulation and testing of control strategies in rehabilitation tasks and robotic assistance for people with motor disabilities [5][6]; linkage with SAR systems (social assistance robots) for emotional relationship therapy with users [7]; industrial processes, virtualizing the scenarios through the use of CAD software for both industrial level purposes using as systems for testing control algorithms without causing damage to the physical system, This is also applicable to the educational environment, since it allows students to interact with a virtual system that simulates the behavior of real plants [8][9]. In these processes the inclusion of robotic systems, belonging to the field of industrial robotics, is also very notorious. Commonly used are those known as manipulators or robotic arms that by simulating the behavior of a human arm, benefits are obtained in the manipulation of objects, positioning precision and repeatability, fundamental attributes when automating tasks in production lines, manufacturing, assembly [10]. These robots have the peculiarity that most of them, due to their large size and weight, have a fixed base, limiting their work space. Therefore, mobile robotics are used for applications such as: tracking and trajectory control [11]; execution of tasks in supervision of the elderly, telecare, provision of medicines, accompanying dependent persons [12].

Mobile robotics in turn are classified according to the configuration of their drive systems as follows: *i) omnidirectional*, which stands out for the mobility obtained from its wheels, which allow the movement of the robotic platform in all directions [13]; *ii) car-like*, which is the basis for the modern automobile model and there are 2 front wheels for steering and 2 rear wheels for robot traction, widely used in research applicable to real vehicles [14]; *iii) uniciclo*, its simplicity in terms of its traction system allows to automate wheelchairs or similar structures for the transport of users with motor disabilities [5][12], that allow the displacement between points or to follow trajectories. And evaluation of different control algorithms and cooperative work control strategies [15].

Therefore, the present work is focused on the development of a HIL environment for testing high-level control algorithms on a unicycle robotic platform in a virtualized environment. For the simulation of the kinematic and dynamic behavior, mathematical modeling is developed; in the case of the dynamic model, obtaining the validation of the dynamic constants is performed by means of a robotic platform built. All this in order to achieve a model that resembles as closely as possible the behavior of a real robot; the control algorithms are evaluated considering scenarios similar to those that can be found in a real working environment. This allows them to be targeted for educational environments in which students need to experiment with these types of platforms.

The document is organized as follows: the formulation of the problem presented at the time of evaluating control algorithms in order to strengthen the teaching-learning process together with the mathematical modeling of the robotic platform are in Section 2. The description of the virtualized environment where the robot performs its tasks is in Section 3. The control scheme and the proposed control algorithms together with the respective stability analysis of the proposed controllers are in Section 4. Finally, Section 5 presents the results obtained in the present investigation.

2 Problem Formulation

Today's automation by mobile robots is covering more functions in the industrial sector for reasons of time, money and productivity. Similarly, in teaching and learning of future professionals to solve solutions in this area. However, there have been drawbacks in the consolidation of knowledge in a practical way and the study and implementation of new control algorithms; in the case of the present work, a unicycle-type mobile robot can' t be easily purchased, thus losing the ability of logical reasoning in the evaluation of such controllers in a practical form. Because the physical acquisition of these devices has a high cost and there is a risk of damage during the respective tests.

In order to strengthen the teaching-learning process, the construction of a unicycle-type mobile robot with more accessible and low-cost elements and sensors has been carried out, in which the kinematic and dynamic modeling of the robot has been studied to evaluate advanced control algorithms.

2.1 Kinematic Modeling

This type of robotic platform with unicycle type traction has a lineal velocity u and an angular velocity ω, in the axes from the reference frame of the unicycle robot $\Re_U$ (see Fig. 1). Therefore, obtaining the kinematic model describing the robot's position involves analyzing the positions and velocities necessary for trajectory tracking.

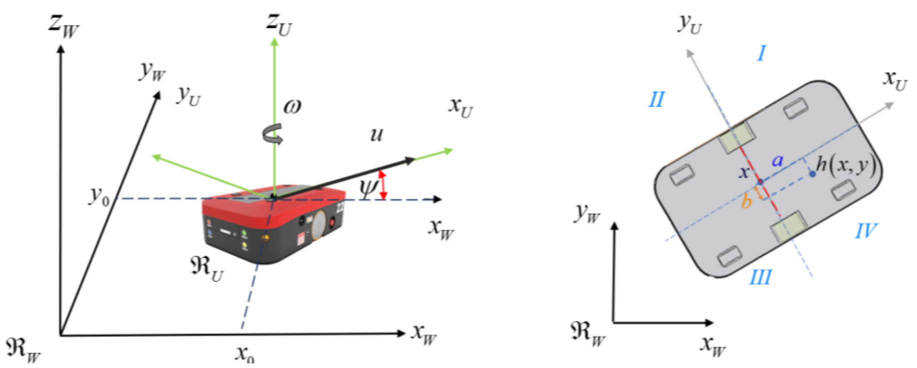


Fig. 1. Reference frame for the movement of the mobile robotic platform

The point of interest h is defined at any position of the robot, depending on the displacements a on the axis x_U and b on the axis y_U, as is the case in Fig. 1 where the point of interest is found displaced in the fourth quadrant of the robotic platform. This results in the following geometric model:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} + \Re_W^U \begin{bmatrix} a \\ b \end{bmatrix} \qquad (1)$$

By deriving (1), the instantaneous kinematic model of the unicycle robot with respect to the reference frame $\Re_W$ is obtained. This model can also be expressed in a matrix form in (2) in order to be used in control schemes considering that the velocity vector of the mobile robot is defined by $\mathbf{v} = [u \quad w]^T$, the rotation matrix $\Re_W^U$ which rotates from the unicycle robot frame to the world inertial frame and the position velocity vector $\dot{\mathbf{h}} = [\dot{x} \quad \dot{y}]^T$.

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} u\cos(\psi) \\ u\sin(\psi) \end{bmatrix} + \frac{d}{dt}\Re_W^U \begin{bmatrix} a \\ b \end{bmatrix} \qquad \text{with, } \Re_W^U(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \qquad (2)$$

2.2 Dynamic Modeling

When considering the dynamic model of the robotic platform, the control algorithms consider the effects of the intrinsic and extrinsic forces of the system. All this by means of the analysis of the relationships between the movements, forces and torsional moments that cause them. This is how the dynamic model (3) of the robotic platform with unicycle type traction obtained from the [16], In this project the dynamic model is based on this platform. This model has 8 dynamic parameters because it considers the displaced center of mass, which allows to represent the behavior when handling loads that are not located in the center of the robot, or due to the shape of the load.

$$\mathbf{v}_{ref}(t) = \mathbf{M}(\varsigma)\dot{\mathbf{v}}(t) + \mathbf{C}(\varsigma, \mathbf{v})\mathbf{v}(t) \qquad (3)$$

$$\mathbf{M}(\varsigma) = \begin{bmatrix} \varsigma_1 & -\varsigma_7 \\ -\varsigma_8 & \varsigma_2 \end{bmatrix} \qquad \mathbf{C}(\varsigma, \mathbf{v}) = \begin{bmatrix} \varsigma_4 & -\varsigma_3\omega \\ \varsigma_5\omega & \varsigma_6 \end{bmatrix}$$

where, $\mathbf{v}_{ref}$ is the vector of control signals or reference velocities of the robot; $\dot{\mathbf{v}}(t)$ robotic platform accelerations; $\mathbf{M}$ is the mass matrix; $\mathbf{C}$ the matrix of centrifugal forces; and both matrices contain the dynamic parameters $\varsigma \in \Re^l$ with $l = 8$, which will be calculated for each sampling period.

3 Virtual Environment

In teaching-learning, virtual environments should consider human-robot interaction with situations that arise in the industry. Therefore, this section describes the development of a virtual environment that allows observing the behavior of a unicycle robot in a simulated industrial environment. With the objective of evaluating advanced control algorithms when considering real situations, by considering the mathematical models obtained in previous sections inside the virtual environment; since this is how represent the behavior of the unicycle-type robotic platform.

The scheme for virtualization of the environment (see Fig. 2), consists of 4 blocks: *i) External Resources*, using software CAD the 3D elements are represented based on real elements, both the robot and different objects that are located in the virtual environment developed in the Unity 3D platform, to increase the immersion of an industrial environment; *ii) Scripting*, the programming scripts allow the emulation of the robot in a virtual environment by including the mathematical models obtained, in this block also includes the management of the resources with which the user interacts; *iii) Device Virtual*, these input-output devices are tools that increase user immersion and interaction with the virtual environment by directly contacting the visual and auditory senses; *iv) Controler*, allows the implementation of advanced control algorithms through a low-cost Raspberry-Pi board, closing the control loop by wireless communication. Matlab software that enables the exchange of data with the virtual environment by including Dynamic Link Library (DLL). In order to establish a link for the exchange of information without using an additional program, in this case the link between Matlab and Unity3D is created.

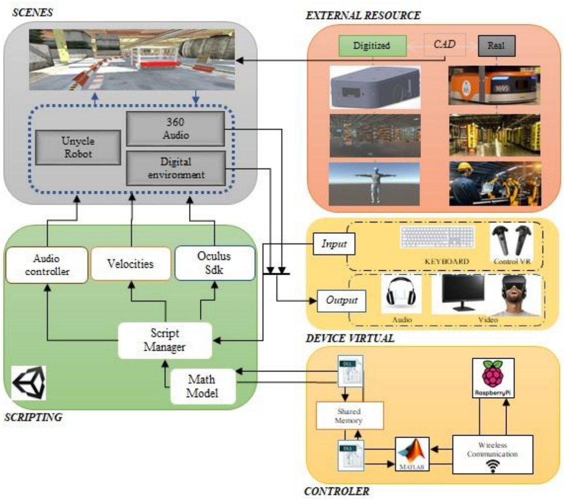


Fig. 2. Proposed diagram of the virtual environment.

4 Control Algorithm

The control scheme (see Fig. 3) is designed so the unicyle robot trajectory tracking and consists of two blocks: *i) the kinematic controller* in charge of calculating the position errors in each sampling period and are used to drive the mobile robot in a direction that decreases these errors; *ii) adaptive dynamic compensation*, to calculate the new parameters necessary for the compensation of the dynamics of the mobile platform in each sampling period.

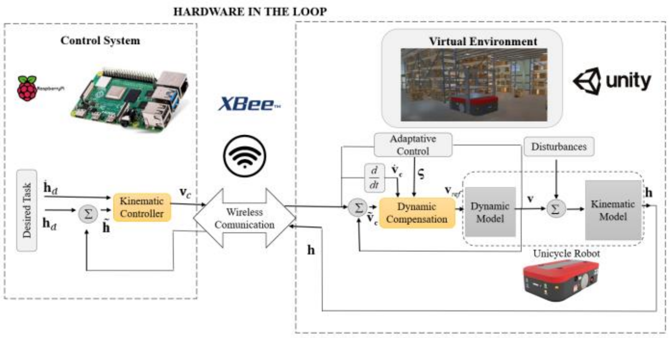


Fig. 3. Control scheme for the unicycle robotic platform using HIL.

4.1 Kinematic Control.

The kinematic controller is based on the calculated kinematic model and is represented as follows:

$$\mathbf{v}_c = \mathbf{J}^{-1}\left(\dot{\mathbf{h}}_d + \boldsymbol{\kappa}_{\tilde{\mathbf{h}}} \tanh\left(\boldsymbol{\kappa}_2 \tilde{\mathbf{h}}\right)\right) \qquad (6)$$

where, $\mathbf{v}_c$ represents the velocities calculated by the kinematic controller, $\mathbf{J}^{-1}$ the Jacobian matrix containing the inverse kinematics of the mobile robotic platform, $\mathbf{h}_d$ the matrix of desired velocities for trajectory compliance, $\boldsymbol{\kappa}_{\tilde{\mathbf{h}}} > 0$, $\boldsymbol{\kappa}_2 > 0$ are the weight or gain matrices positive defined to compensate for control errors and $\tilde{\mathbf{h}}$ the position error matrix itself which is saturated by the hyperbolic tangent.

4.2 Dynamic adaptive control.

The adaptive dynamic control allows to calculate the dynamic parameters in each sampling period, as robotic platforms tend to work in variable conditions. Both for transporting loads with different weights and working on different surfaces. This is omitted when calculating the dynamic constants with weights and under specific conditions only once, therefore, the design considers the dynamic model expressed as follows:

$$\underbrace{\begin{bmatrix} u_r \\ \omega_r \end{bmatrix}}_{\mathbf{v}_{ref}} = \underbrace{\begin{bmatrix} \varsigma_1 & -\varsigma_7 \\ -\varsigma_8 & \varsigma_2 \end{bmatrix}}_{\mathbf{M}} \underbrace{\begin{bmatrix} \dot{u} \\ \dot{\omega} \end{bmatrix}}_{\dot{\mathbf{v}}} + \underbrace{\begin{bmatrix} -\omega^2 & u & 0 & 0 \\ 0 & 0 & u\omega & \omega \end{bmatrix} \begin{bmatrix} \varsigma_3 \\ \varsigma_4 \\ \varsigma_5 \\ \varsigma_6 \end{bmatrix}}_{\boldsymbol{\eta}} \qquad (7)$$

and compactly (7) is represented by: $\mathbf{v}_{ref} = \mathbf{M}\dot{\mathbf{v}} + \boldsymbol{\eta}$ which results from considering (3) in its parametric form and then the control law (8)

$$\mathbf{v}_{ref} = \mathbf{M\sigma} + \boldsymbol{\eta} \tag{8}$$

where, $\sigma = [\sigma_1 \quad \sigma_2]^{\mathrm{T}}$ with the following conditions:

$$\sigma_1 = \dot{u}_c + k_u \tilde{u}, \quad k_u > 0,$$
$$\sigma_2 = \dot{\omega}_c + k_\omega \tilde{\omega}, \quad k_\omega > 0,$$

with velocities errors $\tilde{\omega} = \omega_c - \omega$ y $\tilde{u} = u_c - u$. Next, rewrite (8) in (9):

$$\mathbf{v}_{ref} = \mathbf{\Omega}(\sigma_1, \sigma_2, u, \omega)\boldsymbol{\varsigma} \tag{9}$$

where,

$$\mathbf{\Omega} = \begin{bmatrix} \sigma_1 & 0 & -\omega_c\omega & u_c & 0 & 0 & -\sigma_2 & 0 \\ 0 & \sigma_2 & 0 & 0 & u_c\omega & \omega_c & 0 & -\sigma_1 \end{bmatrix}, \boldsymbol{\varsigma} = [\varsigma_1 \ \varsigma_2 \ \varsigma_3 \ \varsigma_4 \ \varsigma_5 \ \varsigma_6 \ \varsigma_7 \ \varsigma_8]^T$$

Due to the uncertainties at the time of calculating the control actions caused by the parameter values used and the dynamic effects not considered in the modeling. In order to solve these problems and improve the performance of the controller, the values of the parameters used in the controller are adapted to reduce the errors. In the case of any uncertainty in the unicycle robot parameters, the control law:

$$\mathbf{v}_{ref} = \mathbf{\Omega}\hat{\boldsymbol{\varsigma}} = \mathbf{\Omega}\boldsymbol{\varsigma} + \mathbf{\Omega}\tilde{\boldsymbol{\varsigma}} = \mathbf{M\sigma} + \boldsymbol{\eta} + \mathbf{\Omega}\tilde{\boldsymbol{\varsigma}}$$

where, $\boldsymbol{\varsigma}$ y $\hat{\boldsymbol{\varsigma}}$ are the actual dynamic parameters and estimated parameters of the unicycle-type robotic platform, and the vector of errors of the dynamic parameters is represented by $\tilde{\boldsymbol{\varsigma}} = \hat{\boldsymbol{\varsigma}} - \boldsymbol{\varsigma}$.

4.3 Stability Analysis

By considering (7) in a compact form it follows $\mathbf{M}\dot{\mathbf{v}} + \boldsymbol{\eta} = \mathbf{M\sigma} + \boldsymbol{\eta} + \mathbf{\Omega}\tilde{\boldsymbol{\varsigma}}$ and is equivalent to $\mathbf{M}(\boldsymbol{\sigma} - \dot{\mathbf{v}}) = -\mathbf{\Omega}\tilde{\boldsymbol{\varsigma}}$. In which the following are $\boldsymbol{\sigma} - \dot{\mathbf{v}} = \dot{\tilde{\mathbf{v}}} + \mathbf{K}\tilde{\mathbf{v}}$, where the velocities errors are $\tilde{\mathbf{v}} = \mathbf{v_c} - \mathbf{v}$ y $\mathbf{K} = diag(k_u, k_\omega) > 0$ to obtain $\mathbf{M}(\dot{\tilde{\mathbf{v}}} + \mathbf{K}\tilde{\mathbf{v}}) = -\mathbf{\Omega}\tilde{\boldsymbol{\varsigma}}$, same that represents the equation of the controller's errors. Then propose a Lyapunov candidate function $V(\tilde{\mathbf{v}}, \tilde{\boldsymbol{\varsigma}}) = \frac{1}{2}\tilde{\mathbf{v}}^{\mathrm{T}}\mathbf{M}\tilde{\mathbf{v}} + \frac{1}{2}\tilde{\boldsymbol{\varsigma}}^{\mathrm{T}}\boldsymbol{\rho}\tilde{\boldsymbol{\varsigma}}$ and its partial derivative $\dot{V}(\tilde{\mathbf{v}}, \tilde{\boldsymbol{\varsigma}}) = -\tilde{\mathbf{v}}^{\mathrm{T}}\mathbf{MK}\tilde{\mathbf{v}} - \tilde{\mathbf{v}}^{\mathrm{T}}\mathbf{\Omega}\tilde{\boldsymbol{\varsigma}} + \tilde{\boldsymbol{\varsigma}}^{\mathrm{T}}\boldsymbol{\rho}\dot{\tilde{\boldsymbol{\varsigma}}}$. where, $\boldsymbol{\rho} \in \Re^{8x8}$ is a positive definite diagonal matrix, the matrix $\mathbf{M} > 0$ must be greater than zero and the vector of the real dynamic constants is considered constant with the errors $\dot{\tilde{\boldsymbol{\varsigma}}} = \dot{\hat{\boldsymbol{\varsigma}}}$.

By means of the law for updating parameters $\dot{\hat{\boldsymbol{\varsigma}}} = \boldsymbol{\rho}^{-1}\mathbf{\Omega}\tilde{\mathbf{v}}$ to substitute in the derivative of the Lyapunov candidate function $\dot{V} = -\tilde{\mathbf{v}}^{\mathrm{T}}\mathbf{MK}\tilde{\mathbf{v}} \le 0$, The same that allows to verify the stability with the control errors as bounded signals. For integration, $V(t) - V(0) = -\int_0^T \tilde{\mathbf{v}}^{\mathrm{T}}\mathbf{MK}\tilde{\mathbf{v}}dt$ and without considering $V(t)$ is obtained

$V(0) \geq \int_0^T \tilde{\mathbf{v}}^{\mathrm{T}} \mathbf{MK} \tilde{\mathbf{v}} dt$. With **MK** defined positive and symmetrical $\lambda_{\min}(\mathbf{MK})\|\tilde{\mathbf{v}}\|^2 \leq \tilde{\mathbf{v}}^T \mathbf{MK} \tilde{\mathbf{v}} \leq \lambda_{\max}(\mathbf{MK})\|\tilde{\mathbf{v}}\|^2$, where $\lambda_{\min}(.)$, $\lambda_{\max}(.)$ are the minimum and maximum eigenvalues of the matrix. Finally, there is (10) after the conditions analyzed in [17] considering the constants $\alpha_1 = \chi(\mathbf{MK})$ and $\mu_r = \chi(\mathbf{\Gamma})$.

$$\dot{V} \leq -\alpha_1 \|\tilde{\mathbf{v}}\|^2 - \mu_r \|\tilde{\varsigma}\|^2 + \tau, \;\; \tau = \mu_r \|\tilde{\varsigma}\|\|\varsigma\|. \tag{10}$$

In the analysis of control errors, considering the position error is determined by proposing a candidate Lyapunov function of quadratic errors (11) and its respective derivative (12)

$$\mathbf{V}(\tilde{\mathbf{h}}) = \frac{1}{2}\tilde{\mathbf{h}}^T(t)\tilde{\mathbf{h}}(t) \tag{11}$$

$$\dot{\mathbf{V}}(\tilde{\mathbf{h}}) = \tilde{\mathbf{h}}^T(t)\dot{\tilde{\mathbf{h}}}(t) \tag{12}$$

By considering velocities errors $\dot{\tilde{\mathbf{h}}}(t) = \dot{\mathbf{h}}_d(t) - \dot{\mathbf{h}}(t)$ and equalize with the control law (6) and the kinematic model results in a negative definite function

$$\dot{\mathbf{V}}(\tilde{\mathbf{h}}) = -\tilde{\mathbf{h}}^T(t)\boldsymbol{\kappa}_{\tilde{\mathbf{h}}} \tanh\left(\boldsymbol{\kappa}_2 \tilde{\mathbf{h}}(t)\right)$$

Therefore, in order to guarantee the stability of the control law, the condition must be fulfilled that $\boldsymbol{\kappa}_{\tilde{\mathbf{h}}} > 0, \boldsymbol{\kappa}_2 > 0$, to ensure that $\tilde{\mathbf{h}} \to 0$ when time tends to infinity.

5 Results

This section presents the results obtained from the implementation of the control algorithms, using the Hardware in the Loop technique through the development of the virtual environment where the mathematical models obtained from the unicycle robot are included to represent its behavior. In addition, a unicycle-type mobile robot with proprioceptive sensors was built to evaluate the proposed controller, (see Fig. 4).

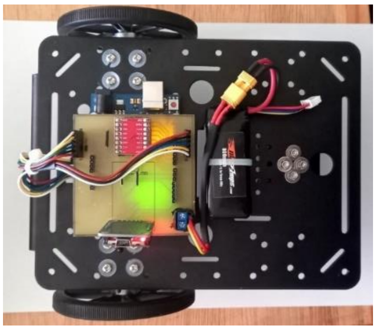


Fig. 4. Mobile unicycle robot built

5.1 Construction Description

This subsection details the construction of a unicycle type mobile robot which is divided into 4 blocks (see Fig. 5).

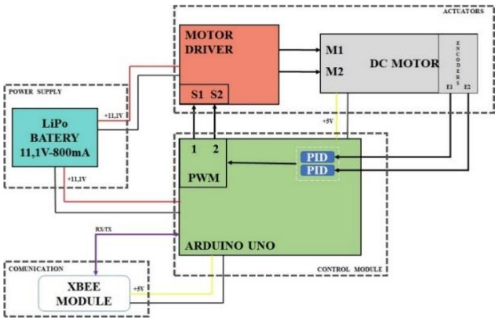


Fig. 5. Hardware block diagram of the unicycle mobile robot hardware.

i) Power supply: A LiPo battery (11.1V-800mA) provides power for the system; *ii) Control module*: there is an Arduino Uno, which transmits and receives the control signals; the controllers (PID) are used to compensate the internal dynamics of the mobile robot; *iii) Communication:* coordinates the connection between the robot and the Raspberry Pi board, considering a wireless transmission that allows high-speed data transmission over long distances, based on the IEEE 802.15 standard; *iv) Actuators*, DC motors with their respective velocity sensor to close the control loop and are also connected to a controller that supports a constant current of 1.2 A for each motor.

5.2 Identification and Validation

In the identification of the parameters of the dynamic model of the unicycle mobile robot, it is considered signals in the form of steps as excitation signals of each velocity performed by the unicycle robot.

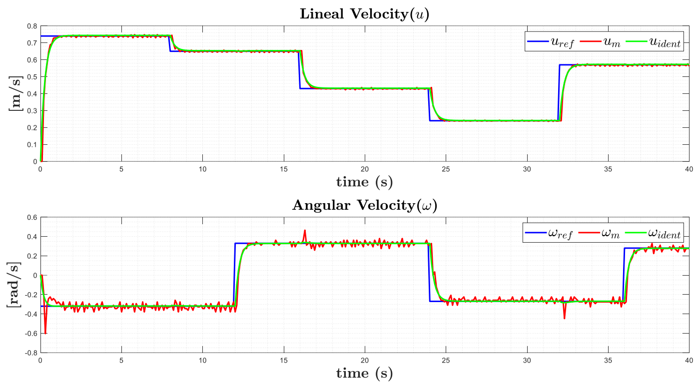


Fig. 6. Validation of dynamic parameters. The subscript *ref* represents desired value to the robotic system; *m* represents current velocity of the system; *ident* represents velocity calculated by the dynamic model.

The Fig. 6 shows the results obtained in the process of identification and validation of the computed dynamic model. The maximum interactions are applied to these data and the following parameters are obtained:

$$\zeta_1 = 0.3163,\ \zeta_2 = 0.2978,\ \zeta_3 = 0.0012,\ \zeta_4 = 0.9949,\ \zeta_5 = 0.0035,\ \zeta_6 = 1.0033,\ \zeta_7 = 0.0020,\ \zeta_8 = 0.0316$$

These values were taken on a tile floor.

5.3 Virtual Hardware in the Loop Implementation

The implementation of the HIL technique in a virtual environment (see Fig. 7) tries to resemble as much as possible an industrial environment where these types of robots are used for logistics; robots in these areas handle loads of different weights while following certain trajectories, which is one of the applications of adaptive dynamic control. The developed system allows the evaluation of the proposed controllers for the unicycle robot by observing the compliance of the trajectory assigned to the robotic platform in the virtual environment, including sounds and visual effects that increase the immersion in the virtualized environment. Control system is located in the Raspberry Pi card to perform the autonomous trajectory tracking; on the other part, the dynamic and kinematic models is located in the Unity3D software, which allows to simulate the behavior of the mobile robot. Finally, everything is linked through a wireless communication that allows to close the control loop.

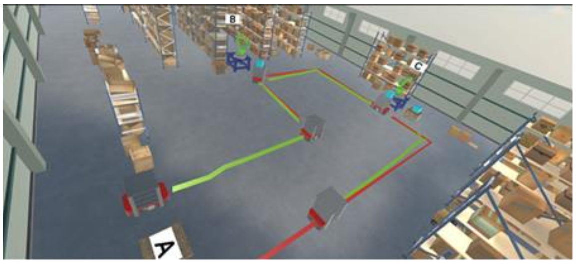
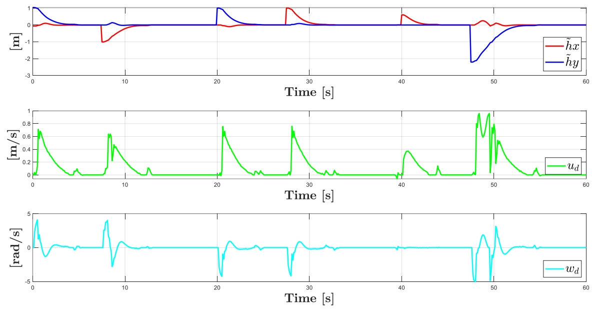
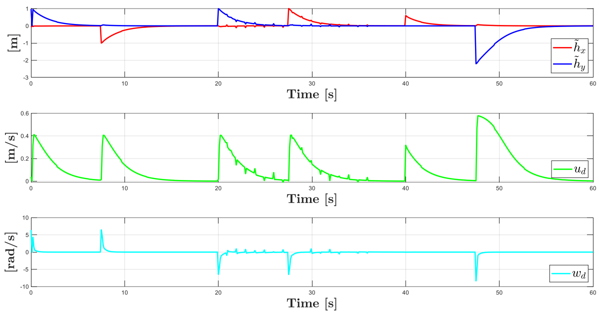


Fig. 7. Virtual environment of the unicycle mobile robot.

Then, when implementing the proposed control scheme, the following results were obtained (Fig. 8). The trajectory consists of three parts; section A-B shows the behavior of the robot without load; In section B-C a load is added to the robotic platform and finally section C-A is the return without load.



a) Constructed Robot



b) HIL Environment

Fig. 8. Control errors in experimental trajectory tracking tests.

The Fig. 8 shows the control errors of the proposed scheme considering a sampling period of 100 milliseconds, $\tilde{\mathbf{h}}_x$ defines the position error in meters that the robotic platform has with respect to the reference plane. $\Re_W$ on the axis X; while $\tilde{\mathbf{h}}_y$ defines it with respect to the axis Y. In the tests performed. The adaptive dynamic compensator is activated between 20 and 40 seconds and shows that in both cases the control errors decrease considerably, tending to zero in compliance with the assigned trajectory. Furthermore, it is remarkable the similarity of the behavior in the two graphs, due to the fact that the HIL technique considers both kinematic and dynamic mathematical models obtained and validated through the use of the built robotic platform.

6 Conclusions

Considering the kinematic and dynamic model allows us to simulate the behavior of the robotic platform more accurately; all this because the dynamic constants are obtained experimentally using a real robot, making the use of the HIL technique feasible; the behavior of the mathematical model obtained and the reading of the sensors located on the robotic platform have a high percentage of similarity in the validation process. This technique in conjunction with the virtual environment developed provides an immersive environment in which the operation of the virtualized robot and the operation of the control scheme in the fulfillment of the assigned trajectory can be observed in great detail, this is of great help for use in teaching-learning processes where, due to high costs, experimentation with new proposals for advanced control algorithms is limited. Finally, the adaptive dynamic compensator in conjunction with the kinematic controller in cascade control mode shows a good performance in this type of unicycle type traction robotic platform focused on the work of handling loads on different media or surfaces, since the dynamic parameters are calculated in each sampling period adapting to the changes that the robot may present, such as increased loads or change in the displacement Surface.

References

1. Webert, H., Döß, T., Kaupp, L., Simons, S.: Fault Handling in Industry 4.0: Definition, Process and Applications. Sensors 22(6), (2022).
2. Tsanousa, A., Bektsis, E., Kyriakopoulos, C., González, G.A., Leturiondo, U., Gialampoukidis, I., Karakostas, A., Vrochidi, S., Kompatsiaris, I.: Review of Multisensor Data Fusion Solutions in Smart Manufacturing: Systems and Trends. Sensors 22(5),(2022)
3. Fortino, G., Guerrieri, A., Pace, P., Savaglio, C., Spezzano, G.: IoT Platforms and Security: An Analysis of the Leading Industrial/Commercial Solutions. Sensors 22(6), (2022).
4. Jian, M.-S., Pan, C.-J.: Blockchained Industry Information Handoff Based on Internet of Things Devices with Intelligent Customized Object Recognition. Sensors22(6), (2022).

5. Ortiz, J.S., Palacios-Navarro, G., Andaluz, V.H., Guevara, B.S.: Virtual Reality-Based Framework to Simulate Control Algorithms for Robotic Assistance and Rehabilitation Tasks through a Standing Wheelchair. Sensors 21(15), (2021).
6. Caporaso, T., Grazioso, S., Di Gironimo, G.: Development of an Integrated Virtual Reality System with Wearable Sensors for Ergonomic Evaluation of Human–Robot Cooperative Workplaces. Sensors 22(6), (2022).
7. Cohavi, O., Levy-Tzedek S.: Young and old users prefer immersive virtual reality over a social robot for short-term cognitive training. International Journal of Human-Computer Studies, vol. 161, (2022).
8. Wang, J., Qi, Y.: A Multi-User Collaborative AR System for Industrial Applications. Sensors 22(4), (2022).
9. Cueva, C.A., Erazo, K.G., Naranjo, C.A., Saá, F., Andaluz, V.H.: Virtual Environment of an Industrial Process for Learning and Teaching. In: Mesquita, A., Abreu, A., Carvalho, J.V. (eds) Perspectives and Trends in Education and Technology. Smart Innovation, Systems and Technologies, vol 256. Springer, Singapore (2022).
10. Kana, S., Gurnani, J., Ramanathan, V., Turlapati, S.H., Ariffin, M.Z., Campolo, D.: Fast Kinematic Re-Calibration for Industrial Robot Arms. Sensors 22(6), (2022).
11. Mu, S., Shibata, S., Yamamoto, T.: Development of a User-following Mobile Robot with a Stand-up Assistance Function. Cognitive Robotics, pp. 83–95 (2022).
12. Pallares, B., Rozo, T., Camacho, E., Guarnizo, J., Calderon, J.: Design and construction of a cost-oriented mobile robot for domestic assistance. IFAC-PapersOnLine, vol. 54, no. 13, pp. 293–298. (2021).
13. Gallo, L.V., Paste, B.D., Ortiz, J.S., Andaluz, V.H.: Control of an Omnidirectional Robot Based on the Kinematic and Dynamic Model. In: Botto-Tobar, M., Montes León, S., Camacho, O., Chávez, D., Torres-Carrión, P., Zambrano Vizuete, M. (eds) Applied Technologies. ICAT 2020. Communications in Computer and Information Science, vol 1388. Springer, Cham (2021).
14. Qadeer, N., Shah, J.H., Sharif, M., Khan, M.A., Muhammad, G., Zhang, Y.-D.: Intelligent Tracking of Mechanically Thrown Objects by Industrial Catching Robot for Automated In-Plant Logistics 4.0. Sensors 22(6), (2022).
15. Quispe, M.A., Molina, M.C., Ortiz, J.S., Andaluz, V.H.: Unicycle Mobile Robot Formation Control in Hardware in the Loop Environments. In: Botto-Tobar, M., Montes León, S., Camacho, O., Chávez, D., Torres-Carrión, P., Zambrano Vizuete, M. (eds) Applied Technologies. ICAT 2020. Communications in Computer and Information Science, vol 1388. Springer, Cham (2021).
16. Ortiz, J., Palacios, G, Andaluz, V.H. Recalde, L.: Three-Dimensional Unified Motion Control of a Robotic Standing Wheelchair for Rehabilitation Purposes, Sensors 21(9), (2021).
17. Martins, F., Celeste, W., Carelli, R., Sarcinelli-Filho, M., Bastos-Filho, T.: An adaptive dynamic controller for autonomous mobile robot trajectory tracking. Control Engineering Practice, vol. 16, no. 11, pp. 1354-1363, (2008).