Virtual Simulator for Collaborative Tasks of Aerial Manipulator Robots

Oscar A. Aguirre, Juan Carlos Ñacato, and Víctor H. Andaluz

Departamento de Eléctrica y Electrónica

Carrera de Ingeniería en Electrónica e Instrumentación

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

Tutor	Ing. Andaluz Ortiz Víctor Hugo, PhD.
Co-Tutor	Ing. Rivas Lalaleo David Raimundo, PhD

15 de febrero del 2022

Virtual Simulator for Collaborative Tasks of Aerial Manipulator Robots

Oscar A. Aguirre, Juan Carlos Ñacato, and Víctor H. Andaluz
Universidad de las Fuerzas Armadas ESPE
Sangolquí-Ecuador
{oaaguirre, jcnacato, vhandaluz1}@espe.edu.ec

Resumen — **Este artículo propone el desarrollo e implementación de un simulador virtual 3D para analizar y observar el comportamiento de robots manipuladores aéreos en tareas de control colaborativo, así como navegación autónoma y teleoperada a través de una sala multiusuario. El entorno virtual se desarrolló utilizando fotogrametría 3D que sirve como escenario real para probar las tareas de control colaborativo. La interfaz de comunicación entre el entorno y los robots manipuladores aéreos es el motor gráfico UNITY 3D, que tiene un enlace bidireccional con el software matemático científico MATLAB, utilizando bibliotecas (DLL) para retroalimentar y compensar errores de control. Finalmente, los resultados de la simulación serán presentados y discutidos para validar y probar las estrategias de control colaborativo propuestas.**

Palabras clave - robótica de servicios; multi usuario; control colaborativo, digitalización del entorno.

Abstract — **This article proposes the development and implementation of a 3D virtual simulator to analyze and observe the behavior of aerial manipulator robots in collaborative control tasks, as well as autonomous and tele-operated navigation through a multi-user room. The virtual environment was developed using 3D photogrammetry which serves as a real scenario to test the collaborative control tasks. The interface for communication between the environment and the aerial manipulator robots is the UNITY 3D graphic engine, which has a bi-directional link with the scientific mathematical software MATLAB, using libraries (DLLs) to provide feedback and compensate for control errors. Finally, the simulation results will be presented and discussed to validate and test the proposed collaborative control strategies.**

Keywords - service robotics; multi-user; collaborative control, environment digitalization.

I. INTRODUCTION

For a long time, robotics has evolved and experienced changes in order to solve multiple problems in different areas such as medicine, industry, military, etc. [1]. There is a field known as service robotics that introduces autonomous robots specifically designed to assist humans; the different services that these can offer depend largely on their mechanical design and there is a great variety of them such as terrestrial, aerial, marine, using legs, wheels, propellers, allowing their mobility in any environment and according to the need [2].

The great development of service robotics gave rise to the Unmanned Aerial Vehicles (UAV), having great impact on applications such as: civil works, security, agriculture and communications [3]; however, when it is required to fulfill more complex tasks that need both navigation and manipulation capabilities, the implementation of one or more robotic arms in the system is needed [4]. The mechanical connection between an aerial mobile platform (UAV) with one or multiple robotic arms is known as an aerial mobile manipulator robot (AMR) [5], which can be used in tasks such as welding, transport of objects, among others [6].

The rise of service robotics is the main factor in the development of intelligent control systems, with the aim of performing autonomous tasks efficiently [7] [8]. Today there are tele-operated AMR that allow users to perform object manipulation tasks in hard-to-reach places or hazardous environments [9]. Tele-operation could be defined as a remote operation or manipulation of a system to perform specific tasks in any environment with a high degree of reliability [10].

Nowadays, the advance of technology allows carrying out experiments with drones, however, failing a task can result in an economic loss during control tests, so there is software capable of simulating this interaction and training people who do not have experience in manippulation UAVs to carry out tasks without any risk with respect to the hardware [11]. According to Sterman [12], flight simulator applications have a high impact as a tool, since they transmit knowledge oriented towards constructionism, interactive learning or learning by doing. Some related research works that focus on the use of UAV simulators are Rao et al [13], who carried out a UAV simulator to exercise firefighters in the probable situation of a forest fire, so that they can visualize the affected areas. There are other works such as [13], [14] and [15] that use UAV simulators for emergencies or navigation in virtual environments of real environments.

In this context, this article focuses on the application of virtual reality technology as a tool for the simulation of collaborative control techniques of aerial manipulator robots in a shared environment (multi-user), virtualized, which allows interaction with the 3D model of the aerial manipulator robot, executing a trajectory tracking in a tele-operated way, in order to perform a stability and robustness analysis of the control strategy implemented.

II. VIRTUAL ENVIRONMENT

Figure 1 proposed diagram for the development of the virtual simulator is shown, which is implemented in the UNITY 3D graphic engine. The system is composed of a simulation environment built by photogrammetry and CAD modeling, SCRIPS programming to manipulate each of the objects in a multi-user room, input and output devices which allows the development of the simulated virtual environment.

Photogrammetry allows the development of the simulation environment with 3D models creating a virtual environment. Features are programmed to make the environment more real, such as the state of the weather and physical properties of the objects in the simulator.

The environment includes an aerial manipulator robot built for each of the users, which allows collaborative control tasks from the local station of each of the users connected to the same multiplayer room, with the aim of interacting in the same environment by moving objects from one place to another.

CAD modeling is defined by the 3D model of the aerial manipulator robot built in computer-aided design (CAD) software acquiring all its features and configurations of a real aerial manipulator robot, to carry out control tests and observe behavior in the simulated Environment.

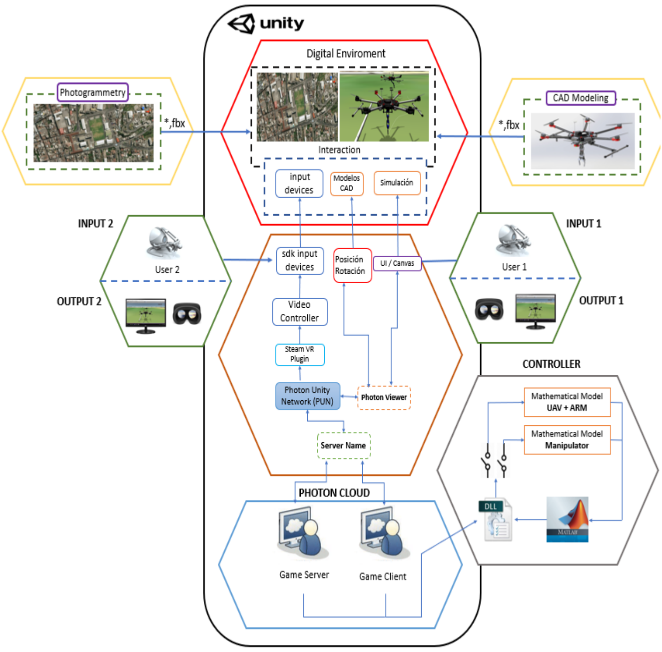

Figure 1. Proposed block diagram of the simulator

SCRIPTS are programmed by code blocks that allow the exchange of information between input and output devices connected to the system. For the communication between the input devices (Novit Falcon), native programming of the device in the Visual Basic (CHAID3D) software is used, for the execution of collaborative tasks of the aerial manipulator robots. The API Weather will allow implementing climate data to the simulated environment to influence the behavior of the collaborative control; finally a SCRIPT is programmed for the bilateral communication between the Matlab mathematical software and the 3D UNITY graphic engine, where the information is shared by means of a Dynamic-link-library (DLL) in which it implements the method of shared memory SM in the RAM.

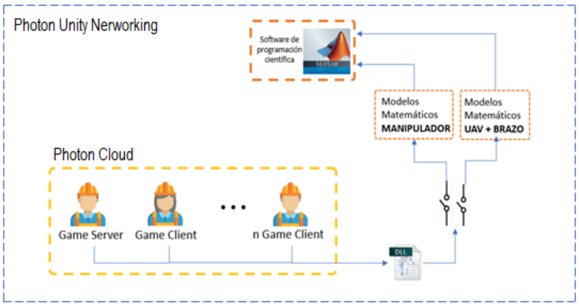

Figure 2. Photon Cloud Scheme

Photon Cloud is the communication interface between two computers through a multiplayer room as shown in the Fig. 2, the program that makes this link is PUN (Photon Unity Nerworking), it is native to the 3D Unity graphic engine, it has a client/server architecture where the first computer becomes the server of the multi-user platform and the players that connect to the room afterwards are the clients. The data is stored and shared through the cloud. Finally, the interaction between two totally independent devices will start to take place in order to perform the collaborative control tasks, where the control actions that are injected to the air handler will be performed through the SM, and also the velocities that enter will be received through the haptic device (Novit Falcon), to then send them to the Matlab mathematical software, thus obtaining the feedback to compensate the control errors.

III. CONTROL STRUCTURE

The scheme in Fig. 3 presents the block diagram implemented for the control structure. The operator can interact with the aerial manipulator robot through the haptic device, which sends the desired velocity to MatLab, at that moment the mathematical algorithm acts, so it sends the respective actions to the actuators to correct the position errors at the operating end, all this information is fed back from Unity's virtual environment.

The tele-operation scheme proposed in Fig. 3 is subdivided into*: i) Local Site*, made up of one or more human operators, who through haptic devices generate reference signals for each robot through haptic devices; *ii) Remote Site*, is made up of the interaction of multiple aerial manipulator robots executing collaborative tasks in virtual environments implemented in the Unity3D graphics engine; and finally *iii) Communication channel*, transfers the bilateral information between the local site and the remote site; in this work it is considered that there is no delay in sending and receiving information, *i.e.*, $\xi_1 = \xi_1 = 0\ [\mathrm{s}]$.

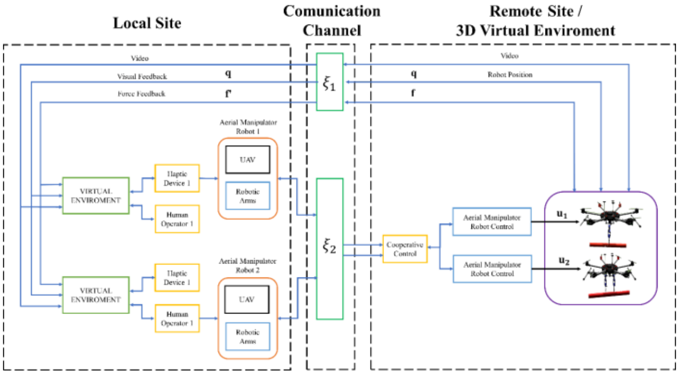

Figure 3. Remote operation of the aerial manipulator robot

A. *Kinematic Modeling*

The kinematic model of the aerial manipulator robot as show the Fig. 4 is composed of two 3DOF robotic arms mounted on a UAV. The UAV is guided through 3 linear velocities u_l, u_m, u_n ; where, u_l represents the velocity in forward direction, u_m is the velocity pointing to the left side, u_n is the velocity in the z -axis; Furthermore, there is an angular velocity ω , that represents the rotation in anti-clockwise direction with respect to the z -axis. While the maneuverability commands of each robotic arm are represented by an angular velocity for each link [16].

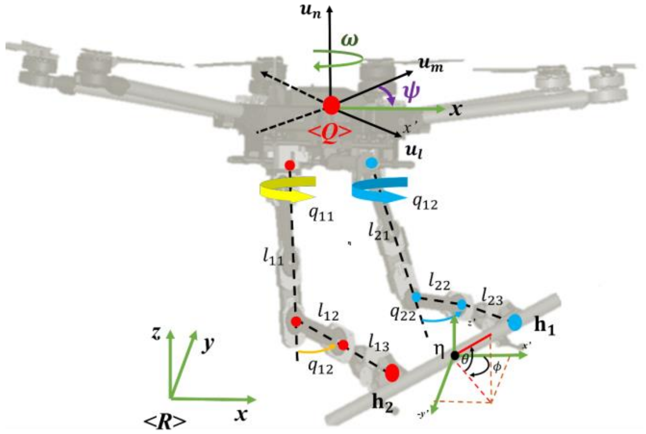

Figure 4. Aerial manipulator robot.

The kinematic model of the aerial manipulator robto is defined as follows:

$$\dot{\mathbf{h}}(t)=\mathbf{J}\left(\mathbf{q}_{\mathbf{q}},\mathbf{q}_{\mathbf{a1}},\mathbf{q}_{\mathbf{a2}}\right)\mathbf{v}(t) \tag{1}$$

where, $\mathbf{J}(\mathbf{q})\in\Re^{mxn}$ con $m=6$ y $n=10$ represents the Jacobian matrix that defines a linear mapping betweem the velocity vector of the mobile aerial manipulator $\mathbf{v}\in\Re^{n}$ where $\mathbf{v}(t)=[u_l \quad u_m \quad u_m \quad \omega \quad \dot{q}_{11} \quad \dot{q}_{12} \quad \dot{q}_{13} \quad \dot{q}_{21} \quad \dot{q}_{22} \quad \dot{q}_{23}]^{\mathrm{T}}$ and the velocity vector of the operating end $\dot{\mathbf{h}}\in\Re^{m}$ where $\dot{\mathbf{h}}(t)=[\dot{\mathbf{h}}_1 \quad \dot{\mathbf{h}}_2]^{\mathrm{T}}=[\dot{h}_{1x} \quad \dot{h}_{1y} \quad \dot{h}_{1z} \quad \dot{h}_{2x} \quad \dot{h}_{2y} \quad \dot{h}_{2z}]^{\mathrm{T}}$.

B. *Object Manipulation*

In this subsection, the strategy for manipulation of objects through two robotic arms on a UAV is presented , see Fig. 4.

To determine the kinematic model of the aerial manipulator robot, a virtual point is defined in the plane, $X-Y-Z$, between the midpoint of each end-effector of the robotic arms; the virtual point is defined as, $\mathbf{p}=[p_x \quad p_y \quad p_z]^{\mathrm{T}} = \frac{1}{2}\left[(h_{x1}+h_{x2}) \quad (h_{y1}+h_{y2}) \quad (h_{z1}+h_{z2})\right]^{\mathrm{T}}$ represents the position of it centroid in the inertial frame $<R>$.

The vector structure of the object is defined as $s=\left[d \quad \theta \quad \phi\right]^{\mathrm{T}}$, where d represents the distance between the position of the end-effector $\mathbf{h}_1$ y $\mathbf{h}_2$; θ y ϕ represents its orientation with respect to the global Y -axis and Z -axis, respectively on the inertial framework $<R>$ [17] [18].

$$\begin{cases} d=\sqrt{(h_{x2}-h_{x1})^2+(h_{y2}-h_{y1})^2+(h_{z2}-h_{z1})^2} \\ \theta=\tan^{-1}\left(\frac{h_z}{h_x}\right) \\ \phi=\tan^{-1}\left(\frac{h_y}{h_x}\right) \end{cases} \tag{2}$$

The point of interest of the system is represented in simplified form as $\boldsymbol{\eta}=\left[\mathbf{p} \quad \mathbf{s}\right]$.

Taking the time derivative of the forward and the inverse kinematics transformations we can obtain the relationship between the time variations of $\dot{\mathbf{h}}(t)$ and $\dot{\boldsymbol{\eta}}(t)$, represented by the Jacobian matrix $\mathbf{J}$, which is given by

$$\dot{\boldsymbol{\eta}}(t)=\boldsymbol{\Gamma}(\mathbf{h})\dot{\mathbf{h}}(t)$$

and in the inverse way is given by

$$\dot{\mathbf{h}}(t)=\boldsymbol{\Gamma}^{-1}(\boldsymbol{\eta})\dot{\boldsymbol{\eta}}(t).$$

In the other hand, the desired shape and position of the object to be manipulated is defined as $\boldsymbol{\eta}_d=\left[p_d \quad s_d\right]^T$ and its desired variations $\dot{\boldsymbol{\eta}}_d=\left[\dot{p}_d \quad \dot{s}_d\right]^T$. The formation error is defined $\tilde{\boldsymbol{\eta}}=\boldsymbol{\eta}_d-\boldsymbol{\eta}$ as taking the first derivative of time the following expression is obtained $\dot{\tilde{\boldsymbol{\eta}}}=\dot{\boldsymbol{\eta}}_d-\dot{\boldsymbol{\eta}}$. Hence, the proposed formation control law is defined as,

$$\dot{\mathbf{h}}(t)=\boldsymbol{\Gamma}^{-1}\left(\dot{\boldsymbol{\eta}}_d+\mathbf{K}\tanh\left(\tilde{\boldsymbol{\eta}}\right)\right) \tag{3}$$

where, $\mathbf{K}$ it's a diagonal matrix of positive gain. Now, to demonstrate stability, a Lyapunov-based controller is proposed. Defining a positively defined candidate function as, $\mathbf{V}(\tilde{\boldsymbol{\eta}})=\frac{1}{2}\tilde{\boldsymbol{\eta}}^T\tilde{\boldsymbol{\eta}}>0$. The closed-loop equation of the system is obtained by replacing (3) with (1).

$$\dot{\tilde{\boldsymbol{\eta}}} + \mathbf{K}\tanh\left(\tilde{\boldsymbol{\eta}}\right) = 0 \tag{4}$$

Introducing (4) in the time derivative of $\dot{\mathbf{V}}(\tilde{\mathbf{n}})$ is obtained,

$$\dot{\mathbf{V}}\left(\tilde{\boldsymbol{\eta}}\right) = -\tilde{\boldsymbol{\eta}}^T \mathbf{K}\tanh\left(\tilde{\boldsymbol{\eta}}\right) < 0\,. \tag{5}$$

As described, the equilibrium point is asymptotically stable, *i.e.* $\tilde{\boldsymbol{\eta}}\left(t\right) \to 0$ asymptotically.

Relaxing the assumption of perfect velocity following is considered a difference $\delta_{\tilde{\boldsymbol{\eta}}}(t)$ between desired and actual variations as $\delta_{\tilde{\boldsymbol{\eta}}} = \dot{\boldsymbol{\eta}}_d - \dot{\boldsymbol{\eta}}$. Equation (5) can be defined as, $\dot{\mathbf{V}}\left(\tilde{\boldsymbol{\eta}}\right) = \tilde{\boldsymbol{\eta}}^T\delta_{\tilde{\boldsymbol{\eta}}} - \tilde{\boldsymbol{\eta}}^T\mathbf{K}\tanh\left(\tilde{\boldsymbol{\eta}}\right)$. A sufficient condition to $\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}})$ be defined as negative is, $\left|\tilde{\boldsymbol{\eta}}^T\mathbf{K}\tanh\left(\tilde{\boldsymbol{\eta}}\right)\right| > \left|\tilde{\boldsymbol{\eta}}^T\delta_{\tilde{\boldsymbol{\eta}}}\right|$. For large values of $\tilde{\mathbf{n}}$ it can be considered that: $\mathbf{K}\tanh(\tilde{\boldsymbol{\eta}}) \approx \mathbf{K}$. While $\dot{\mathbf{V}}(\tilde{\boldsymbol{\eta}})$ will be of definite negative only if $\left\|\mathbf{K}\right\| > \left\|\delta_{\tilde{\boldsymbol{\eta}}}\right\|$ making errors $\tilde{\mathbf{n}}$ decrease. For small values $\tilde{\boldsymbol{\eta}}$, it can be expressed as $\mathbf{K}\tanh(\tilde{\boldsymbol{\eta}}) \approx \mathbf{K}\tilde{\boldsymbol{\eta}}$ and $\left|\tilde{\boldsymbol{\eta}}^T\mathbf{K}\tanh\left(\tilde{\boldsymbol{\eta}}\right)\right| > \left|\tilde{\boldsymbol{\eta}}^T\delta_{\tilde{\boldsymbol{\eta}}}\right|$ can be written as $\left\|\tilde{\boldsymbol{\eta}}\right\| > \left\|\delta_{\tilde{\boldsymbol{\eta}}}\right\| / \lambda_{\min}(k)$ which implies that the error $\tilde{\boldsymbol{\eta}}$ is delimited by, $\left\|\tilde{\boldsymbol{\eta}}\right\| \le \left\|\delta_{\tilde{\boldsymbol{\eta}}}\right\| / \zeta\lambda_{\min}(k)$ with $0 < \zeta < 1$. Therefore, if $\delta_{\tilde{\boldsymbol{\eta}}}(t) \neq 0$ the training error $\tilde{\boldsymbol{\eta}}\left(t\right)$ is ultimately delimited.

C. Aerial Manipulator Controller

From the object manipulation control the desired positions $\mathbf{h}_d = [h_{1xd} \quad h_{1yd} \quad h_{1zd} \quad h_{2xd} \quad h_{2yd} \quad h_{2zd}]^T$ of the aerial manipulator with a variation $\dot{\mathbf{h}}_d = [\dot{h}_{1xd} \quad \dot{h}_{1yd} \quad \dot{h}_{1zd} \quad \dot{h}_{2xd} \quad \dot{h}_{2yd} \quad \dot{h}_{2zd}]^T$ are obtained; therefore, the control error is defined as $\tilde{\mathbf{h}}(t) = \mathbf{h}_d(t) - \mathbf{h}(t)$. Hence, the proposed formation control law is defined as,

$$\mathbf{u}\left(t\right) = \mathbf{J}^{\#}\left(\dot{\mathbf{h}}_d + \mathbf{Q}\tanh\left(\tilde{\mathbf{h}}\right)\right) \tag{6}$$

where, $\mathbf{Q}$ it's a diagonal matrix of positive gain. Now, to demonstrate stability is defined candidate function as, $\mathbf{V}(\tilde{\mathbf{h}}) = \frac{1}{2}\tilde{\mathbf{h}}^T\tilde{\mathbf{h}} > 0$. Hence, introducing the closed-loop equation in the time derivative of $\dot{\mathbf{V}}(\tilde{\mathbf{h}})$ is obtained,

$$\dot{\mathbf{V}}\left(\tilde{\mathbf{h}}\right) = -\tilde{\mathbf{h}}^T\mathbf{Q}\tanh\left(\tilde{\mathbf{h}}\right) < 0\,. \tag{7}$$

As described, the equilibrium point is asymptotically stable, *i.e.* $\tilde{\mathbf{h}}\left(t\right) \to 0$ asymptotically.

Similar to formation control, if you relax the assumption of perfect velocity tracking you have to $\tilde{\mathbf{u}} = \mathbf{u}_d - \mathbf{u}$ where (7) is defined as $\dot{\mathbf{V}}\left(\tilde{\mathbf{h}}\right) = \tilde{\mathbf{h}}^T\tilde{\mathbf{u}} - \tilde{\mathbf{h}}^T\mathbf{Q}\tanh\left(\tilde{\mathbf{h}}\right)$. Therefore, the error $\tilde{\mathbf{h}}$ is delimited by, $\left\|\tilde{\mathbf{h}}\right\| \le \left\|\tilde{\mathbf{u}}\right\| / \zeta\lambda_{\min}(\mathrm{Q})$ with $0 < \zeta < 1$. Therefore, if $\tilde{\mathbf{u}}(t) \neq 0$ the formation error $\tilde{\mathbf{h}}(t)$ is ultimately bounded .

IV. EXPERIMENTAL RESULTS

In order to evaluate and validate the performance of both the simulator and the controller, two experimental tests will be carried out. The first one will observe the behavior of the aerial manipulator robot when a single user performs transport and object manipulation tasks; the second test will be carried out by two users in the multiplayer room performing collaborative tasks.

A. Experimental Test 1:

The first experimental test as shown in Fig. 5, presents the tele-operation of the aerial manipulator robot for one user, the objective of the experiment is to transport the rods from one place to another, checking the effectiveness of the control algorithm implemented.

By means of the haptic input device (NOVIT FALCON) the operator enters the desired positions which are then translated into velocities, which the controller sends to the aerial manipulator robot in the simulator, the same one that feeds back the velocity information, and it will be sent to the control law to act so that the errors go to zero.

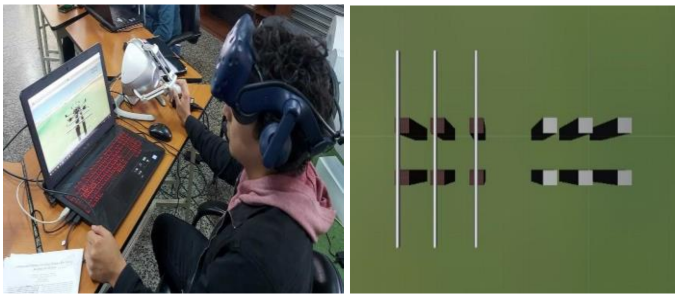

Figure 5. Experimental test by a user.

The virtual reality device allows the immersion of the operator, allowing him to fulfill the required tasks as shown in Fig.6.

Figure 6. Start the tele-operation of the aerial manipulator

Fig. 7 presents the multiple experiments of transport and manipulation of the bars, performed by a single user.

Figure 7. Tele-operation task of the aerial manipulator robot

The graph of Fig. 8, presents the results of the experiment, in them we can see the stability of the trajectory control algorithm in the tasks mentioned above, the graphs present comparison of velocities (linear and angular), sent through haptic device, and those that have been fed back by the aerial manipulator robot in the simulator.

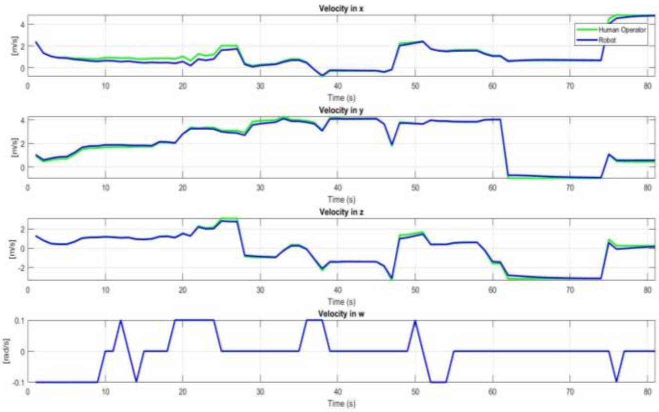

Figure 8. Comparison of velocities applied by the operator versus velocities of the aerial manipulator robot in the single-user simulator.

B. *Experimental Test 2:*

The second experimental test consists of creating a multiplayer room for local tele-operation; with the objective of carrying out a collaborative task of transporting and manipulation loads, the players will enter the virtual environment with a user name, as shown in the input interface in Fig. 9

MANIPULADOR AEREO MULTIUSUARIO

Control aereo colaborativo autonomo y teleleoperado

Usuario 1 :

Alex

Ingresar

MANIPULADOR AEREO MULTIUSUARIO

Control aereo colaborativo autonomo y teleleoperado

Usuario 2 :

Juan

Ingresar

Figure 9. Multi-user platform for collaborative control

As shown in Fig. 10, to carry out the collaborative work experiment, each user is required to have a haptic device, so that they can control the aerial manipulator robot independently, in order to carry out tasks together with user number 2.

Figure 10. Multi-user platform for collaborative control

The evolution of experiment 2 is shown in Figs. 11 -14.

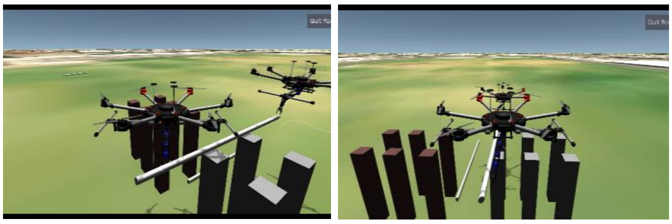

Figure 11. Collaborative tele-operation control of aerial manipulator robots

Figure 12. Start of the collaborative control task

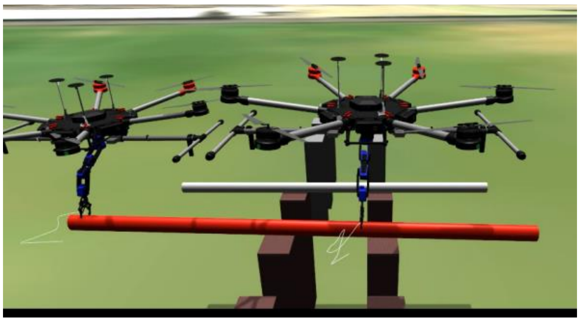

Figure 13. Collaborative object transport

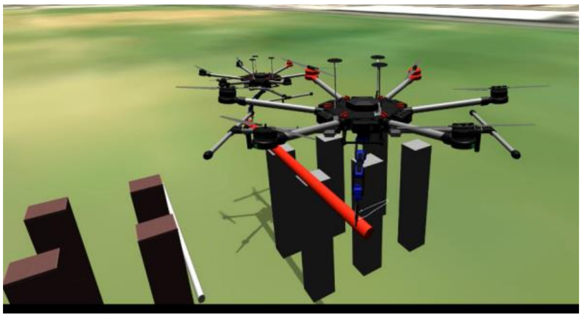

Figure 14. Finalization of the Collaborative Task

The behavior of the controller is shown in Fig. 15 and 16, making a comparison of both linear and angular velocities, applied by the haptic device, versus the velocities fed back through the simulator, both for user 1 and user 2.

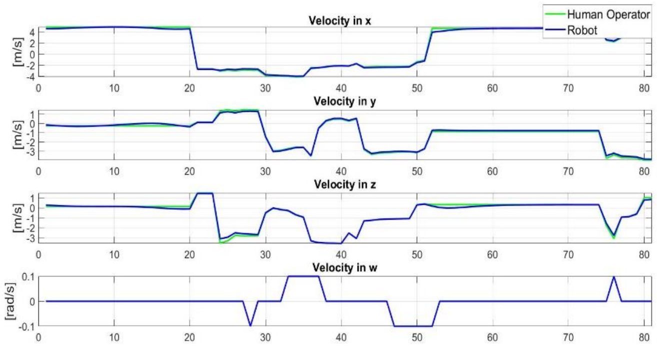

Figure 15. Comparison of velocities applied by the operator vs. velocities of the aerial manipulator robot in the simulator for the user 1.

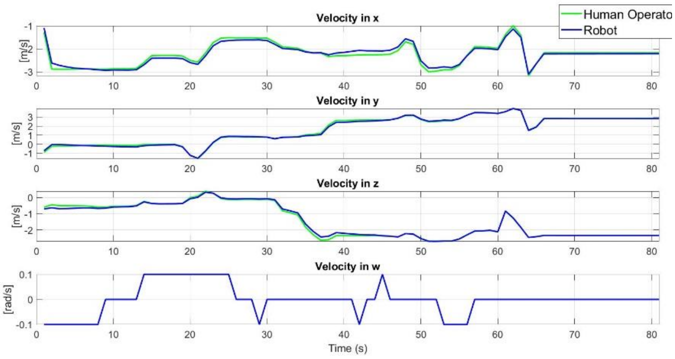

Figure 16. Comparison of velocities applied by the operator vs. velocities of the aerial manipulator robot in the simulator for the user 2

V. CONCLUSIONS

In the present, a simulator implemented in the Unity 3D platform was proposed, which allowed the evaluation of collaborative control algorithms in transport and object manipulation tasks for aerial manipulator robots. The design of the controller is based on a kinematic control that meets the purpose of the movement. Stability and robustness are tested by the Lyapunov method. In addition, the graphic engine presents great advantages when using Virtual Reality as an immersion method for the operator when executing different tasks. The simulation results have shown that the implemented control algorithm allows the robots to comply with the trajectory tracking, leading to zero control errors. As future work, it is intended to implement in a real aerial manipulator robot, the control law, to determine the effectiveness of the virtual simulator as an evaluation platform

ACKNOWLEDGEMENTS

The authors would like to thank the Coorporación Ecuatoriana para el Desarrollo de la Investigación y Academia- CEDIA for their contribution in innovation, through the CEPRA projects, especially the project CEPRA-XIII-2019-08 *"Collaborative system of air robots for handling loads with optimal resource consumption"*; also the Universidad de las Fuerzas Armadas ESPE and the Research Group ARSI, for the support for the development of this work.

REFERENCES

[1]. Andaluz, Víctor H., et al.: Robust control with dynamic compensation for human-wheelchair system. In: Zhang, X., Liu, H., Chen, Z., Wang, N. (eds.) ICIRA 2014. LNCS (LNAI), vol. 8917, pp. 376–389. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-139661_37

[2]. Andaluz, Víctor H., Ortiz, Jessica S., Sanchéz, Jorge S.: Bilateral control of a robotic arm through brain signals. In: De Paolis, L.T., Mongelli, A. (eds.) AVR 2015. LNCS, vol. 9254, pp. 355–368. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-22888-4_26

[3]. Llort M, Aguasca A, Lopez-Martinez C, Martinez-Marin T. Initial evaluation of SAR capabilities in UAV multicopter platforms. IEEE J. Sel. Topics Appl. Earth Obs. Remote Sensing. 2018;11(1):127-140.

[4]. Naldi, R., Gentili, L., Marconi, L.: Modeling and control of the interaction between flying robots and the environment. IFAC Proc. Vol. 43(14), 975–980 (2010)

[5]. Fumagalli, M., Naldi, R., Macchelli, A., Forte, F., Keemink, A., Stramigioli, S., Carloni, R., Marconi, L.: Developing an aerial manipulator prototype: physical interaction with the environment. IEEE Robot. Autom. Mag. 21(3), 41–50 (2014)

[6]. Orsag, M., Korpela, C., Bogdan, S.: Valve turning using a dual-arm aerial manipulator. In: Unmanned Aircraft Systems (ICUAS), vol. 2014(1), pp. 1–7 (2014)

[7]. A. Ferrara and G. P. Incremona, "Design of an Integral Suboptimal Second-Order Sliding Mode Controller for the Robust Motion Control of Robot Manipulators," in IEEE Transactions on Control Systems Technology, vol. 23, no. 6, pp. 2316-2325, Nov. 2015.

[8]. J. Osuský and J. Cigánek, "Trajectory tracking robust control for two wheels robot," 2018 Cybernetics & Informatics (K&I), Lazy pod Makytou, 2018, pp. 1-4.

[9]. Lawrence, D.A.: Stability and transparency in bilateral teleoperation. IEEE Trans. Robot. Autom. 9(5), 624–637 (1993). 2015 24th IEEE International Symposium on Communication (RO-MAN), pp. 431–437. IEEE

[10]. Lichiardopol S.A Survey on Tekeoperation. Technische Universitat Eindhoven, DCT report, December 2007.

[11]. Bustamante, A., Guerrero, G., Rodrigues, N., & Pereira, A. (2017, June). UAV simulator for grown-up people quality of life enhancement. In 2017 12th Iberian Conference on Information Systems and Technologies (CISTI) (pp. 1-6). IEEE.

[12]. J. Sterman, «Interactive web-based simulations for strategy and sustainability: The MIT Sloan LearningEdge management flight simulators, Part I,» System Dynamics Review, vol. 30, nº 1-2, pp. 89- 121, 2014.

[13]. G. Rao, P. Rao y R. Duvvuru, «A Drone Remote Sensing for Virtual Reality Simulation System for Forest Fires: Semantic Neural Network Approach,» IOP Conference Series: Materials Science and Engineering, vol. 149, nº 1, pp. 120-131, 2016.

[14]. G. De la Torre, M. Ramallo y C. Elizabeth, «Workload perception in drone flight training simulators,» Computers in Human Behavior, pp. 449-454, 2016.

[15]. L. Nævdal, «A Comparative Analysis of the Relationship between Gaming Controllers and Game Mechanics - PDF,» UNIVERSITY OF BERGEN Master thesis in Information Science, 08 Junio 2013. [En línea].

Available: http://bora.uib.no/handle/1956/7042. [Último acceso: 18 Febrero 2017].

[16]. Ortiz J.S. et al. (2018) Modeling and Kinematic Nonlinear Control of Aerial Mobile Manipulators. In: Zeghloul S., Romdhane L., Laribi M. (eds) Computational Kinematics. Mechanisms and Machine Science, vol 50. Springer, Cham.

[17]. Andaluz, V. H., Jessica, S., Ortiz, María, F., Molina, Jose Varela and Vicente Morales (2017, December). Coordinated Control of a Omnidirectional Double Mobile Manipulator. K.J. Kim et al. (eds.), IT Convergence and Security 2017, Lecture Notes in Electrical Engineering 449,

[18]. Andaluz V.H., Molina M.F., Erazo Y.P., Ortiz J.S. (2017) Numerical Methods for Cooperative Control of Double Mobile Manipulators. In: Huang Y., Wu H., Liu H., Yin Z. (eds) Intelligent Robotics and Applications. ICIRA 2017. Lecture Notes in Computer Science, vol 10463. Springer, Cham