UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

Meaningful Learning Processes of Service Robots for Tracking Trajectories through Virtual Environments.

Tercero Gualpa, Jhonatan Wladimir

Departamento de Eléctrica, Electrónica y Telecomunicaciones

Carrera de Ingeniería en Electrónica e Instrumentación

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

Mgs. Ortiz Moreano, Jessica Sofía

23 de enero del 2024

Latacunga

Meaningful Learning Processes of Service Robots for Tracking Trajectories through Virtual Environments.

Jhonatan W. Tercero and Jessica S. Ortiz
Universidad de las Fuerzas Armadas ESPE, L Sangolquí-Ecuador
{jwtercero,jsortiz4}@espe.edu.ec

Abstract. This paper is devoted to the study and control of an aerial manipulator robot (AMR) to perform tracking tasks autonomously, in order to apply LQR linear system control algorithms and application methods in a 3D virtual environment. Prior to obtaining a linearized kinematic model of the robotic systems, which allows to perform missions that require both navigation and manipulation capabilities in partially structured areas or environments. Through the use of the advanced control algorithm, a virtualized environment was developed in a 3D simulator for educational processes as a form of testing, which allows to evaluate the movement and evolution of the control errors, both for verification and visualization of the RMA behavior. Finally, the stability and robustness of the proposed RMA control is tested and experimentally analyzed using the DJI Matrice 600 Pro UAV tethered to an anthropomorphic 3DOF robotic arm. Therefore, these results are exposed and discussed to validate the proposed controller and ensure its correct operation.

Keywords: Control, linearization, stability and robustness.

1 Introduction

In recent years, robotics research has fostered the development of multiple applications focused on improving and assisting humans. Its development has expanded in several areas, in order to generate sustainable solutions in the fields of industry, education, medicine, training, military fields, among others [1, 2]. Therefore, robotics is considered as a support and service tool outside of industry, giving rise to service robotics. Service robots are related to the interaction and collaborative work between robots and humans, in partially structured environments [2]. In this context, service robotics is intended to provide support for activities that may be considered risky for humans. Preventing users from being affected by activities in hostile environments, unfamiliar environments and/or repetitive activities that may affect the user [3]. The International Federation of Robotics (IFR) defines a service robot as a robot that performs useful tasks for humans or equipment to recognize and manipulate objects in different locations and from different heights, have locomotion on different types of surfaces, interact with a human, distinguish different people, among others [4, 5]. Since there are several areas of work, they have had to adapt to these new changes, the educational sector is one of the most influential factors in the process of teaching and researching these service robots. [6, 7].

Prior to service robot research, a wide variety of them have been found, such as domestic service robots, professionals and humanoids, these robots are systems designed to perform dangerous tasks as well as repetitive activities or Jobs, Therefore, a professional service robot which emphasizes the RMA is considered as part of the research, which consists of a robotic arm anchored on a UAV aerial mobile platform [8]. UAVs can perform completely autonomous tasks in unstructured spaces and one of their most important applications is navigation and localization, but they are limited to perform more complex tasks requiring greater precision [9, 10], Similarly, the anthropomorphic 3DOF robotic arm has a very versatile configuration, that provide skills when performing tasks in locations that are difficult to access or at high risk to humans, the solution proposed for this type of scenario is an autonomous system that allows the user to perform physical work remotely in a safe mode [11], by analyzing the limitations of each robotic system and combining the advantages of the UAV with the versatility of the anthropomorphic 3DOF robotic arm, this system can perform reprogrammable handling tasks such as gripping, placing and pushing objects in various tasks [12]. They range from conventional serial manipulators to manipulators that take advantage of limited displacements of the center of mass and even multiple Degrees of Freedom (DOF) [13]. This robotic system is generally characterized by a high degree of redundancy, combining the manipulation capability of a fixed-base anthropomorphic 3DOF robotic arm with the navigation of a UAV and the localization of various spaces [13].

For the execution of various tasks with RMAs, multiple control strategies have been developed to solve positioning tasks, path tracking and trajectory tracking [14]. Which is considered a controller for trajectory tracking, based on the direct kinematics of the UAV and the anthropomorphic 3DOF robotic arm, a control algorithm is determined for each system that considers as control point the operating end of the robotic arm and as inputs the UAV velocities and the angular velocities of the robotic arm [14].

Thus, an LQR linear system controller is proposed, for control error correction when autonomously executing RMA trajectory tracking tasks [15]. It is based on the elaboration of the advanced control algorithm on the basis of the mathematical models obtained from the RMA [16]. The proposed LQR controller is based on control error dynamics, obtained from the kinematic model of the UAV and the anthropomorphic 3DOF robotic arm; i.e., the linearized behavior of the control error during task execution is considered. [17, 18]. The LQR controller provides a gain to compensate for errors and disturbances during task execution, the stability of the proposed control algorithm is mathematically analyzed, in order to evaluate the evolution of control errors in the simulation and experimentation of the robotic system [19, 20].

Whereby graphics engines and 3D simulators are considered, for the validation of research and projects developed in virtual environments applied to new industrial or service processes [21, 22].

To validate the LQR linear system controller proposed in previous paragraphs, the following steps are considered: *i) Simulation,* a 3D simulator will be developed to evaluate the movement of the RMA and the evolution of control errors*; ii) Controller settings,* through the DJI flight software will be developed to adjust the parameters of the proposed control algorithm; *iii) Experimental tests,* several experimental tests will be carried out in order to validate the proposed control. In context, learning these robotic processes allows familiarization with the industrial environment, domestic among others. It helps to generate new knowledge in technological or research areas and thus helps the autonomy of education and learning of human beings. This article consists of five sections including the introduction; section 2 details the conceptualization of the virtual environment process as a 3D simulation; section 3 presents the kinematic model of the UAV and the anthropomorphic 3DOF robotic arm individually for the linearization of the system. The proposed control scheme for the RMA is thus specified. Section 4 shows the development of the virtual environment in Unity 3D software and analyzes the results in the experimentation and simulation part of the RMA virtual environment, which helps to test and validate the optimization of the LQR controller in the robotic system. Finally, section 5 presents the conclusions obtained from the research work.

2 Conceptualization of the Process

To strengthen the teaching-learning process of engineering students and as a simulation tool in RMA research, the development of virtual environments is proposed to understand, simulate and design control algorithms. It is important to mention that RMAs cover a wide variety of electrical and electronic devices. The prices of such a robot are excessive. These robotic systems can be realized by means of graphics engines, such as Unity 3D software that helps to check the mathematical model, design of controllers and their operation by means of three-dimensional animations. It also includes the experimentation process with the DJI Matrice 600 Pro robot anchored to an anthropomorphic 3DOF robotic arm.

2.1 Methodology

This section presents the methodology to be used as shown in Figure 1, shows the development stages, which allow the validation of the proposed control scheme in a 3D virtual simulator.

Figure 1 presents the methodology of an aerial manipulator robot, which consists of the following stages: *i) Mathematical model,* for obtaining the mathematical model, emphasis is placed on the parameters of conceptualization, formulation and validation, in order to achieve direct kinematics of the robotic system, allows to represent the handling and navigation characteristics and restrictions of the RMA; *ii) Control algorithm,* based on the dynamics of the control error, obtained from the kinematic model of the RMA; i.e., the linearized behavior of the control error during the execution of the autonomous path-following task is considered. For control error correction, an LQR controller will be implemented to provide a gain to compensate for errors and disturbances during task execution. The stability and robustness of the linear systems controller is analyzed, the same that will be used to carry out experimental trials. *iii) External resources,* a Unity 3D software graphics engine is used for virtualization, accessing the simulation of the DJI Matrice 600 Pro anchored an anthropomorphic 3DOF Robotic Arm, which will be modeled using CAD tools. *iv) 3D Simulation,* this block simulates the behavior of a robotic system interacting in a 3D virtual environment, and is oriented to meaningful learning. The virtualized environment allows validation of the proposed advanced control algorithm, a control based on the kinematic model to perform autonomous tracking tasks, for both the UAV and the anthropomorphic 3DOF robotic arm. In the same way, we implement environments that are as close to reality as possible, to showcase an interactive and immersive environment. Finally, there is the last stage; *v) Experimentation,* testing of the proposed control algorithm and validation of the kinematic model for both the real RMA and the virtualized RMA. In order to validate the use of virtual environments in meaningful learning processes for new proposals of control algorithms.

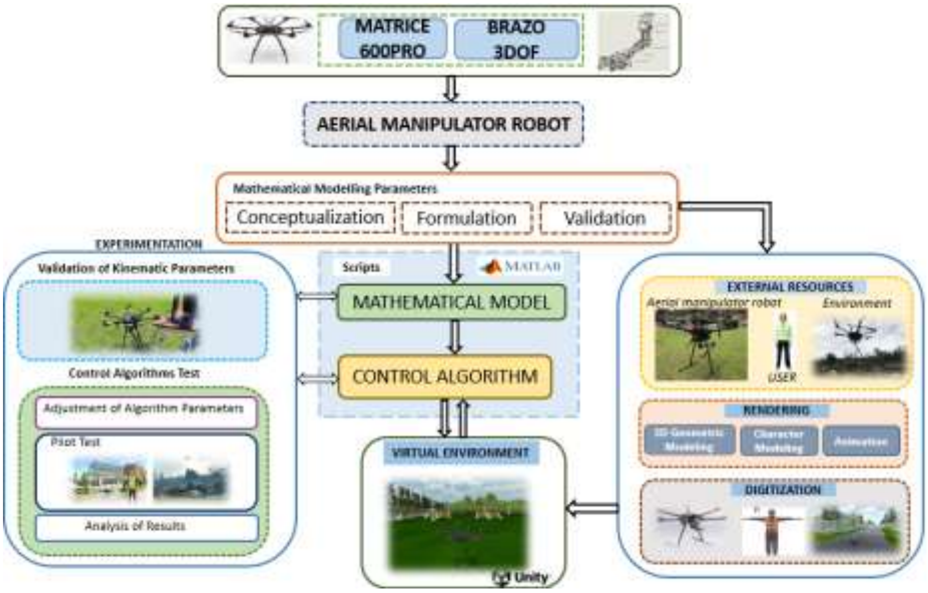

Fig. 1. Methodology for the optimal control algorithm and 3D virtualization of an RMA.

3 Aerial Manipulator Robot

The UAV and the anthropomorphic 3DOF robotic arm form a single robotic system described as RMA. The robotic system can perform a variety of tasks autonomously, such as handling objects with greater precision and performing large-scale work in spaces or places that are difficult for people to access.

3.1 Modeling

For the kinematic modeling of a robotic system, the position and velocity of the robot is analyzed, with the objective of following a trajectory or positioning itself at some reference point in the plane $R(X, Y, Z)$.

The type of UAV to which the kinematic model will be applied is a hexacopter (DJI Matrice 600 Pro) of six rotating propellers, with four operating speeds: up-down, frontal, lateral and rotational. It is considered the point of interest in the center of the UAV, to facilitate modeling.

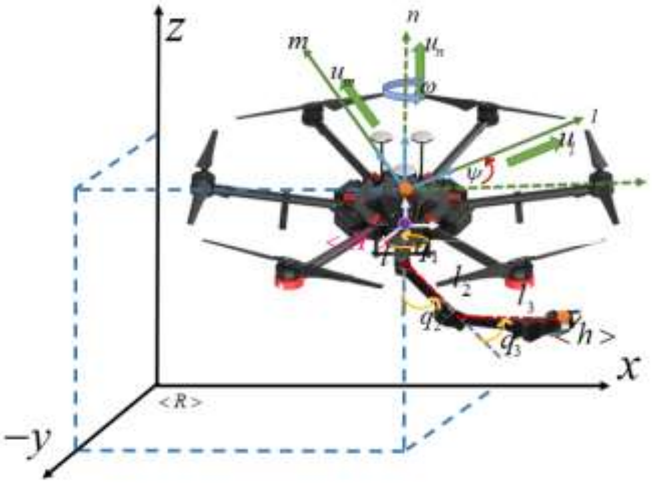

Fig. 2. Aerial Manipulator Robot

The RMA with its respective kinematic model provides the location of the operating endpoint $\mathbf{h_h}$ depending on the configuration of the robotic arm and the location of the UAV.

3.1.1. UVA modeling

The kinematic model of the UAV makes it possible to define the robot's velocities in the workspace $< R >$ in function of the robot maneuvering speeds defined in $< H >$, as follows. Control algorithms representing nonlinear kinematic models are generated.

$$\begin{bmatrix} \dot{r}_x \\ \dot{r}_y \\ \dot{r}_z \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 & 0 \\ sin(\psi) & cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_l \\ u_m \\ u_n \\ \omega \end{bmatrix} \tag{1}$$

$$\dot{\boldsymbol{r}}(t) = \boldsymbol{\Gamma}(\psi)\boldsymbol{u}(t) \tag{2}$$

where, $\dot{\boldsymbol{r}} = [\dot{r}_x \quad \dot{r}_y \quad \dot{r}_z \quad \dot{\psi}]^T$ is the velocity vector in the fixed reference plane $R(X, Y, Z)$, $\boldsymbol{u} = [u_l \quad u_m \quad u_n \quad \omega]^T$ presents the vector of the UAV maneuvering speeds, where u_l is the front control linear velocity;u_m is the lateral control linear velocity; $u_n$linear speed of elevation control and ω is the angular velocity with respect to the axis N of the reference system $H(L, M, N)$; and $\boldsymbol{\Gamma}$ is the Jacobian matrix of the system that converts the maneuvering velocities into velocities of the fixed reference frame, in this case is defined as $\boldsymbol{\Gamma} = diag([\boldsymbol{R}_{(\psi)} \quad 1])$. Where, $\boldsymbol{R}_{(\psi)}$ is a rotation matrix around the axis Z of the fixed reference system $< R >$.

3.1.2. Modeling of the Robotic Arm.

3DOF's robotic arm is anchored to the UAV's center of mass, as shown in Figure 2. The position of the operating end within the moving reference frame $H(L, M, N)$ is defined by $\boldsymbol{h_h} = [h_l, h_m, h_n]^T$. Therefore, the kinematic model of the 3DOF robotic arm is defined as:

$$\begin{bmatrix} \dot{h}_l \\ \dot{h}_m \\ \dot{h}_n \end{bmatrix} = \begin{bmatrix} -S_1(l_2S_2 + l_3S_{23}) & C_1(l_2C_2 + l_3C_{23}) & C_1l_3C_{23} \\ C_1(l_2S_2 + l_3S_{23}) & S_1(l_2C_2 + l_3C_{23}) & S_1l_3C_{23} \\ 0 & l_2S_2 + l_3S_{23} & l_3S_{23} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} \tag{3}$$

where, $C_\alpha = cos(\alpha)$; $S_\alpha = sin(\alpha)$;$C_{\alpha\beta} = cos(\alpha + \beta)$; and $S_{\alpha\beta} = sin(\alpha + \beta)$, l_1, l_2,l_3 represent the dimensions of the 3DOF robotic arm; q_1, q_2, q_3 are the rotation angles of each degree of freedom of the robotic arm. The kinematic model (3) can be represented in matrix form as:

$$\dot{\boldsymbol{h}}_{\boldsymbol{h}}(t) = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}}(t) \tag{4}$$

where, $\boldsymbol{J}(\boldsymbol{q}) \in R^{mxn}$ with $m = n$ represents the Jacobian matrix that transforms the velocities of the articulations $\dot{\boldsymbol{q}}(t)$ at operating end workspace speeds $\dot{\boldsymbol{h}}_{\boldsymbol{h}}(t)$.

3.2 Controller Design

For the implementation of an optimal controller for navigation and handling tasks, LQR control is considered for trajectory tracking, which allows tracking a virtual point that varies over time with reference to the fixed plane $< R >$, Figure 2. In context to perform the linearization of the RMA, it has been subdivided into two systems such as the UAV and the 3DOF robotic arm for their respective analysis.

3.2.1. UAV Controller

Prior to the nonlinear mathematical model, a virtual UAV (desired trajectory) to be followed is considered. Therefore, the virtual model is similar to the simplified kinematic model of the UAV (1).

$$\begin{bmatrix} \dot{r}_{xd} \\ \dot{r}_{yd} \\ \dot{r}_{zd} \\ \dot{\psi}_d \end{bmatrix} = \begin{bmatrix} cos(\psi_d) & -sin(\psi_d) & 0 & 0 \\ sin(\psi_d) & cos(\psi_d) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{ld} \\ u_{md} \\ u_{nd} \\ \omega_d \end{bmatrix} \tag{5}$$

$$\dot{\boldsymbol{r}}_{\boldsymbol{d}}(t) = \boldsymbol{\Gamma}_{\boldsymbol{d}}(\psi_{\boldsymbol{d}})\boldsymbol{u}_{\boldsymbol{d}}(t) \tag{6}$$

Where, $\dot{\boldsymbol{r}}_{\boldsymbol{d}} = \begin{bmatrix} \dot{r}_{xd} & \dot{r}_{yd} & \dot{r}_{zd} & \dot{\psi}_d \end{bmatrix}^T$is the desired velocity vector, $\boldsymbol{u}_{\boldsymbol{d}} = [u_{ld} \quad u_{md} \quad u_{nd} \quad \omega_d]^T$ presents the desired maneuvering velocity vector of the UAV. The variables u_{ld}, $u_{md}$and u_{nd} are desired linear velocities that are oriented on the axes of the UAV structure with respect to the system $< H >$, ω_d is considered as the desired rotational speed, and $\boldsymbol{\Gamma}_{\boldsymbol{d}}$ is the matrix that transforms the desired maneuvering velocities into the desired velocities over the fixed reference frame.

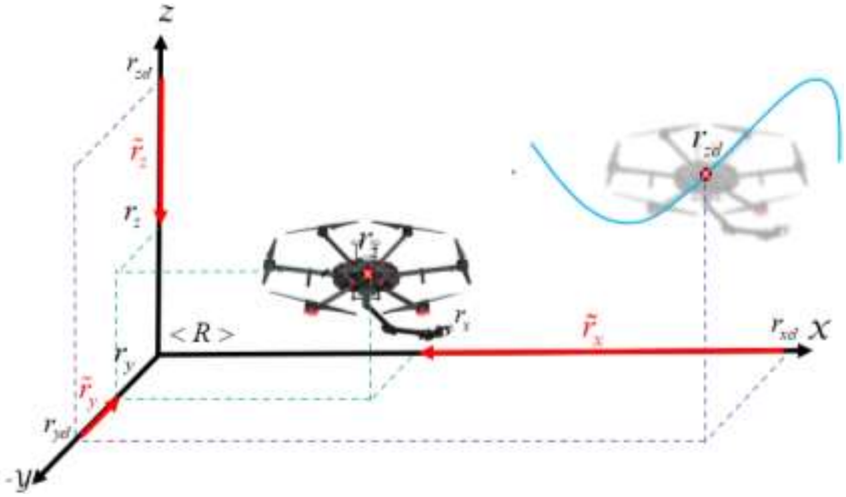

Fig. 3. Dynamic error for UAV trajectory tracking.

The position error of the UAV in reference to the fixed one Figure 3, is determined as follows.

$$\tilde{\boldsymbol{r}} = \boldsymbol{r}_{\boldsymbol{d}} - \boldsymbol{r} \tag{7}$$

$$\tilde{\boldsymbol{r}} = [r_{xd} - r_x \quad r_{yd} - r_y \quad r_{zd} - r_z \quad \psi_d - \psi]^T \tag{8}$$

Where, $\tilde{\boldsymbol{r}} = \begin{bmatrix} \tilde{r}_x & \tilde{r}_y & \tilde{r}_z & \tilde{\psi} \end{bmatrix}^T$ is the vector of errors in$< R >$. Now, for the controller, we obtain the dynamics of the error on the moving reference system$< H >$, therefore, the error in the mobile system is defined as:

$$\tilde{\boldsymbol{r}}_{\boldsymbol{h}} = \boldsymbol{\Gamma}^{-1}(\psi)\tilde{\boldsymbol{r}} \tag{9}$$

Where,$\tilde{\boldsymbol{r}}_{\boldsymbol{h}} = [\tilde{r}_l \quad \tilde{r}_m \quad \tilde{r}_n \quad \tilde{\psi}]^T$ is the error vector and $\boldsymbol{\Gamma}^{-1} = diag\left(\left[\boldsymbol{R}^{-1}_{(\psi)} \quad 1\right]\right)$ is the matrix that transforms the errors from the moving frame to the fixed frame. Now, in order to obtain the dynamics of the error, we derive (9) with respect to time, resulting:

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = \dot{\boldsymbol{\Gamma}}^{-1}\tilde{\boldsymbol{r}} + \boldsymbol{\Gamma}^{-1}\dot{\tilde{\boldsymbol{r}}} \tag{10}$$

Then, substituting (10) $\dot{\tilde{\boldsymbol{r}}} = \dot{\boldsymbol{r}}_{\boldsymbol{d}} - \dot{\boldsymbol{r}}$ y $\tilde{\boldsymbol{r}} = \boldsymbol{\Gamma}\tilde{\boldsymbol{r}}_{\boldsymbol{h}}$ results.

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = \dot{\boldsymbol{\Gamma}}^{-1}\boldsymbol{\Gamma}\tilde{\boldsymbol{r}}_{\boldsymbol{h}} + \boldsymbol{\Gamma}^{-1}(\dot{\boldsymbol{r}}_{\boldsymbol{d}} - \dot{\boldsymbol{r}}) \tag{11}$$

By substituting in $\dot{\boldsymbol{r}}_{\boldsymbol{d}}$ y $\dot{\boldsymbol{r}}$ in (11) is obtained:

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = \dot{\boldsymbol{\Gamma}}^{-1}\boldsymbol{\Gamma}\tilde{\boldsymbol{r}}_{\boldsymbol{h}} + \boldsymbol{\Gamma}^{-1}(\boldsymbol{\Gamma}_{\boldsymbol{d}}\boldsymbol{u}_{\boldsymbol{d}} - \boldsymbol{\Gamma}\boldsymbol{u}) \tag{12}$$

The mathematical expression (12) is expressed as follows.

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = \dot{\boldsymbol{\Gamma}}^{-1}\boldsymbol{\Gamma}\tilde{\boldsymbol{r}}_{\boldsymbol{h}} + \boldsymbol{\Gamma}^{-1}\boldsymbol{\Gamma}_{\boldsymbol{d}}\boldsymbol{u}_{\boldsymbol{d}} - \boldsymbol{u} \tag{13}$$

Written in a matrix form it results:

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = -\dot{\psi} \begin{bmatrix} sin(\psi) & -cos(\psi) & 0 & 0 \\ cos(\psi) & sin(\psi) & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 & 0 \\ sin(\psi) & cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \tilde{r}_l \\ \tilde{r}_m \\ \tilde{r}_n \\ \tilde{\psi} \end{bmatrix} +$$

$$\begin{bmatrix} cos(\tilde{\psi}) & sin(\tilde{\psi}) & 0 & 0 \\ -sin(\tilde{\psi}) & cos(\tilde{\psi}) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{ld} \\ u_{md} \\ u_{nd} \\ \omega_d \end{bmatrix} - \begin{bmatrix} u_l \\ u_m \\ u_n \\ \omega \end{bmatrix} \tag{14}$$

Where, $\tilde{\psi} = \tilde{\psi}_d - \tilde{\psi}$ is the orientation error. Now, let's consider a perfect follow-up, so if $\tilde{\psi} \approx 0$ it turns out that $cos(\tilde{\psi}) \approx 1$ and $sin(\tilde{\psi}) = \tilde{\psi}$. In this way, the dynamics of the error is linearized, resulting in:

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = -\dot{\psi} \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{r}_l \\ \tilde{r}_m \\ \tilde{r}_n \\ \tilde{\psi} \end{bmatrix} + \begin{bmatrix} 1 & \tilde{\psi} & 0 & 0 \\ -\tilde{\psi} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{ld} \\ u_{md} \\ u_{nd} \\ \omega_d \end{bmatrix} - \begin{bmatrix} u_l \\ u_m \\ u_n \\ \omega \end{bmatrix} \tag{15}$$

As $\omega = \dot{\psi}$ and solving (15) results:

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = \begin{bmatrix} \omega\tilde{r}_m + u_{ld} - u_{md}\tilde{\psi} - u_l \\ -\omega\tilde{r}_l + u_{ld}\tilde{\psi} + u_{md} - u_m \\ u_{nd} - u_n \\ \omega_d - \omega \end{bmatrix} \tag{16}$$

Now, we obtain a linear model of the error behavior, taking into account that the variation of the velocity errors can be written as $u_{id} - u_i = \Delta u_i$ for $i = l, m, n$; and $\omega_d - \omega = \Delta\omega$, results:

$$\begin{bmatrix} \dot{\tilde{r}}_l \\ \dot{\tilde{r}}_m \\ \dot{\tilde{r}}_n \\ \dot{\tilde{\psi}} \end{bmatrix} = \begin{bmatrix} 0 & \omega & 0 & -u_{md} \\ -\omega & 0 & 0 & u_{ld} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{r}_l \\ \tilde{r}_m \\ \tilde{r}_n \\ \tilde{r}_\psi \end{bmatrix} + I \begin{bmatrix} \Delta u_l \\ \Delta u_m \\ \Delta u_n \\ \Delta\omega \end{bmatrix} \tag{17}$$

The linearized system can be represented as a state model:

$$\dot{\tilde{\boldsymbol{r}}}_{\boldsymbol{h}} = A\tilde{\boldsymbol{r}} + B\Delta\boldsymbol{u} \tag{18}$$

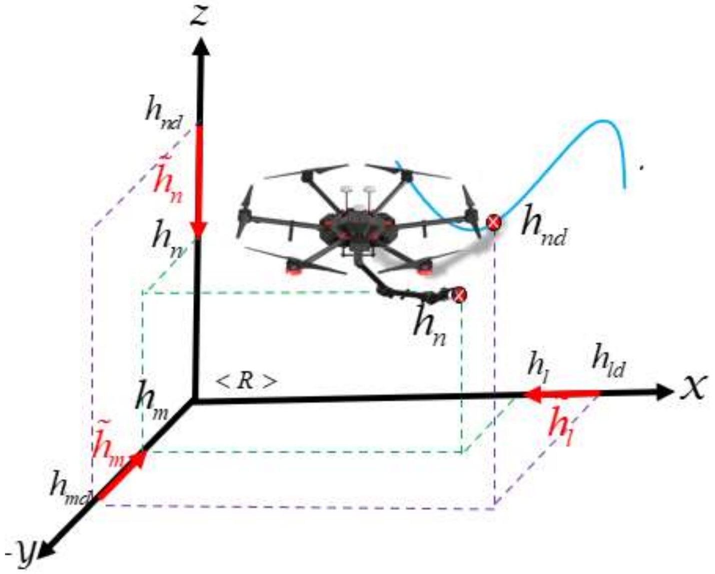

Fig. 4. Dynamic error for trajectory tracking of the 3DOF robotic arm

3.2.2. 3DOF robotic arm controller

Transforming the kinematics of the robotic arm from the moving reference frame to the fixed reference frame, by means of pre-multiplication of the rotation matrix $\boldsymbol{R}(\psi)$ a (4), results $\dot{\boldsymbol{h}}_{\boldsymbol{R}} = \boldsymbol{R}(\psi)(\boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}})$. Writing in a compact form result:

$$\dot{\boldsymbol{h}}_{\boldsymbol{R}}(t) = \boldsymbol{R}(\psi)\dot{\boldsymbol{h}}_{\boldsymbol{h}}(t) \tag{19}$$

Similarly, the position of the operating end in the fixed reference frame is obtained $\mathbf{h}_{\boldsymbol{R}} = \mathbf{R}(\psi)\mathbf{h}_{\mathbf{h}} = [h_x, h_y, h_z]^T$. Like the UAV, it works with error;

$$\tilde{\boldsymbol{h}}_{\boldsymbol{R}} = \boldsymbol{h}_{\boldsymbol{Rd}} - \boldsymbol{h}_{\boldsymbol{h}} \tag{20}$$

Where, $\boldsymbol{h}_{\boldsymbol{Rd}}$ is the desired virtual position of the robotic arm measured in the fixed reference frame. Then, similar to the UAV the desired trajectory can be represented as $\dot{\boldsymbol{h}}_{\boldsymbol{Rd}}(t) = \boldsymbol{R}_{\boldsymbol{d}}(\psi_d)\dot{\boldsymbol{h}}_{\boldsymbol{hd}}(t)$, i.e., as a virtual robotic arm to follow. Now, representing the error of the operating endpoint in the mobile system as $\widetilde{\boldsymbol{h}}_{\boldsymbol{h}} = \boldsymbol{R}^{-1}\widetilde{\boldsymbol{h}}_{\boldsymbol{R}}$ and then deriving with respect to time results:

$$\dot{\widetilde{\boldsymbol{h}}}_{\boldsymbol{h}} = \dot{\boldsymbol{R}}^{-1}\widetilde{\boldsymbol{h}}_{\boldsymbol{R}} + \boldsymbol{R}^{-1}\dot{\widetilde{\boldsymbol{h}}}_{\boldsymbol{R}} \tag{21}$$

Now, substituting the error in fixed system $\widetilde{\boldsymbol{h}}_{\boldsymbol{R}} = \widetilde{\boldsymbol{R}}\boldsymbol{h}_{\boldsymbol{h}}$ and the derivative of the error $\dot{\widetilde{\boldsymbol{h}}}_{\boldsymbol{R}} = \dot{\boldsymbol{h}}_{\boldsymbol{Rd}} - \dot{\boldsymbol{h}}_{\boldsymbol{h}}$ in (20), results:

$$\dot{\widetilde{\boldsymbol{h}}}_{\boldsymbol{h}} = \dot{\boldsymbol{R}}^{-1}\boldsymbol{R}\widetilde{\boldsymbol{h}}_{\boldsymbol{h}} + \boldsymbol{R}^{-1}\left(\dot{\boldsymbol{h}}_{\boldsymbol{Rd}} - \dot{\boldsymbol{h}}_{\boldsymbol{h}}\right) \tag{22}$$

As $\dot{\boldsymbol{h}}_{\boldsymbol{Rd}}(t) = \boldsymbol{R}_{\boldsymbol{d}}(\psi_d)\dot{\boldsymbol{h}}_{\boldsymbol{hd}}(t)$ equation (22), results:

$$\dot{\widetilde{\boldsymbol{h}}}_{\boldsymbol{h}} = \dot{\boldsymbol{R}}^{-1}\boldsymbol{R}\widetilde{\boldsymbol{h}}_{\boldsymbol{h}} + \boldsymbol{R}^{-1}\boldsymbol{R}_{\boldsymbol{d}}\dot{\boldsymbol{h}}_{\boldsymbol{hd}} - \boldsymbol{R}^{-1}\boldsymbol{R}\dot{\boldsymbol{h}}_{\boldsymbol{h}} \tag{23}$$

Where, $\boldsymbol{R}(\psi) \in R^{mxn}$ with $m = n$ represents the rotation matrix of the mobile system to the fixed reference system;$\widetilde{\boldsymbol{h}}_{\boldsymbol{h}} = \left[\tilde{h}_l, \tilde{h}_m, \tilde{h}_n\right]^T$ is the error in the mobile system. Substituting the values in (23) and applying the same procedure to the UAV, we obtain the error dynamics of the 3DOF arm as a function of the operating end velocities in the reference system$< H >$. As shown in Figure 4.

$$\dot{\widetilde{\boldsymbol{h}}}_{\boldsymbol{h}} = \begin{bmatrix} \omega\tilde{h}_m - \dot{h}_l + \dot{h}_{ld} - \dot{h}_{md} \\ -\omega\tilde{h}_l - \dot{h}_m + \dot{h}_{ld} - \dot{h}_{md} \\ \dot{h}_{nd} - \dot{h}_n \end{bmatrix} \tag{24}$$

Finally, writing the two linear systems obtained (17) and (24) results:

$$\dot{\tilde{\boldsymbol{e}}} = \begin{bmatrix} \dot{\tilde{r}}_l \\ \dot{\tilde{r}}_m \\ \dot{\tilde{r}}_n \\ \dot{\tilde{h}}_l \\ \dot{\tilde{h}}_m \\ \dot{\tilde{h}}_n \\ \dot{\tilde{r}}_\psi \end{bmatrix} = \begin{bmatrix} 0 & \omega & 0 & 0 & 0 & 0 & -u_{md} \\ -\omega & 0 & 0 & 0 & 0 & 0 & u_{ld} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \omega & 0 & \dot{h}_{l_d} \\ 0 & 0 & 0 & -\omega & 0 & 0 & \dot{h}_{m_d} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{r}_l \\ \tilde{r}_m \\ \tilde{r}_n \\ \tilde{h}_l \\ \tilde{h}_m \\ \tilde{h}_n \\ \tilde{r}_\psi \end{bmatrix} + \boldsymbol{I} \begin{bmatrix} \Delta u_l \\ \Delta u_m \\ \Delta u_n \\ \Delta\dot{h}_l \\ \Delta\dot{h}_m \\ \Delta\dot{h}_n \\ \Delta\omega \end{bmatrix} \tag{25}$$

Compactly written linear system of control errors in the moving frame, it turns out:

$$\dot{\tilde{\boldsymbol{e}}} = \boldsymbol{\Lambda}\tilde{\boldsymbol{e}} + \boldsymbol{\beta}\boldsymbol{\Delta v} \tag{26}$$

3.3 Control Scheme

The control of an RMA reference in the dimensions, weight and disturbances in the environment, is a complex problem if the dimensions and weight of the manipulator are relevant with respect to the weight of the platform.

The proposed control loop on both the UAV and the anthropomorphic 3DOF arm, is handled to find an entry of $\boldsymbol{\Delta v}$, The error also converges to zero in an optimal way. Whereby a linearized pre-error LQR control is proposed and the feedback gain results from the cost functional and the performance index.

$$\boldsymbol{J}_{\boldsymbol{F}} = \frac{1}{2}\int_0^\infty \left(\tilde{\boldsymbol{e}}_{(t)}^T Q\tilde{\boldsymbol{e}}_{(t)} + \boldsymbol{\Delta v}_{(t)}^T R\boldsymbol{\Delta v}_{(t)}\right)dt \tag{27}$$

$\boldsymbol{J}_{\boldsymbol{F}}$ refers to the cost function to be minimized; $\boldsymbol{Q}$ is a semidefinite quadratic matrix that carries the states of the system; $\boldsymbol{R}$ is a square matrix containing the error correction actions. The control law defined for trajectory tracking by velocities is expressed as follows.

$$\boldsymbol{u}_c = K\boldsymbol{r} + \boldsymbol{u}_{ref} \tag{28}$$

Where K, is the gain resulting from the minimization of the cost functional, the resulting loop feedback is as follows.

$$\begin{bmatrix} u_{c_l} \\ u_{cm} \\ u_{cn} \\ \dot{h}_{cl} \\ \dot{h}_{cm} \\ \dot{h}_{cn} \\ \omega_c \end{bmatrix} = \begin{bmatrix} \Delta u_{cl} \\ \Delta u_{cm} \\ \Delta u_{cn} \\ \Delta\dot{h}_{cl} \\ \Delta\dot{h}_{cm} \\ \Delta\dot{h}_{cn} \\ \Delta\omega_c \end{bmatrix} + \begin{bmatrix} u_{ref}\,cos(\psi) \\ u_{ref}\,sin(\psi) \\ \dot{z}_{ref} \\ \dot{h}_{cld} \\ \dot{h}_{cmd} \\ \dot{h}_{cnd} \\ \omega_c \end{bmatrix} \tag{29}$$

$\Delta\boldsymbol{u}_{\boldsymbol{c}} = [\Delta u_{cl} \quad \Delta u_{cm} \quad \Delta u_{cn} \quad \Delta\dot{h}_{cl} \quad \Delta\dot{h}_{cm} \quad \Delta\dot{h}_{cn} \quad \Delta\omega_c]^T$ are the optimal compensation velocities for path following error correction. The following graph shows the proposed control loop to follow the trajectory of a UAV with optimal gain.

Three optimal speeds for the robotic arm that compensate for error are determined by:

$$\dot{\boldsymbol{q}}_{\boldsymbol{c}} = \boldsymbol{J}^{-1}[\dot{h}_{cl} \quad \dot{h}_{cm} \quad \dot{h}_{cn}]^T \tag{30}$$

Where, $\dot{\boldsymbol{q}}_{\boldsymbol{c}} = [\dot{q}_{1c} \quad \dot{q}_{2c} \quad \dot{q}_{3c}]^T$ are the maneuverability speeds for the robotic arm joints. While u_{cl}, u_{cm}, u_{cn} y ω_c are the three linear velocities and angular velocity of maneuverability of the UAV respectively.

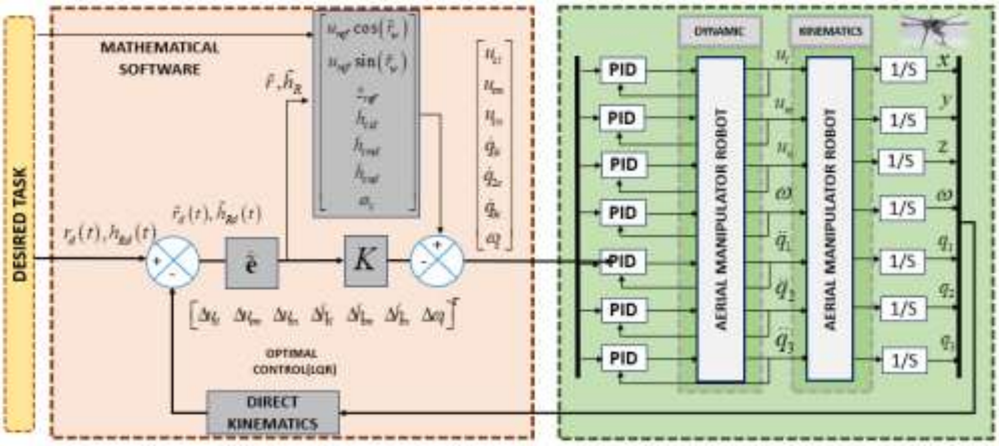

Fig. 5. Control Diagram of the Aerial Manipulator robot.

The proposed control algorithm for trajectory tracking, which includes the internal structure of the RMA closed loop, as shown in Figure 5. The scheme is based on a two-stage design. The first stage where the two systems of the UAV and the anthropomorphic arm are developed the optimal control of compensation and kinematics based on the structure of their models, respectively; It is worth mentioning that this stage is hosted in Matlab mathematical software for the implementation of an advanced control algorithm and simulation of RMA trajectory tracking. In the second stage, the mathematical models (kinematics) that allow describing the real movements of the aerial manipulator robot within the 3D simulator are housed; In addition, the simulator is equipped with an interactive menu for the user to change the task of the robotic system.

4 Analysis and Results.

The main objective is to demonstrate the stability and performance of the proposed LQR controller of the UAV and the anthropomorphic 3DOF robotic arm in a virtualized environment using the Unity 3D platform. Describes the process and the simulation stages that the RMA has undergone with its respective optimal control.

To validate the control of linear LQR systems, an experimental test with the Matrice 600 Pro anchored to a 3DOF anthropomorphic robotic arm is proposed [1], Figure 2. The control algorithm is solved in Matlab software and the data is sent and received through the communications architecture developed in (26).

Experiment: The experiment consists of defining a desired trajectory to validate the behavior of the RMA, several experimental tests are performed, consisting in defining several gains $\boldsymbol{Q}$ and $\boldsymbol{R}$ for the proposed cost function. The desired trajectory and parameters for the controller are defined in Table 1 for the experiment. Table 2 shows the gains established for the experimental tests, together with the functional value obtained during each test.

Table 1: Trajectory and controller parameters for the experiment.

Parameters	Values	Parameters	Values
r_{xd}	$\boldsymbol{8\,cos\left(0.3\left(\frac{t}{3}\right)\right)+5}$	$\boldsymbol{u_{ref}}$	$\sqrt{\dot{\boldsymbol{r}}_{\boldsymbol{xd}}^{\boldsymbol{2}}+\dot{\boldsymbol{r}}_{\boldsymbol{yd}}^{\boldsymbol{2}}}$
r_{yd}	$6\,sin\left(0.3\left(\frac{t}{3}\right)\right)+6$	$\boldsymbol{\omega_{ref}}$	$\frac{(\dot{r}_{xd}\ddot{r}_{yd}-\ddot{r}_{xd}\dot{r}_{yd})}{(\dot{r}_{xd}^2+\dot{r}_{yd}^2)}$
r_{zd}	$0.5\,sin\left(1.2\left(\frac{t}{3}\right)\right)+10$	u_{lr}	$(min(u_{ref})+max(u_{ref}))/2$
h_l	0.25	$\dot{h}_{cld}$	$\left(min\left(\sqrt{\dot{r}_x^2+\dot{r}_y^2}\right)+max\left(\sqrt{\dot{r}_x^2+\dot{r}_y^2}\right)\right)$ $/2$

h_m	$0.2\,cos(0.15t)$	$\dot{h}_{cmd}$	$\left(min\left(\sqrt{\dot{r}_x^2+\dot{r}_y^2}\right)+max\left(\sqrt{\dot{r}_x^2+\dot{r}_y^2}\right)\right)$ $/2$
h_n	$0.15\,cos(0.15t)-0.5$	$\dot{h}_{cnd}$	$\left(min\left(\sqrt{\dot{r}_x^2+\dot{r}_y^2}\right)+max\left(\sqrt{\dot{r}_x^2+\dot{r}_y^2}\right)\right)$ $/2$
$r_{\psi d}$	$tan^{-1}\left(\frac{\dot{r}_{yd}}{\dot{r}_{xd}}\right)$	$\boldsymbol{\omega_c}$	$(min(\omega_{ref})+max(\omega_{ref}))/2$

Table 2: Proposed gain values for $\boldsymbol{J_F}$ and performance index value obtained.

Gain	Values	Gain	Values	$\boldsymbol{J_F}$
$\boldsymbol{Q_1}$	$\boldsymbol{diag}[\mathbf{1\quad 1\quad 1\quad 5\quad 5\quad 5\quad 1}]$	$\boldsymbol{R_1}$	$\boldsymbol{diag}[\mathbf{600\quad 600\quad 600\quad 250\quad 250\quad 250\quad 25}$	1.4894×10^5
$\boldsymbol{Q_2}$	$\boldsymbol{diag}[\mathbf{10\quad 10\quad 10\quad 20\quad 20\quad 20\quad 1}$	$\boldsymbol{R_2}$	$\boldsymbol{diag}[\mathbf{500\quad 500\quad 500\quad 500\quad 500\quad 500\quad 50}$	1.5309×10^5
$\boldsymbol{Q_3}$	$\boldsymbol{diag}[\mathbf{20\quad 20\quad 20\quad 20\quad 20\quad 20\quad 2}$	$\boldsymbol{R_3}$	$\boldsymbol{diag}[\mathbf{80\quad 80\quad 80\quad 50\quad 50\quad 50\quad 50}]$	1.6539×10^4

4.1 Virtual environment

As a first part of the tests, the controller is pre-simulated, by simulating the controller in the Virtual Reality environment, the gains and parameters of the controller are adjusted. For the simulator, the initial position of the aerial manipulator robot is arbitrarily assigned, the desired trajectory is defined and then the simulation is executed in conjunction with Matlab and Unity, which has a virtual environment and the virtualized robot. Figure 6 shows the physical robot used for the experimental tests and the robot inside the virtual environment for the simulation.

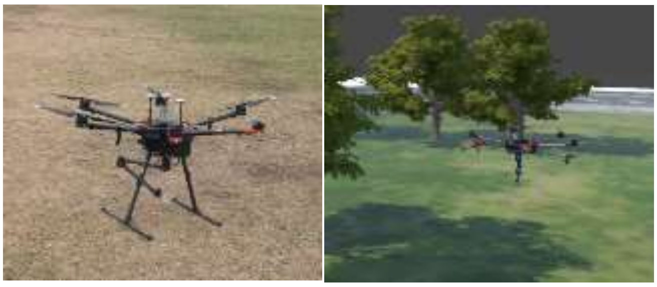

Fig. 6. Aerial Manipulator Robot (Real and Virtual).

The virtual reality scenario is similar to the one in which the experimental tests are performed. In Figure 7. The robot is presented interacting in the real environment and in the virtual environment. As the main objective is to follow a trajectory to accomplish that task autonomously.

Fig. 7. RMA physical and virtual environment.

4.2 Experimental tests

Once the controller gains have been adjusted, the next step is to perform real tests. The experiments are performed according to the parameters in Table 1 and Table 2. Three tests are performed to validate the operation of the proposed controller, each test consists of assigning different gains to the functional in order to obtain an optimum gain K to evaluate the robot's behavior. Figure 8 shows the movement of the robot in space, the graph corresponds to the experimental test with gain $\boldsymbol{Q_3}$ and $\boldsymbol{R_3}$ in the cost functional, which shows a better performance. It can be observed how the arm and the UAV follow the desired trajectory

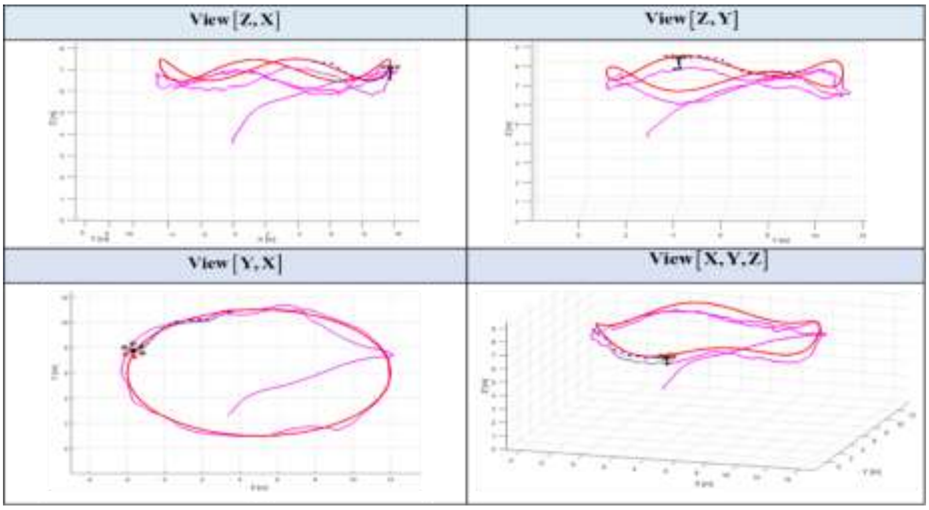

Fig. 8. Tests during trajectory tracking (Red line: Desired UAV trajectory. Magenta line: desired trajectory of the robotic arm. Blue line: Trajectory followed by the UAV. Green line: Tracked path of the robotic arm).

Figure 9 shows the evolution of UAV errors, it can be seen that the best performance is obtained with gains $\boldsymbol{Q_3}$ and $\boldsymbol{R_3}$, (yellow line) as the errors approach zero. While with the earnings of $\boldsymbol{Q_1}$ and $\boldsymbol{R_1}$ (blue line) in the functional, the errors are kept oscillating around zero. And with the earnings of $\boldsymbol{Q_2}$and $\boldsymbol{R_2}$ errors approach zero slowly, i.e. error correction is slow.

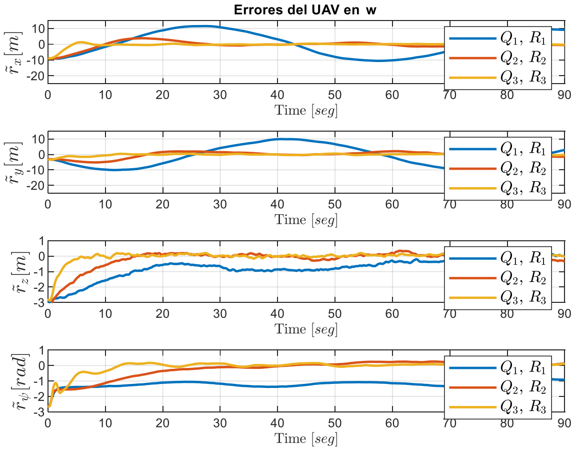

Fig. 9. UAV tracking control errors related to the map $< R >$

Finally, Figure 10. It presents the evolution of the robotic arm errors during the experimentation. Similarly, the error response with the values of $\boldsymbol{Q_3}$ and $\boldsymbol{R_3}$ are the best, since it keeps the control errors at zero.

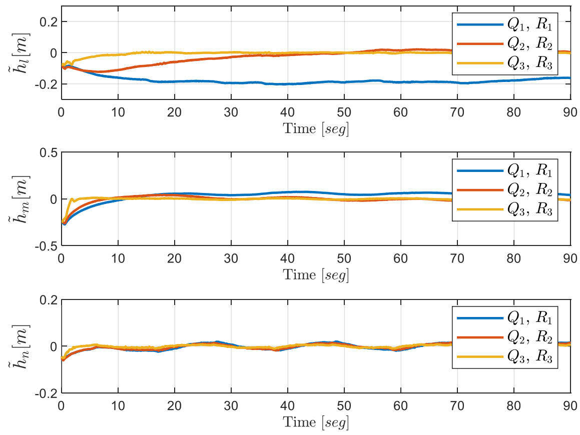

Fig. 10. 3DOF robotic arm control errors concerning the moving plane $<H>$.

5 Conclusions

A LQR controller was proposed based on the kinematic model of the aerial manipulator. The linearized model of the error behavior is done at the optimal operating points, i.e., how much the control errors approach zero. This allowed to obtain a linear model to use the LQR technique to obtain optimal gains for the controller. The functional gains $\boldsymbol{Q}$ and $\boldsymbol{R}$ of $\boldsymbol{J_F}$ allow to adjust the behavior of the robot, i.e., to assign weight to errors and control actions. As can be seen in the experiments the behavior of the robot changes according to these gains, allowing to obtain the desired behavior for error creation. In addition, the implementation of the virtual simulator allows to evaluate the proposed controller to adjust the controller parameters, this simulation developed not only allows to evaluate the behavior for this controller, the simulator is suitable for evaluating any type of proposed controller. And then perform the actual tests with the physical robot, thus avoiding causing damage to the robot during controller tuning.

Acknowledgments

The authors would like to thank the Universidad de las Fuerzas Armadas ESPE for their contribution to innovation, especially in the research project “Advanced Control of Unmanned Aerial Vehicles”, as well as the ARSI Research Group for their support in developing this work.

References

[1] C. P. Carvajal, V. H. Andaluz, F. Roberti, y R. Carelli, «Path-following control for aerial manipulators robots with priority on energy saving», *Control Engineering Practice*, vol. 131, p. 105401, feb. 2023, doi: 10.1016/j.conengprac.2022.105401

[2] N. Atanasoski, y K. Vora, «Surrogate humanity: race, robots, and the politics of technological futures», Duke University Press: *Google-Books-ID:* oyaJDwAAQBAJ,2019.

[3] N. Vallès-Peris y M. Domènec, «Roboticists’ Imaginaries of Robots for Care: The Radical Imaginary as a Tool for an Ethical Discussion», *Engineering Studies*, 12:3, 157-176, DOI: 10.1080/19378629.2020.1821695.

[4] International Federation Robotics, «The Impact of Robots on Productivity, Employment and Jobs», *IFR Press Releases*, April 2017.

[5] E. Vazquez, « Development of an object detection and manipulation system for a service robot. », Universidad autónoma de México, 2018. pp. 9-112.

[6] W. Zeng, «An empirical research on China’s policy for ICT integration in Basic Education from 1988 to 2021», *Educ. Technol. Res. Dev.* 2022, 70, 1059–1082.

[7] D. Morales-Doyle, «There is no equity in a vacuum: On the importance of historical, political, and moral considerations in science education», *Cult. Stud. Sci. Educ.* 2019, 14, 485–491.

[8] C. Carvajal, V. Andaluz, F. Roberti, R. Carelli, «Optimal Trajectory Tracking Control for a UAV Based on Linearized Dynamic Error», DOI: 10.1007/978-3-030-55789-8_8, September, 2020.

[9] W. Koch, R. Mancuso, R. West y A. Bestavros, «Reinforcement learning for UAV attitude control»,

ACM Trans. Cyber-Phys. Syst.2019, 3, 1–21.

[10] T. Bacelar, C. Cardeira, y P. Oliveira, «Cooperative load transportation with quadrotors». *In 2019 IEEE international conference on autonomous robot systems and competitions (ICARSC)* (pp. 1–6). http://dx.doi.org/10.1109/ICARSC.2019.8733619.

[11] E. Slawiñski, «Dual Coordination for Bilateral Teleoperation of a Mobile Robot with Time Varying Dela», *IEEE Latin America Transactions*, p. 100(1e), 2020.

[12] H.X. Pham, H.M.La, D. Feil-Seifer, L.V. Nguyen, «Autonomous UAV Navigation Using Reinforcement Learning», January 2018. *Available online:* http://arxiv.org/abs/1801.05086 (accessed on 27 April 2022).

[13] J M.Gómez-de-Gabriel, J M. Gandarias, F J. Pérez-Maldonado, F J. García-Nuñez, E J. FernándezGarcía y A J. García-Cerezo, «Methods for Autonomous Wristband Placement with a Search-and-Rescue Aerial Manipulator», *In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Madrid, Spain, 2018, October.

[14] V. H. Andaluz, C. P. Carvajal, J. A. Pérez y L. E. Proaño, «Kinematic nonlinear control of Aerial mobile manipulators». In Y. Huang, H. Wu, H. Liu, & Z. Yin (Eds.), *Conferencia en la*, *Intelligent robotics and applications*, 2017. pp. 740–749. Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-65298-6_66.

[15] J. S. Ortiz, A. P. Erazo, C. P. Carvajal, J. A Pérez, L. E Proaño y F. M Silva. «Modeling and kinematic nonlinear control of aerial mobile manipulators», In S. Zeghloul, L. Romdhane, M. A. Laribi (Eds.), *Mechanisms and machine science*, *Computational kinematics,2018.* pp. 87–95. Cham: Springer International Publishing, http://dx.doi.org/10.1007/978-3-319-60867-9_11.

[16] R. Kumar, M. Dechering, A. Pai, «Differential flatness based hybrid PID/LQR flight controller for complex trajectory tracking in quadcopter UAVs». *In: 2017 IEEE National Aerospace and Electronics Conference (NAECON).* https://doi.org/10.1109/naecon. 2017.8268755

[17] M. Mueller, N. Smith, B. Ghanem, «A Benchmark and Simulator for UAV Tracking», *Springer International Publishing:* Cham, Switzerland, 2017; Volume 9905.

[18] A. Suarez, A. Caballero, A. Garofano, P. J. Sanchez-Cuevas, G. Heredia, y A. Ollero, «Aerial manipulator with rolling base for inspection of pipe arrays», *IEEE Access*, *8*, 162516–162532. http://dx.doi.org/10.1109/ACCESS.2020.3021126, Conference Name: IEEE Access.

[19] H. Peng, F. Li, J. Liu, Z. Ju, «A symplectic instantaneous optimal control for robot trajectory tracking with differential-algebraic equation models», *IEEE Trans. Ind. Electron*. 67(5), 3819–3829 (2019). https://doi.org/10.1109/TIE.2019.2916390

[20] P. Cardenas Alzate, G. Velez, F. Mesa, «Optimum control using finite time quadratic linear regulator», *Contemp. Eng. Sci.* 11, 4709–4716. https://doi.org/10.12988/ces.2018. 89516

[21] R. Al-Azawi, M.S. Shakkah, «Embedding Augmented and Virtual Reality in Educational Learning Method», *Present and Future.* In Proceedings of the 2018 9th International Conference on Information and Communication Systems *(ICICS)*, Irbid, Jordan, 3–5 April.

[22] J.J. Roldán, E. Peña-Tapia, D. Garzón-Ramos, M. Garzón, J. del Cerro, A. Barrientos, Multi-robot systems, virtual reality and ROS: «Developing a new generation of operator interfaces». *Robot. Oper. Syst. Complet.* 3, 2019, 29–64.