ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Control de un robot manipulador SCARA

Mullo Aimacaña, Rubén Darío

Vicerrectorado de Investigación, Innovación y Transferencia de Tecnología

Centro de Posgrados

Maestría en Electrónica y Automatización mención Redes Industriales

Artículo académico, previo a la obtención del título de Magíster en Electrónica
y Automatización mención en Redes Industriales

Ing. Andaluz Ortiz, Víctor Hugo PhD.

4 de septiembre del 2021

Control of a SCARA Robot Manipulador

Rubén D. Mullo and Víctor H. Andaluz

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

Abstract. This document describes the implementation of a 3D virtual simulator to evaluate the performance of control algorithms, applied to the SCARA robot manipulator for the execution of autonomous tasks. The virtual simulation environment allows immersion and interaction of users in order to manipulate and control robot manipulator tasks within laboratory and virtualized environments for industrial processes. In order to get the robot-environment interaction, is necessary to stablish the Unity 3D graphic engine because it exchanges information with Matlab software to analyze the control strategies. The information transfer is bidirectional and in real time for feedback within the control loop. Finally, the simulation is evaluated by incorporating a control scheme to observe the response of the operator to different trajectories created according to the virtual environment.

Keywords: Virtual Reality**,** Robot Simulation, Control Algorithm, Path Following, Industrial Processes.

1 Introduction

Worldwide, industry is in constant technological evolution by virtue of the incorporation of new automatic systems based on the inclusion of different configuration robotic arms to improve the production process[1]. Using manipulative robots increases production, minimizes the loss of raw material, reduces manufacturing times, increases quality standards, optimizes maintenance times and equipment stoppages, among other aspects. [2, 3]. Presently, in the industrial sector, the most used robots are the anthropomorphic robot manipulator and the SCARA robot manipulator because their morphology is designed to maintain a high repeatability index in the movements, paths and positions to execute operations of material handling, assembling and object selecting [4]. In several industrial processes, platforms are introduced for the object's movement or the coupling with robots to move according to the tasks assigned in the production lines [5, 6]. In the scientific community, there is great interest in proposing different control strategies to improve the manufacturing processes developed by each of the robotic systems, especially manipulative robots [7]. Projects like [8] focus on making standard and efficient control structures to reduce the effort developed by computer systems in robotic arms. However, it is important to mention that manufacturers do not contribute to this objective because they only allow them to be applied to each company manufactured robots. Control strategies must allow to perform complex tasks efficiently and comply with the industrial requirements, in this regard, [9] they introduce an algorithm based on the conventional sliding mode control to manipulate the robot movement in object collection activities and placement in different positions depending on the application. However, one of the problems involved in the experimental evaluation of the proposed control algorithms is the lack of equipment for robotic systems due to high costs, large infrastructures, and other aspects. These tests cannot be applied in a simple way within the companies' industrial processes because they would cause problems in production, product delays and economic losses due to stopping the process lines [10, 11]. The creation of emulators to develop tests to robots interactively are the possible solutions to evaluate control strategies [12].

To continue, a virtual environment allows simulating a robotic system with the purpose of incorporating the physical characteristics of a robot and the environment that surrounds it, in order to perform a real functioning analysis [13]. There are several commercial programs to simulate robotic systems in virtual environments, among the best known are the following: Robcad, RobotStudio, Igrip, Gazebo, Workcell, Factory I/O, KUKA SimPro. These software packages are used for design, simulation, optimization, analysis and offline programming of robotics, without interrupting production and coupling multiple devices [14]. Nevertheless, not all these programs are compatible with all robots, there are certain limitations in the libraries support of each one, some of them do not work on Windows and others can only be applied to their own robots [15]. These computational systems are based on the definition of positions, paths and tasks to develop on the assigned work area, however, do not allow the implementation of control algorithms for evaluating and simulating a desired activity [16].

According to the above, in order to generate an alternative to test and evaluate new control algorithms or for maintenance processes in production lines, the development of a 3D virtual environment is proposed, allowing evaluating control strategies in a robotic system. For the implementation of the virtual environment, the modeling of a SCARA robot manipulator; the devices and industrial equipment is developed using CAD software, and to virtualize a manufacturing process and a robotics laboratory developed in the Unity 3D graphic engine. In addition, the kinematic and dynamic modeling of the manipulator is incorporated through mathematical software to establish an advanced control strategy that allows the robot to move autonomously and strengthen the realism of the environment. The development of the 3D virtual environment considers two options*: (i) Robotic laboratory* aimed at knowing the morphology of the robot where it is possible to handle it through laboratory practices developing point-to-point movements, paths, and performing certain autonomous tasks; making the user has a virtual training about the handling of industrial robots. And, (*ii) Industrial environments*. to clarify, virtual environments are designed based on the union of manipulative robots in production lines, where the user can identify each of the manufacturing areas following the safety standards and procedures. In addition, it is possible develop maintenance plans in the work sections to reduce errors and minimize loss of time. The user has the possibility to recognize and solve real problems that can occur in the industrial plant.

This project is divided into 6 sections including the introduction. The description about the work environment system is explained in section 2. Section 3 describes the multilayer scheme of the application of the industrial virtual environment structure. To continue, section 4 shows the modeling of the SCARA robot manipulator and the design of the advanced control strategy. Section 5 represents the experimental results of the robot manipulator and its applications. Finally, the conclusions are detailed in section 6.

2 System Description

The description of the developed platform consists of the interaction of four blocks: scene, inputs, scripts and outputs as shown in Fig. 1. The environment generated by the interaction allows to simulate controlled systems of robotic manipulators in order to validate control strategies through a didactic learning, an industrial virtualization and a realistic environment.

In the *Scene block* there are the home, the robotics laboratory and an industrial process. *(i) Home* contains the different scenes that the user can choose to start the application; (*ii) Robotic laboratory*, the morphological recognition of the manipulator is established, the creation of point-to-point movements or defined paths for the operator adaptation, meanwhile; (*iii) Industrial process*, a manufacturing process is developed using robotic systems for the assembly of electronic cards and maintenance on the production lines. The system contains elements to generate the realism of the environment through Audio Game Objects, UI canvas, interaction methods, 3D Model manipulation robot and avatar. All of them are interconnected to control modules in order to answer the needs of each virtual environment.

In the entry and exit stage there are devices that allow the immersion and interaction of robotic systems, for example, virtual reality helmets (HTC VIVE and Gear VR) and haptic input controls. These devices are versatile because they have compatibility with different platforms and drivers.

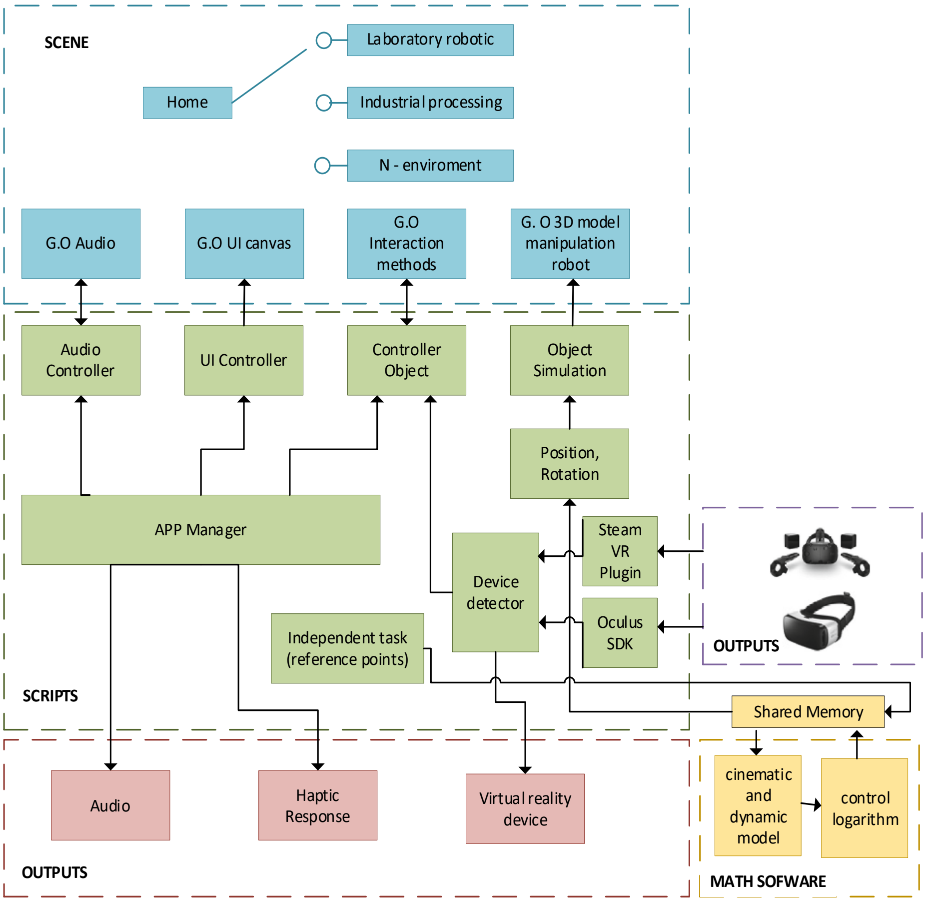


Fig. 1. System description diagram

In the *Script block*, there are several interconnected control blocks to manage the communication between input and output devices and the APP manager allowing to define the operating characteristics of each virtual environment. Therefore, user will be able to choose the robotic system mode through the user interface by selecting the robotic laboratory or an industrial process. The Object Controller will allow to manipulate the industrial equipment of the virtual environment and the location in the work area. The object simulation is directed to the SCARA robot manipulator model that will execute movements according to the assigned task in the virtual environment. In addition, the platform has multi-user support for the interaction of several people at the same time within the 3D environment.

The *Mathematical Software block* has a kinematic and dynamic modeling of a SCARA robot manipulator. In this process, the input variables are paths or points that are assigned according to the task to be developed by the manipulator in the virtual environment, linking to a control algorithm through numerical methods. The control solution will return to 3D virtualization throughout shared memories to observe and evaluate the robustness of the controller.

Finally, in the *Output block*, the relationship between components provides a sensation of a real environment through a total immersion of the audio, a haptic response processing to the inputs in order to improve the interaction between the environment and the user.

3 Virtual System Structure

The virtual reality application contemplates a realistic environment of a robotic laboratory and an industrial process for assembling electronic cards, through the implementation of manufacturing equipment for each SCARA robot manipulator, as shown in Fig. 1. On the other hand, virtual immersion allows understanding interactively the behavior of the manipulator under a control action. For the selection of the robotic system, the user can choose the virtual environment as shown in fig. 2.

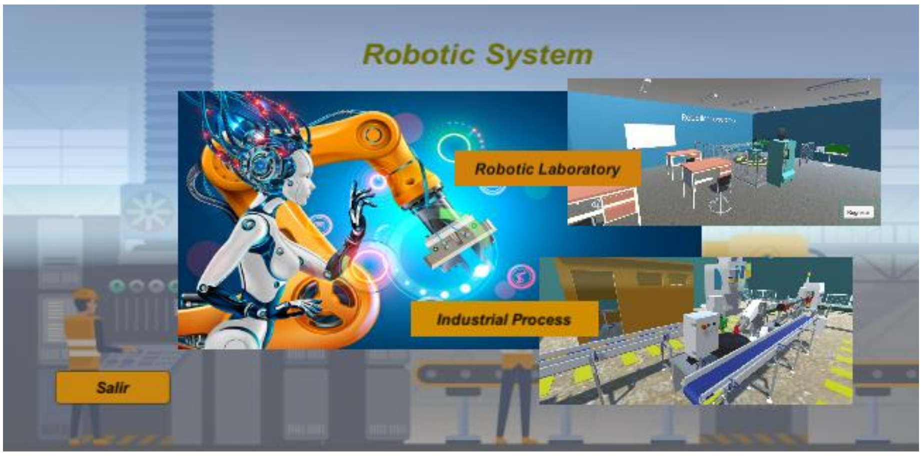


Fig. 2. VR. Application developed in the Game Engine Unity 3D

To achieve this project, the development is based on the execution of a multilayer scheme where all the parameters required for the advancement of the application are attached as shown in Fig. 3.

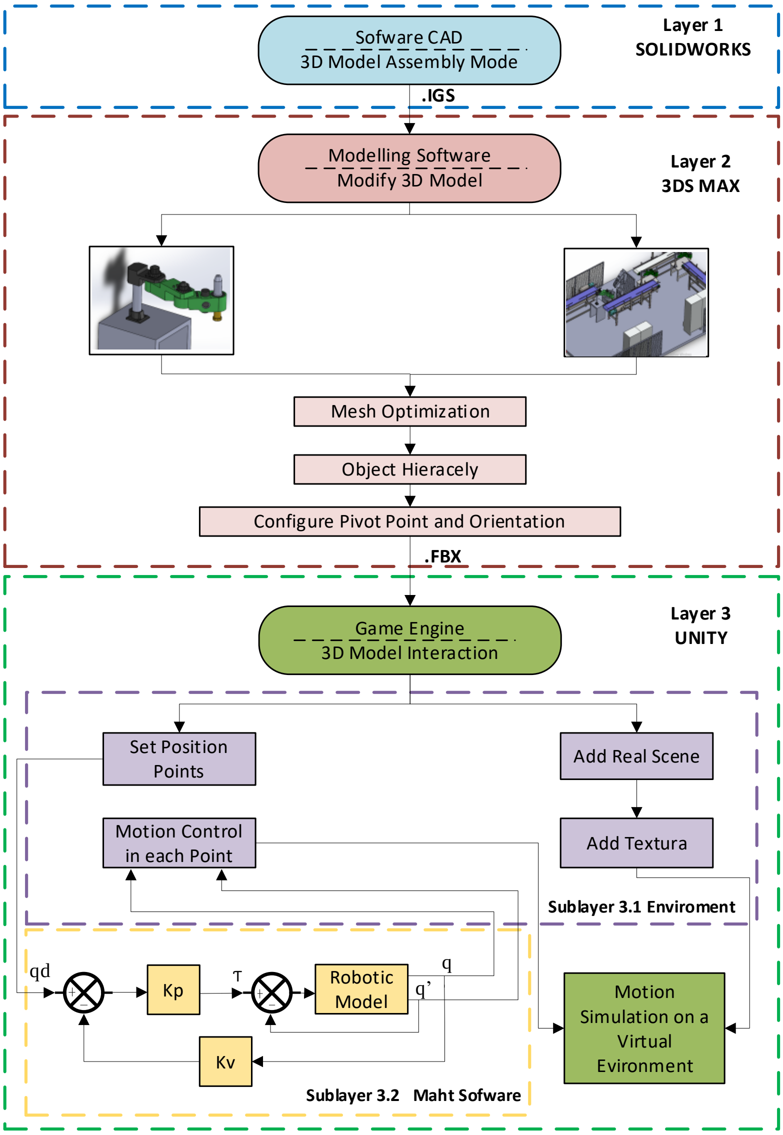


Fig. 3. Virtualization Model

Layer 1. It is the initial phase for the virtualization process. The design, the creation and the import of parts, and 3D assembly is developed in the CAD-SOLIDWORKS software. This tool allows an integrated work for the modeling and validation of the parts that constitute the virtual application, ex, SCARA robot manipulator, conveyor belts, robotic arm, electrical control boards, work tables, control room, electronic accessories, among others, as shown in Fig. 4. Finally, the model with extension *.igs, must be saved to continue with the virtualization procedure with 3DS MAX software.

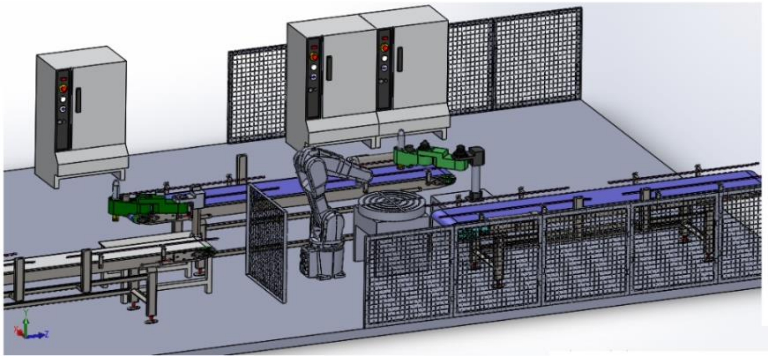


Fig. 4. Electronic card Machining and Assembly Process developed in SolidWorks

Layer 2. In this phase, the CAD 3DS MAX software is applied to generate modeling rendering parameters, hierarchies of the parts according to the number of devices, position and movement restrictions concurring to the operation of each element, textures, and orientation of the parts through a reference or pivot point. Files with the extension *.FBX must be exported for being import into the Game Engine Unity 3D virtual reality environment.

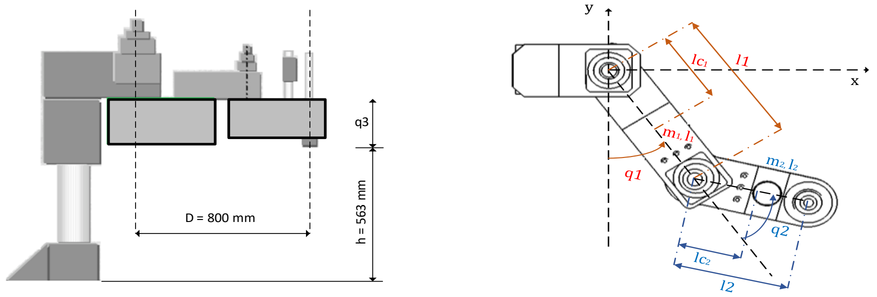
Layer 3. In this phase, the import of the models takes place and the simulation of the virtual environment of both the laboratory and the industrial process of machining and assembling electronic cards is developed using Game Engine Unity 3D. The application implements virtualization and immersion scenes to observe the application of the SCARA robot manipulator within each section and the control that must be maintained in order to avoid having problems in the production lines, making possible to simulate and try real life problems in a didactic way, in order to prevent collisions with the robot's end effectors, production losses, order delays, maintenance time, and other problems.

Sublayer 3.1. This scene determines the section of the robotic manipulators for the location at each work station based on the area and the production line. For the autonomously development of the tasks, a set of points or paths is implemented for the positioning of the end effector. In the first section, the cards, prior to machining, are transported by a band which is collected by a robot manipulator 1 and placed on the work table. On the other hand, during the second section, the electronic card is assembled with the different electronic devices by a manipulator 2. The laboratory is designed with a robot manipulator and some industrial devices for the development of user didactic activities.

Sublayer 3.2. The mathematical model of the SCARA robot manipulator with 3 degrees of freedom is one of the most important. It is based on the kinematics and dynamics of the robot according to its general characteristics, providing adequate control through the implementation of an algorithm based on numerical methods for the execution of manipulator movements in the virtualization environment. Modeling and control are developed with MATLAB software that simulates mathematics, physics, and other applications in real time and interacts with Game Engine Unity3D.

4 Modeling and Control

The modeling of the SCARA robot manipulator is based on the kinematics and dynamics of its morphological structure, see Fig. 5 (b). This modeling is important because IT allows studying the behavior of the manipulator under a control action, it generates an interaction between the robot and the virtual simulation environment. The implementation of the mathematical model is performed in the Matlab software and the exchange of information through a shared memory for the bilateral transmission of data with the virtual environment realized in the Game Engine Unity3D, see Fig. 3 (layer 3). The robot used for the analysis is a rotary arm robot of the Bosch RS800 series with a reach of 800 mm and its third axis vertically of 320 mm, see Fig. 5 (a).



(a) Robot manipulator Structure (b) Robot manipulator Geometric diagram

Fig. 5. Robot manipulator Bosch RS 800

The *direct kinematic model of the robot* establishes a reference system at the base of the robot (first joint) and describes the location of each of the links with respect to said system. For the model, it depends on the dimensions of the robot (l_1, l_2) and the values of the joint angle (q_1,q_2). The z axis is defined as; $z = h - q_3$, where q_3 is the position of the linear joint, and h is the distance between the first joint and the base of the robot.

$$T = {}^0A_1\ {}^1A_2 = \left[\underbrace{\boxed{\begin{matrix} cos\,(q_1+q_2) & -\left(sin\,(q_1+q_2)\right) & 0 \\ sin\,(q_1+q_2) & cos\,(q_1+q_2) & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{matrix}}}_{\text{Rotation in the z axis}} \underbrace{\boxed{\begin{matrix} l_1cos(q_1)+l_2cos(q_1+q_2) \\ l_1sin(q_1)+l_2sin(q_1+q_2) \\ 0 \\ 1 \end{matrix}}}_{\text{Position in the base system}} \right]$$

$$x = l_1sin(q_1) + l_2sin(q_1+q_2)$$
$$y = l_1cos(q_1) + l_2cos(q_1+q_2)$$

The *dynamic model that represents the Robot manipulator* is based on the Lagrange formulation (energy balance), that is, on the difference between the kinetic and potential energy of the robot's joints.

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) + f(q) = T$$

where, $M(q)$ is the inertia matrix $\in R^{n\,x\,n}$; $C\,(q,\dot{q})$ is the Coriolis and the centripetal force caused by the movement of the robot $\in R^{n\,x\,n}$; $g(q)\ \in R^{n}$ gravity vector; and $f\ (q)\ \in R^{n}$ represents friction or disturbances. For the types of robots whose movements are made in the horizontal plane, the gravitational term is minimized. The z axis is modeled based on the moment of force exerted by a motor on the transmission shaft as shown in the equation.

$$\tau_m(t) = J_m\ddot{\theta}_m(t) + \tau_l(t) + \tau_c(t)$$

where, $\tau_m(t)$ represents the motor torque, $\ddot{\theta}_m(t)$ the motor angular acceleration, J_m the rotor inertia, $\tau_l(t)$ the load torque seen from the motor shaft, $\tau_c(t)$the friction torque.

The *control system* is based on two forms: MIMO system applied to $q1$ and $q2$ and an independent SISO system for the z axis. This control algorithm operates on the generated error signal to reduce this value and reach the desired position. The control law is based on PD control and the design of symmetric and positive matrices K_p and K_v.

$$\tau = K_p\,\tilde{q} + \ K_v\,\dot{\tilde{q}}$$

where, $K_p \in R^{n\,x\,n}$ position gain matrix, $K_v\ \in R^{n\,x\,n}$ velocity gain matrix, $\tilde{q}$ position error, $\dot{\tilde{q}}$ velocity error. For determining $\tilde{q}$ is established $\tilde{q} = q_d - q$, where q_d desired position and q previously assigned position. This law applies to the third degree of freedom on the z-axis by $h_{ze} = h_{zd} - h_z$, where h_{ze} height error z axis.

The following, the stability analysis for the joints is presented q_1 y q_2 by Lyapunov's direct method, considering the following function $V(\tilde{q},\dot{q}) = \frac{1}{2}\ \dot{q}^T M(q)\dot{q} + \frac{1}{2}\tilde{q}^T K_p\tilde{q}$. Obtaining the first derivative with respect to time, $\dot{V}(\tilde{q},\dot{q}) = \dot{q}^T M(q)\ddot{q} + \frac{1}{2}\dot{q}^T \dot{M}(q)\dot{q} + \tilde{q}^T K_p\dot{\tilde{q}} = -\dot{q}^T K_v\dot{q}\ \leq 0$. Now considering LaSalle's theorem by means of the Ω set given by:

$$\Omega = \left\{x \in R^{2n}\colon \dot{V}(x) = 0\right\}$$
$$\Omega = \left\{x = \begin{bmatrix}\tilde{q}\\ \dot{q}\end{bmatrix} \in R^{2n}\colon \dot{V}(\tilde{q},\ \dot{q}) = 0\right\}$$
$$\Omega = \{\tilde{q} \in R^{n}, \dot{q} = 0 \in R^{n}\}$$

where, $\dot{V}(\tilde{q},\dot{q}) = 0$ si $\dot{q} = 0$ and for a solution $x(t)$ to belong to Ω for all $t\ \geq\ 0$ it is necessary that $\dot{q}(t) = 0$ for all $t\ \geq\ 0$. Consequently, it follows that if $x(t)\ \in\ \Omega$ for all $t\ \geq\ 0$, then:

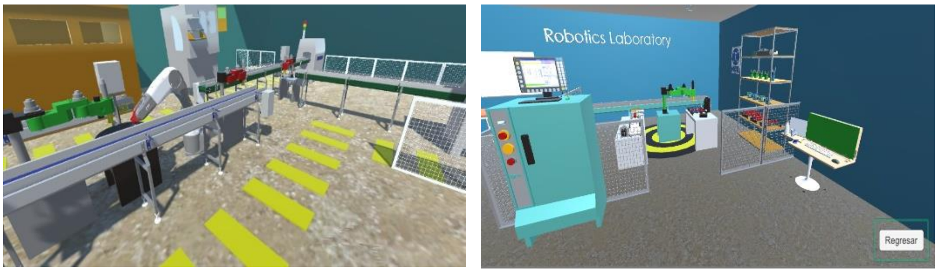
$$0 = M(q_d - \tilde{q}(t))^{-1}K_p\,\tilde{q}(t)$$

Therefore, $[\tilde{q}(0)^T\ \dot{q}(0)^T]^T = 0\ \in R^{2n}$, is the condition on Ω for which $x(t) \in\ \Omega$ for all $t\ \geq\ 0$, this indicates the stability of the system. Finally, it is concluded:

$$\lim_{t\to\infty}\tilde{q}(t) = \lim_{t\to\infty}[q_d - q(t)] = 0$$
$$\lim_{t\to\infty}\dot{q}\,(t) = 0$$

5 Resultado y discusión

This Section presents the performance of the 3D virtual application and the behavior of the SCARA robot manipulator under a control action. Two environments are presented; Robotic Laboratory and Industrial process, allowing the immersion and interaction of the user with the different industrial equipment. Experiments on point-to-point autonomous control and continuous paths are developed within a virtual simulation environment developed in the Game Engine Unity3D, see Fig. 6.



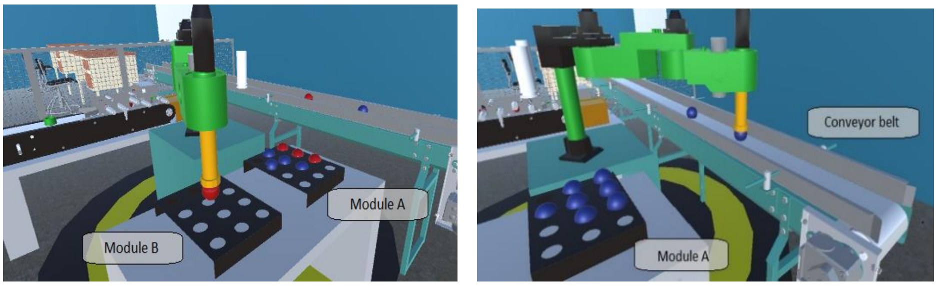
(a) Industrial Process (b) Robotic Laboratory

Fig. 6. 3D virtual environment design

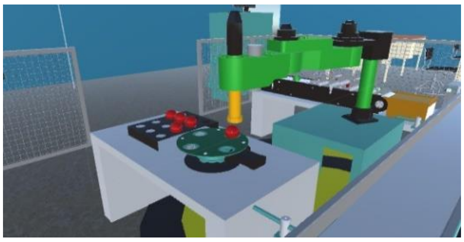
Robotic Laboratory is aimed at practically strengthening the handling and control of an industrial robot manipulator of the SCARA typology. The developed environment consists of a robotic cell equipped with conveyor belts, control cabinet, electrical control board, mechanical parts, servomotors, end effector, among others. The user can enter the robotic cell to recognize the robot morphological in a dynamic and interactive way, as shown in Fig. 7.

Fig. 7. Morphology of the SCARA robot manipulator

The robotic laboratory consists of different didactic equipment to practice control and manipulation tasks; *(i) Object selection*, the user can control the robot manipulator to select the object color type (red or blue) from module A and place in the empty spaces of module B, see Fig. 8 (a). (*ii) Movement synchronization*, user-defined path is established for the object movement by the robot manipulator and the conveyor belt, see Fig. 8 (b). Another application is the synchronization with the rotary table, the robot performs autonomous movements to pick up objects from a module and place them on the table during synchronized times, see Fig. 8 (c).



(a) Object handling (b) Autonomous movements



(c) Synchronization of movements - rotary table

Fig. 8. Laboratory practices

Industrial Process. An industrial environment was performed, focused on the assembly of electronic cards, see Fig. 9. The application aims to adapt users to real industrial environments, allowing the identification of the processes to be controlled through the immersion and interaction of the operator in environments of work, following safety rules and procedures.

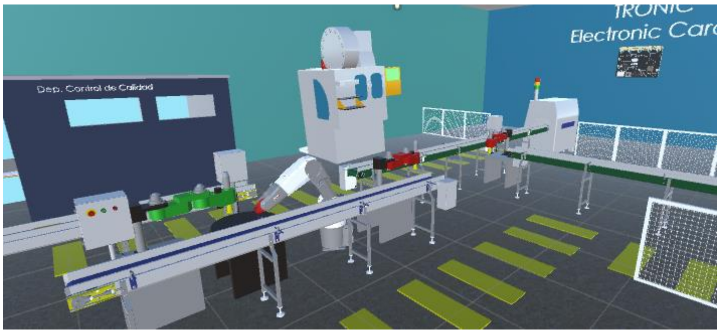
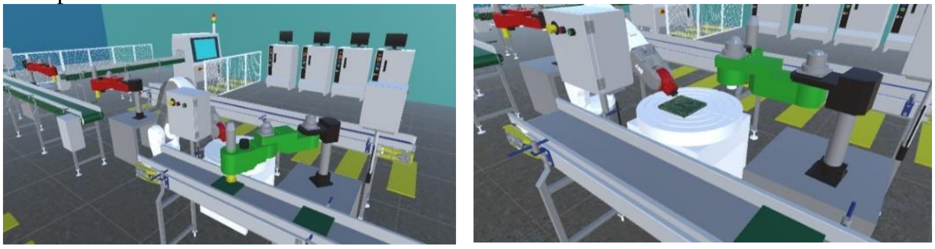
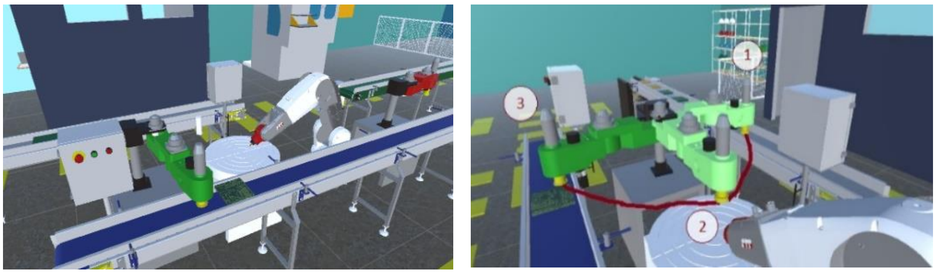


Fig. 9. Industrial process of electronic cards assembly

In the machining area, the SCARA robot manipulator handles the cards through a set path. Fig. 10 (a). It presents the entry of the cards throughout a conveyor belt, therefore, the robot end effector picks it up and places it on the table. In the work area, it is possible to see the machining performed by the robotic arm and the final location of the machined card, see Fig. 10(b)(c). The Fig. 10(d). shows the autonomous movement of the robot manipulator.



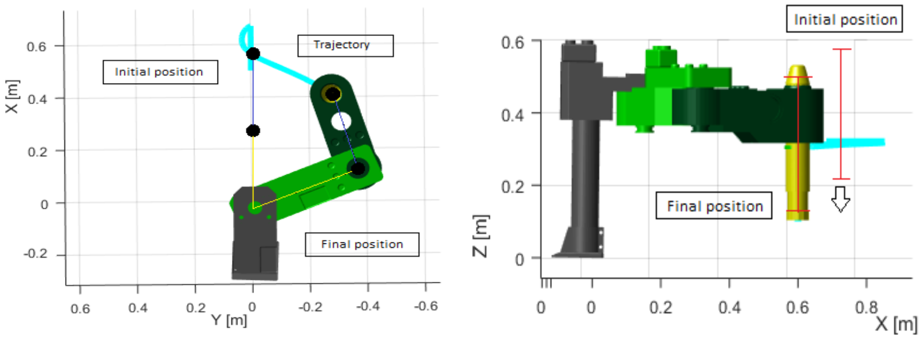
(a) Initial position of the manipulator (b) Electronic card machining



(c) Final position of the manipulator (d) Movement sequence

Fig. 10. Machining of electronic boards

Several experiments were developed to analyze the response of the controller. The Fig. 11(a) presents the movement of the joints q_1 and q_2 to set the end cap location to $x = 0.4$, $y = 0.3$ in the plane, using the values of $q_1 = -1.2$ and $q_2 = 1.5$. On the other hand, Fig. 11(b) presents the displacement made by the end effector to establish a position on the z axis of 0.1. This indicates that the control action applied to the robot manipulator model is working properly. Fig. 12. Indicates the SCARA robot manipulator's evolution time of control errors.



(a) Path described by the end effector (b) Movement in the z axis

Fig. 11. Interpretation of trajectories of the SCARA robot manipulator

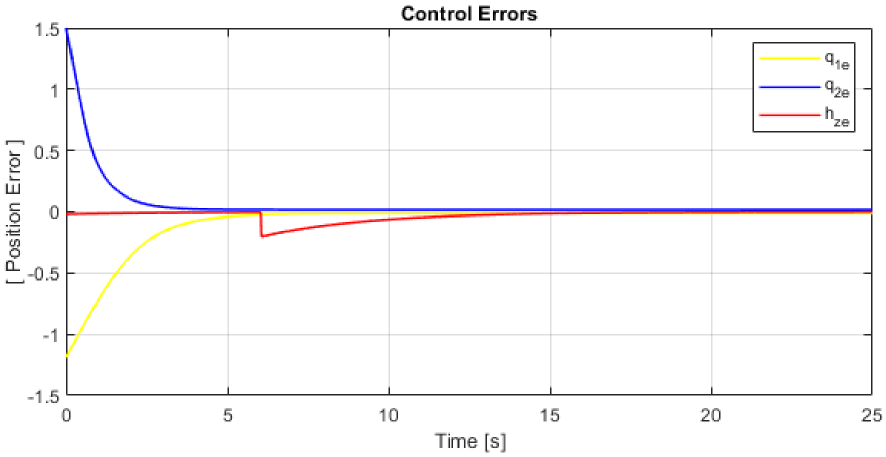


Fig. 12. Evolution of control errors of the SCARA manipulator robot

Finally, to establish the efficiency of the application, a functional test is performed on technicians in the robotics area. Each participant receives an introduction to both the classic robotics software and the new 3D virtual application. There is a protocol of activities at a didactic and industrial level to be developed by the participant. Fig. 13 shows the results obtained.

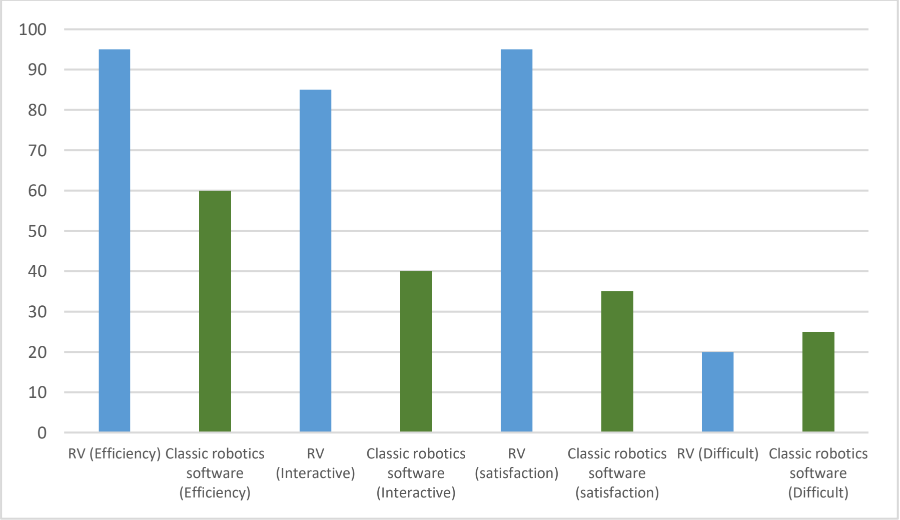


Fig. 13. Results of the evaluation of robotic systems

The results indicate that this application contributes efficiently and interactively to the operators' skills for the control and monitoring of robotic systems in a didactic way and in industrial manufacturing processes.

6 Conclusions

In this work, a 3D virtual simulator and a SCARA robot manipulator model were developed to analyze the behavior under the action of an established control law. This application considers a communication between MATLAB – Unity3D for the bilaterally exchange of information. The virtual simulator improves the technical training processes in the robotics area because it allows an immersion and interaction with the didactic and industrial teams of each virtual environment: *(i) Robotic Laboratory,* allows users to recognize the morphology of the SCARA robot manipulator, develop guided practices through conveyor belts, rotary tables, and object modules. While in (*ii) Industrial Process,* it presents manufacturing systems to become familiar with the industries in a real way, users can develop the trajectory control of robotic systems for different work areas, maintaining safety standards and processes in each one of them.

References

1. J. Kern y C. Urrea, "Modelamiento y simulación de un robot redundante de tipo manipulador SCARA", Revista Latinoamericana de Educación Física, vol. 5 (4), ISSN 1870-9095, 2011.
2. V. Andaluz, D. Castillo, R. Miranda y J. Alulema, "Virtual Reality Applied to Industrial Processes", Springer, Cham, vol. 10324, ISBN 978-3-319-60922-5, pp. 59-74, 2017.
3. A. Tibaduiza, I. Amaya y S. Rodríguez, "Implementación de un control fuzzy para el control cinemático directo en un robot manipulador", Ingeniare. Revista chilena de ingeniería, vol. 19 (3), ISSN 0718-3305, pp. 312-322, 2011.
4. M. G. ,. M. M. K. T. Amir Amiri, "Modelado y control de un robot SCARA utilizando la teoría de retroalimentación cuantitativa", Revista de Ingeniería de Sistemas y Control, vol. 223 (7). https://doi.org/10.1243/09596518JSCE733, pp. 919-928, 2009.
5. I. González, F. Torres y J. Mota, "Mobile Learning para Robótica a través de Aplicación Móvil usando Android - Java - Matlab", IX Congreso Internacional de Computación 2019, 2019.
6. M. Galicki, "Finite-time trajectory tracking control in a task space of robotic manipulators", Automatica, vol. 67, ISSN 0005-1098, pp. 165-170, 2016. https://doi.org/10.1016/j.automatica.2016.01.025
7. K. Zeng, G. Luo, M. Wu y X. Lai, "Trajectory tracking for a 3-DOF robot manipulator based on PSO and adaptive neuro-fuzzy inference system", 35th Chinese Control Conference (CCC), IEEE, ISSN: 1934-1768. https://doi.org/10.1109/ChiCC.2016.7553213, 2016.
8. J. Ortiz, A. Erazo, C. Carvajal, J. Pérez, L. Proaño, F. Silva y V. Andaluz, "Modeling and Kinematic Nonlinear Control", Springer, Cham, vol. 50 (11), pp. 87-95, 2017. https://doi.org/10.1007/978-3-319-60867-9
9. J. Shi, H. Liu y N. Bajcinca, "Robust control of robotic manipulators based on integral sliding mode", Journal of Control, vol. 81, nº 10, pp. 1537-1548, 2008. https://doi.org/10.1080/00207170701749881
10. V. Andaluz, F. Chicaiza, C. Gallardo, W. Quevedo, J. Varela, J. S. Sánchez y O. Arteaga, "Simulador Unity3D-MatLab en tiempo real para aplicaciones robóticas", Realidad

aumentada, realidad virtual y gráficos por computadora, LNCS 9768, Springer, Cham, pp. 246-263, 2016. https://doi.org/10.1007/978-3-319-40621-3_19

11. G. Guoyou, J. Chunsheng, C. Tao, H. Chun, W. Lina y L. Zifan, "Research on Robot Position Control Based on Improved PD algorithm", Conferencia Anual Académica Juvenil 2019 de la Asociación China de Automatización (YAC), ISBN electrónico: 978-1-7281-3936-4, https://doi.org/10.1109 / YAC.2019.8787668, 2019.
12. A. Visioli y G. Legnani, "On the trajectory tracking control of industrial SCARA robot manipulators", IEEE Transactions on Industrial Electronics, vol. 49, pp. 224 - 232, 2002. https://doi.org/10.1109/41.982266
13. M. Taylan y L. Canan, "Mathematical modelling, simulation and experimental verification of a scara robot", Simulation Modelling Practice and Theory, vol. 13 (3), pp. 257-271, 2005. https://doi.org/10.1016/j.simpat.2004.11.004
14. M. Salman, F. Ionescu y R. Taha, "Kinematic Modeling and Simulation of a SCARA Robot by Using Solid Dynamics and Verification by MATLAB/Simulink", European Journal of Scientific Research, vol. 37 (3), ISSN 1450-216, pp. 388-405, 2009.
15. M. Shariatee, A. Akbarzadeh, A. Mousavi y S. Alimardani, "Design of an economical SCARA robot for industrial applications", Second RSI/ISM International Conference on Robotics and Mechatronics (ICRoM), Electronic ISBN:978-1-4799-6743-8, 2014. https://doi.org/10.1109/ICRoM.2014.6990957
16. S. Yamacli y H. Canbolat, "Simulación de un robot SCARA con PD y controladores de aprendizaje", Práctica y teoría del modelado de simulación, vol. 16 (9), pp. 1477-1487, 2008. https://doi.org/10.1016/j.simpat.2008.08.004