IMAGE LICENSED BY INGRAM PUBLISHING

A 3D-Printable Docking System for Aerial Robots

Controlling Aerial Robotic Manipulators in Outdoor Industrial Applications

By Pablo Ramon Soria, B.C. Arrue, and Anibal Ollero

This article proposes a novel docking system design for unmanned aerial vehicles (UAVs) that provides measurements of the robot's position at high frequency. These measurements are used to control the aerial robot, enabling it to hover while it performs any kind of manipulation task in GPS-denied industrial environments without causing the UAV to drift or putting at risk the platform and its environment. The novel tool is designed as an arm end effector, preventing the aerial manipulator from colliding while in operation. A cascade controller is proposed to close the position loop. An additional use case for the docking system is described in the "Experimental Validation" section; it consists of performing position-based servoing (PBS) of a second manipulator using the position provided by the docking tool and the manipulator's kinematic model.

Digital Object Identifier 10.1109/MRA.2018.2884744
Date of publication: 24 January 2019

1070-9932/19©2019IEEE

The prototype system is 3D-printed in acrylonitrile butadiene styrene (ABS). This article presents outdoor experimental results. The accuracy of the system is evaluated against GPS, a state-of-the-art visual algorithm, and a laser total station as ground truth. The measurements obtained from the docking tool exceed a frequency of 1,000 Hz, which overtakes common localization algorithms.

Recent Reviews of UAVs

In recent years, interest in aerial robots has increased. Their ability to move freely within a space allows robots to work in several situations. Montambault et al. [1] reviewed multiple civil and industrial applications with UAVs. Notably, applications for inspection and maintenance tasks in factories and power stations are on the rise. In particular, robots' ability to access high-altitude locations, which can be dangerous for human operators, has made them very popular for pertinent applications, even though their use is generally only for location and inspection.

Large-scale industrial facilities, such as factories or energy power stations, require intensive and costly labor inspection and maintenance. Innovative solutions use robots to automate such tasks in situ. Stokkeland et al. [2] employed a combination of line and feature detection to locate wind turbines for inspection purposes. Addabbo et al. [3] used thermal images to locate and inspect solar panels in photovoltaic plants.

However, extra effort needs to be made if robots are to perform maintenance or repair tasks. To enable UAVs to interact with industrial facilities, recent research provides them with manipulators. Orsag et al. [4] pointed out the key challenges for controlling UAVs with embedded manipulators. Lippiello and Ruggiero [5] implemented an impedance control to improve the robot's actuation against external disturbances produced by the arm and applied the control algorithm in simulations. In [6], the authors proposed a multilayered control that takes into account the movements of the built-in arm to improve the control of the UAV. Another approach can be found in [7]; here, the authors described an adaptive sliding-mode controller.

Korpela et al. [8] developed an aerial robot equipped with a pair of manipulators for actuating valves. They studied the forces applied to the robot resulting from turning the valve and showed experiments in an indoor controlled environment. Another dual-arm system for UAVs was developed in [9]. In this study, the authors focused on the torque effects on the UAV due to the movement of the pair of arms while hovering. Typically, built-in arms are articulated for robots, and work in [10] uses a parallel manipulator at the bottom of the UAV.

Parra-Vega et al. [11] analyzed the forces exerted on a UAV in contact with stiff environments, such as a wall, in simulations. With a similar purpose, the authors in [12] designed a device that measures forces in contact with stiff surfaces. They used that information to control the position of the aerial robot and the force exerted by it.

Nevertheless, most studies perform the experiments at indoor testbeds using a motion-capture system such as OPTITRACK or VICON. These localization systems provide accurate and fast measurements of the positions of the robots. They are useful for validating algorithms but are not realistic assumptions for outdoor applications.

Several studies about positioning robots are based on vision sensors, such as color cameras; red, green, blue plus depth cameras; and lidars. In these cases, the location of the robot is tackled following a simultaneous localization and mapping (SLAM) procedure. Algorithms such as ORB-SLAM [13], Real-Time Appearance-Based-MAP [14], lidar odometry and mapping [15], and many others have proved to be good general-purpose solutions. Nonetheless, they rely on visual landmarks and high-cost computational operations, which must be performed in the onboard computers. Additionally, the algorithms can become "lost" due to occlusions or lack of landmarks.

Other visual methods, called *visual odometry estimators*, are based on computing an estimation of the position of the robot. Forster et al. [16] proposed semidirect visual odometry, a visual algorithm based on the exploitation of the gradients in intensity to compute an estimation of the position of the robot. Sun et al. [17] proposed an algorithm that fuses the odometry computed from visual features with the multirotor's inertial measurement unit (IMU) to provide a robust estimation of its pose.

The main contribution of the current study is the design of a novel tool, henceforth referred to as the *docking tool,* that allows aerial robots to remain in flight close to a target position in outdoor environments without requiring any other external device. This device has the advantage of providing a high rate of robot position measurements with few computational costs; thus, the computer's remaining power resources can be used for other tasks or even to carry a smaller computer without compromising the platform's payload.

Low-Cost Docking System

This section describes the hardware design of the novel tool. Developing tools for aerial robots is usually more constrained than for in-ground systems due to payload limitations and stability issues. In this article, the following assumptions are adopted:

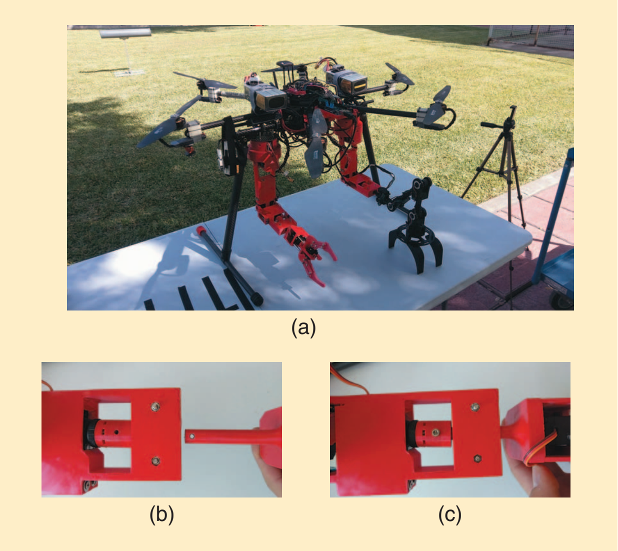
- Perturbations produced by wind are relatively small.
- The UAV has a low-level controller that inputs the desired speed in Cartesian coordinates and outputs motor speed.

To accomplish the first assumption, the system was tested in low-wind conditions. For the second assumption, the robot was equipped with a PIXHAWK [18] autopilot that uses px4 software [19].

The aerial robot has two built-in arms, each with a different tool. The right arm is provided with a gripper to perform various manipulation tasks. The docking tool is attached to the left arm and provides the position of the UAV relative to its attachment base. These measurements are used for stabilizing it close to the manipulation space. Figure 1 shows the aerial robot with all the tools.

Docking Tool Model

This section describes the model of the docking tool, which consists of a passive multilink arm with sensors in the joints to measure the angles between the links. The main criteria during the design of the tool were to minimize the total weight and the friction in the joints, thus reducing the torques exerted on the arm and subsequently on the UAV. To reduce the weight, the structural parts are designed to be thin and hollow. The components are 3D-printed using ABS, making the tool lightweight and easier to replace. Furthermore, the production costs are lower than when using aluminum or carbon fiber, and the components do not need to be built, machined, or postprocessed.


(a)

(b)

(c)

Figure 1. (a) A dual-arm aerial robot with a gripper and a docking tool. (b) and (c) The quick-release system for attaching different tools, i.e., the gripper or the docking tool.

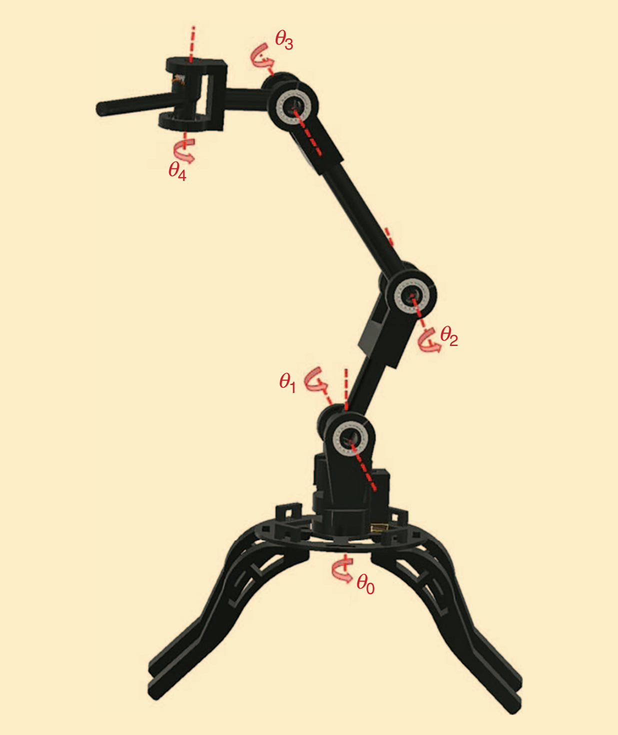


Figure 2. The CAD model of the docking tool with a base for attaching pipes to joint edges.

The tool is not actuated, i.e., it does not need any motor, making it lighter and eliminating the battery weight. Figure 2 shows the CAD model of the tool. Bearings have been placed in the joints to minimize friction. These are made of acetal plastic, which makes them 10 times lighter than common metal bearings.

The tool is composed of five joints. The base joint (or θ_0) provides a rotation on the z-axis. The following two joints (θ_1 and θ_2) make up a two-link arm that gives the robot free 3D movement within the work zone. Joint θ_3 is set to provide an extra degree of freedom (DoF), allowing the robot to remain parallel to the floor independent of the position of the two-link section. Finally, the last joint (θ_4) adds to the robot another DoF, making it able to maintain the heading effortlessly.

The kinematic model is shown in (1):

$$T_{UAV} = f(\theta_0, \theta_1, \theta_2, \theta_3, \theta_4) = T_0 \cdot T_1 \cdot T_2 \cdot T_3 \cdot T_4, \tag{1}$$

being

$$T_0 = \begin{bmatrix} c\theta_0 & -s\theta_0 & 0 & 0 \\ s\theta_0 & c\theta_0 & 0 & 0 \\ 0 & 0 & 1 & l_0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{2}$$

$$T_i = \begin{bmatrix} 1 & 0 & 0 & l_i \\ 0 & c\theta_i & -s\theta_i & 0 \\ 0 & s\theta_i & c\theta_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \forall i = 1, 2, 3, \tag{3}$$

$$T_4 = \begin{bmatrix} c\theta_1 & 0 & s\theta_1 & 0 \\ 0 & 1 & 0 & l_4 \\ -s\theta_1 & 0 & c\theta_1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4}$$

The docking tool system has 5 DoF for the drone's movement. An additional joint in the axis of the last bar has also been considered to provide free rotation related to the roll of the UAV. However, this rotation is significantly small due to the assumption of small perturbations, so the joint is not included in the final design.

The joints of the tool are provided with potentiometers that are used for measuring the angles. The voltage signals from the potentiometers are measured by an electronic device connected to the onboard computer. Then, the signals are mapped to the angles because the voltage in the resistances changes linearly; the mapping of variables is a linear map.

The sensors are wired using internal holes on the joints, as shown in Figure 3(c). This minimizes the forces exerted by the cables on the joints.

The angles measured from the potentiometers are used to estimate the current pose of the UAV. Together with the information about the arms, these measurements are used to close the loop of the control system described in the "Control Loop" section. Figure 4 shows a 3D virtual visualization at different times of real experiments.

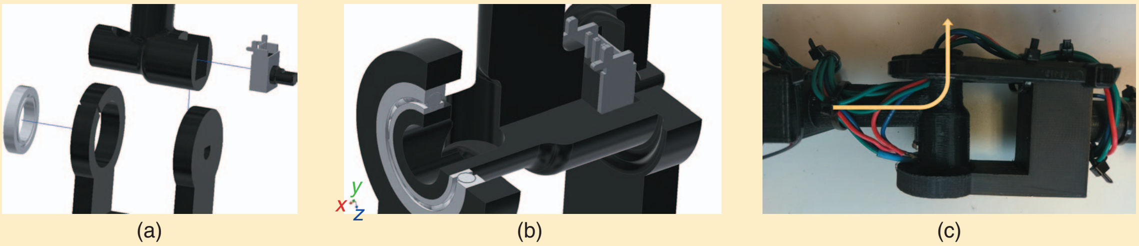


Figure 3. (a) An image of the component exploded. (b) The part of the joint sliced to see the internal holes and the components assembled. (c) The wiring system, passing through the holes seen in (a) and (b).

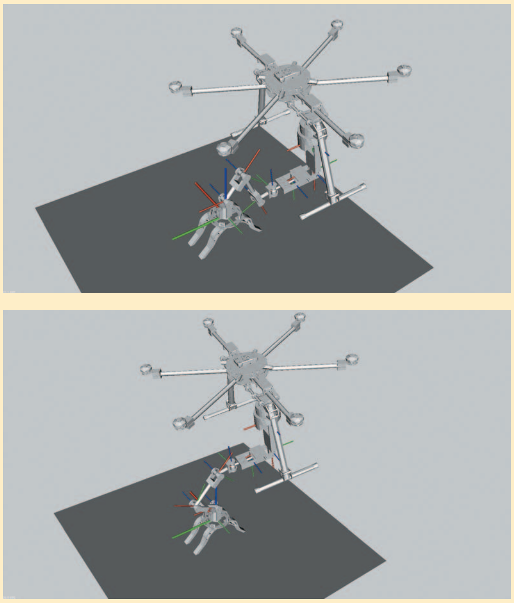


Figure 4. Two images that show the virtual visualization of the aerial platform with the docking tool during the experiments.

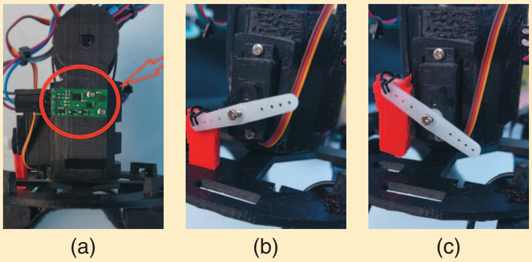


Figure 5. (a) The accelerometer placed in the base joint to measure the orientation of the base while it is docked. (b) and (c) The servo that locks the base of the docking tool until it is placed.

Furthermore, the base of the docking tool is provided with two additional components, shown in Figure 5. The first is a three-axis accelerometer; the second is a servo that locks the rotation of the base during flight to improve the coupling phase.

Table 1. The specifications of the low-cost docking tool.

Device/Characteristic	Value	
Potentiometer resistance (KW)	20	
Potentiometer angle range (°)	270	
Operating voltage (V)	5	
Power consumption (W)	0.25	
Servos	SG90 Pro 9g	
Accelerometer	L3GD20H and LSM303D Carrier	
Longitudes (m)	l_0	0.071
	l_1	0.105
	l_2	0.155
	l_3	0.07
	l_4	0.075
Total weight (g)	150	
Base material	Plastic ABS	

The use of the accelerometer is essential to guarantee that the 6D pose of the end effector can be used to control the UAV. Because the arm that holds the docking tool remains parallel to the horizontal plane of the platform, it is important to know the orientation of the base of the docking tool to properly compute not only the position but the full pose. If, during the attachment, the base is not entirely parallel, for any reason this will lead to a rotation in the end effector.

The second component, the servo, has been integrated to prevent the base from rotating while the platform is flying. As the joints are not actuated, the base tends to rotate, making the docking stage difficult. The purpose of the servo is to lock until the platform has docked. More detailed specifications of the docking tool components are summarized in Table 1.

Finally, Figure 6 shows the workspace of the docking tool's end position and the quality of each position. Quality is evaluated using the distance of the joints to the saturation points. This distance is mapped to a value within 0 and 1, and all the values are multiplied, obtaining a single bounded value. Consequently, if one of the joints gets close to a saturation limit, the overall quality of the position decreases. The values are represented using a heat-map scale, with blue being the best value and red the worst. These values are intended to be used

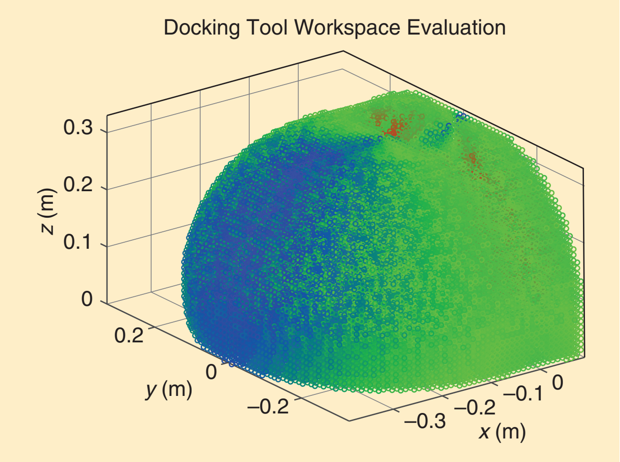


Figure 6. The reachability of the end position of the docking tool with heat-map colors evaluating the closeness of the joints to the saturation limits.

in future work to perform smarter control while trying to keep the position of the docking tool in a well-conditioned volume of the workspace.

The Arm Model

Each arm is composed of three rotation joints and a general purpose end-tool socket with an extra rotation in the direction of the arm. The first rotation is in the z-axis, and the two remaining rotations compose a two-link arm.

Hence, each arm has 4 DoF. The end effector has a quick-release system that makes it easy to replace any tool, as shown in Figure 1(b) and (c). The arms are part of the open source project Hecatonquiros (https://github.com/bardo91/hecatonquiros) developed by the Group of Robotics, Vision, and Control at the University of Seville, Spain. This project aims for a cheap and easy-to-use framework for aerial manipulation.

In the "Experimental Validation" section, a use case for the docking tool proposed in this article is described. The position provided by the tool is used to estimate the relative pose of the second arm's end effector regarding the attached position. Based on this information, it is possible to perform a PBS algorithm to keep the end effector in a fixed point or location.

The robot manipulator is defined by n joints, with the scalar variables $\Theta = \theta_1, \ldots, \theta_n$ describing their values. Having the geometric design of the arm, each link has a position $S = s_1, \ldots, s_n$. Using Denavit–Hartenberg parameters [20], it is possible to determine each DH_i matrix, which transforms between link $i-1$ to i, i.e., forward kinematics.

To perform the position-based visual servoing (PBVS), the system needs to compute an adequate set of joints Θ that place the end effectors (a gripper, in this case) on a target location T_t, i.e., inverse kinematics. However, this problem is not straightforward when using complex manipulators. Iterative methods [21] and/or sampling methods [22] offer a good solution to this problem and fit well for the PBVS. The forward kinematic functions $\Theta \mapsto S(\Theta)$ are $\mathbb{R} \mapsto \mathbb{R}^3$. These functions can be linearly approximated using Jacobian matrices in the current state Θ; velocities can be expressed as

$$S(\dot{\theta}) = J(\Theta)\dot{\Theta} \qquad J(\Theta) = \left(\frac{\delta s_i}{\delta \theta_j}\right)_{i,j}.$$

In this article, the right arm is controlled using the information from the docking tool and applying a Jacobian damped least square gradient descent method [23], [24] that is more robust to inverse and pseudoinverse methods [24] close to singularities and instabilities in the Jacobian matrices:

$$\text{error} = \begin{cases} X_k - X_{k-1} \\ Q_k - Q_{k-1} \end{cases} \qquad J_c = \begin{bmatrix} J_X \\ J_\Theta \end{bmatrix}$$

$$\Delta\theta_k = (J_c^T(\theta_k) \cdot J_c(\theta_k) + \lambda^2 \cdot I)^{-1} \cdot \text{error}, \tag{5}$$

where X_k and Q_k are the target poses of the gripper at instant k, θ_k is the joint angle, J_X and J_Q are the position and orientation Jacobians at k, and λ is the damping coefficient to reduce the issues related to the inversion of the matrix.

Control Loop

A cascade control system is proposed for positioning the aerial robot. Figure 7 shows the controller structure. The inner loop corresponds to the internal controller provided by the px4 software. It consists of Cartesian speed control that translates from the desired velocity to the corresponding actuation on the motors. The outer loop uses the position obtained by the docking tool to produce a target speed to control the robot.

The outer controller is a proportional–integral–derivative (PID) tuned to provide quick responses to the perturbations on the UAV and the drifts of the internal controller (generally due to errors in the internal estimators of the px4 software: GPS errors, IMU drifts, and so on). The position of the docking tool is compared against a target position to compute an error $e(t)[m]$. This error in position feeds the controller, which produces at the output a control signal $u(t)$, i.e., a reference speed in $[ms]$ that feeds the autopilot.

While the tool is docked, it throws measures of the relative position from its

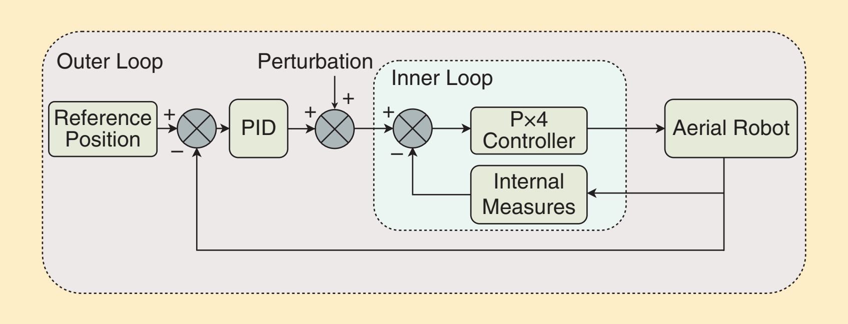


Figure 7. The scheme of the cascade control system.

base. The error that feeds the control loop is computed by the difference in the current position and a reference position. To smooth the state of the robot, the data obtained from the tool are filtered using an extended Kalman filter [25].

The PID was coded with an antiwindup block to avoid large oscillations due to the integral factor. Additionally, the output speeds are saturated to prevent abrupt control signals due to the derivative terms. Table 2 presents the PID parameter values that were tuned from the experiments, starting from a controller flying freely in the air and then tuning the parameters with the information from the docking tool.

Table 2. PID parameters.

	K_p	K_i	K_d	Antiwindup	Signal Saturation (m/s)
x	0.8	0.01	0.7	10	2
y	0.8	0.01	0.7	10	2
z	0.3	0.03	0.7	10	1

Experimental Validation

This section presents the experiments performed to validate the tool.

Experimental Setup

In addition to the components described in the "Low-Cost Docking System" section, the aerial robot needs other devices to perform the experiments. Figure 8 shows all system components. The arms are actuated using the pulsewidth modulation ports of an Arduino Mega ADK microcontroller. The potentiometers on the joints of the docking tool are connected to the analog inputs of the microcontroller as well. The Arduino microcontroller is connected to an onboard Intel NUC computer, which is used as the main computer. Here, the sensors' lectures are gathered to produce the control signals sent to the autopilot (PIXHAWK). The autopilot receives the target speed and controls the multirotor. Additionally, a power supply system is added to feed each device at appropriate voltages.

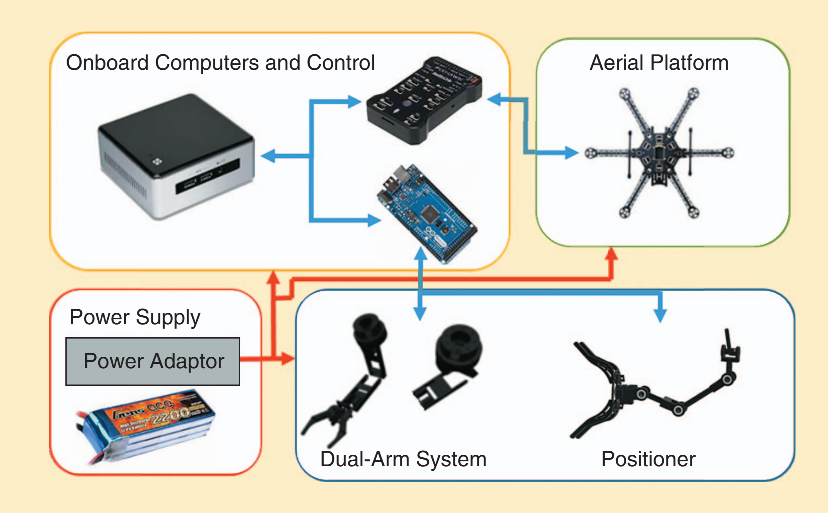


Figure 8. The components of the autonomous docking system.

A set of practical conclusions were obtained during the first stages of the development process:

- To increase the arms' operation range and prevent internal collisions, a foldable landing gear was built in.
- It was observed that the potentiometers' signals saturate before reaching the mechanical extremes. This reduces the tool workspace. In particular, the mechanical range is 170°, and the signal range is 150°. For this reason, it is good to keep the UAV in a position where the joints are not in the limits.
- The docking tool joints are not actuated. During the experiments, the tool hangs until it is docked. In particular, the base joint can rotate. For this reason, that joint is locked with a micro servo, which unlocks the joint once the tool is placed.

Test-Bench and Tool Characterization

A static experiment was performed in a testbench. The purpose is to measure two variables: the accuracy and frequency of the measurements. The data provided by the docking tool are compared against a ground truth obtained by a Leica Total Station MS502 (https://leica-geosystems.com/products/total-stations). It uses a laser that locates a prism within an error of 0.3–0.4 mm.

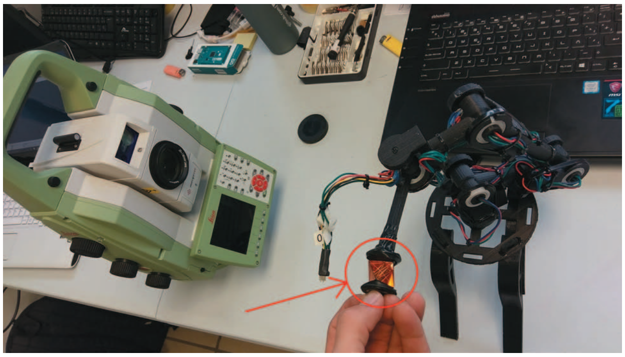


Figure 9. A testbench with a laser system for measuring the accuracy of the docking tool.

The docking tool is placed on a table, and the prism is attached at the end of the positioning system, as shown in Figure 9.

Figure 10 shows the results in the testbench. The end position of the tool is moved to describe a 3D cross, as it attempts to perform the movements over each axis independently. Figure 10(a) shows the end position of the tool; the solid line is the position measured by the docking tool, and the dashed line is the position measured by the total station taken as ground truth. Figure 10(b) shows the difference between both measures.

The mismatch in the z-axis, within time counts 2,375 and 2,675, corresponds to the fact that the values of the potentiometers saturate when going down. The joints exceed the allowed range, which leads to bad angle measurements. Similar effects can be seen at the limits of the movements in the x- and y-axes, but, in these cases, the effect is only slightly noticeable. This happened intentionally in this experiment to show the saturation effect. Nevertheless, because the workspace and the closeness of the joints to the saturation limits are known at any time, this effect can be mitigated while flying.

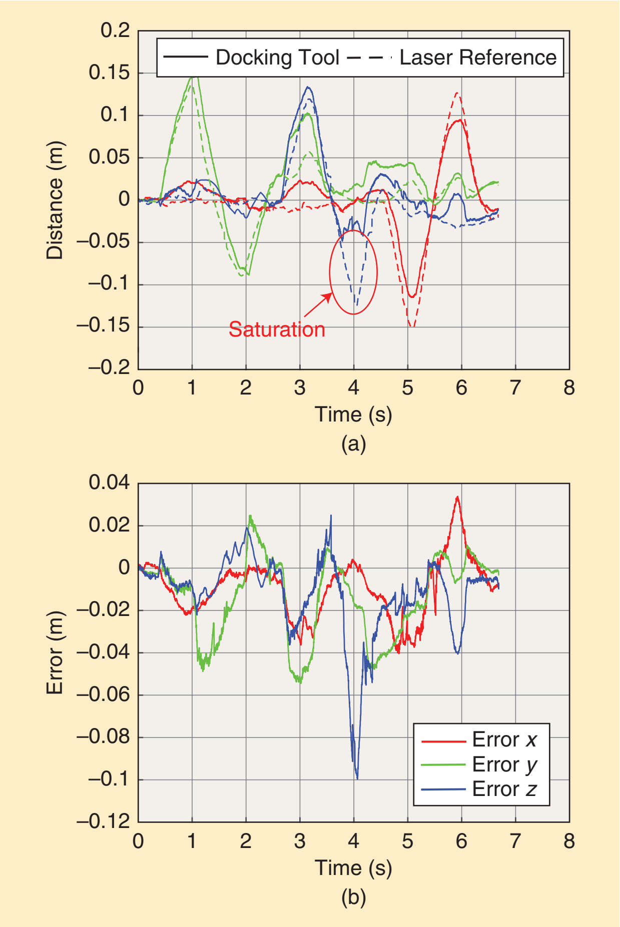


Figure 10. (a) The x, y, and z components of the trajectory of the end effector during the experiment in the testbench measured by the docking tool and the laser system (x: red; y: green; z: blue). (b) The relative error among the measurements. The relative error has been enlarged for figure clarity.

GPS Positioning Characterization

The first outdoor experiment was executed using GPS to obtain the position reference for the control loop. This experiment was shown to characterize the magnitude of typical errors using this common positioning device and comparing it with our positioning system. The real position of the robot is measured with the total station. Figure 11 shows the error in the position of the UAV according to a fixed set point measured with the total station. The experiments were performed using the hardware specified in the "Experimental Setup" section on a clear day with very low wind conditions.

In this experiment, it was observed that the errors can be large in some situations. Moreover, these experiments were performed outdoors on a clear day. Thus, there could be worse conditions, such as noise or GPS denial that would induce larger errors and put the platform and the environment in danger. This inaccuracy exceeds the workspace of the arms, making any kind of inspection or manipulation task in the target zone difficult.

Docking and Autonomous Control

During the experiment (https://youtu.be/Vk9G7lb_r6I), the UAV takes off and moves to the target position to attach the docking tool. Then, it starts measuring the relative position of the robot and performs the autonomous control. The position of the UAV is also acquired using the total station as the ground truth. Figure 12 shows snapshots of different experiments of the robot docked to a pipe.

Additionally, a camera is attached to the docking tool to compare the results with a monocular vision system (ORB_SLAM2 [13]). In Figure 13, the data recorded during an experiment using the docking system can be observed. Figure 13(a)–(c) shows the difference between the current position and the reference position measured by the different localization systems. The solid line represents the difference in position by the docking tool. The dashed line is the one measured by the total station, and the solid line with dots is the one measured by the vision algorithm. Figure 13(d) shows the control signals produced by the PID and generated by the outer loop of the cascade controller from the error in the position obtained by the docking tool.

Table 3 compares the errors and deviations of the different localization systems studied against the docking tool during the experiments at each axis. It is evident that relying solely on GPS is unsafe for the platform. Both the visual algorithm and the docking tool provide

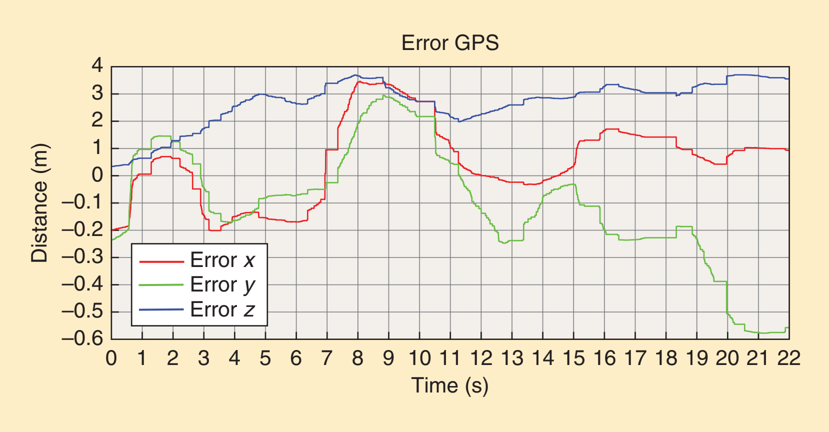


Figure 11. The position of the UAV measured by the total station during a hovering test using GPS as a position estimator.

similar measures, with the docking tool slightly less accurate. However, the visual localization algorithm consumes significant computer resources, and it gives the location of the robot up to 25 Hz. Conversely, the measurements obtained from the docking system achieve a frequency of 1,200 Hz, which overtakes the vision speed. Moreover, the computer vision approach depends strongly on the lighting conditions.

Position Servoing of the Second Manipulator Fed With Docking Tool Measurements

A final experimental setup has been proposed to enhance the usefulness of the docking tool. As mentioned previously, the aerial platform has two built-in manipulators. The purpose of the docking tool is to provide the aerial platform with a robust position estimation from the attaching point so that it is then able to perform any task with the second arm. In the experiment described in this section, the

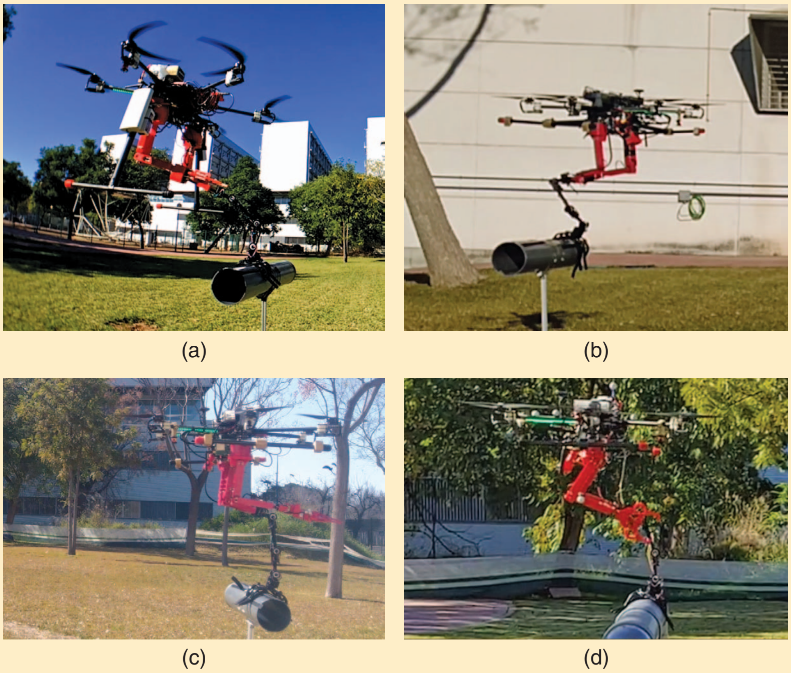

(a)
(b)
(c)
(d)

Figure 12. (a)–(d) Snapshots of the robot docked to a pipe during different experiments. The joints of the docking tool passively adapt to the UAV position, which can vary due to external perturbations.

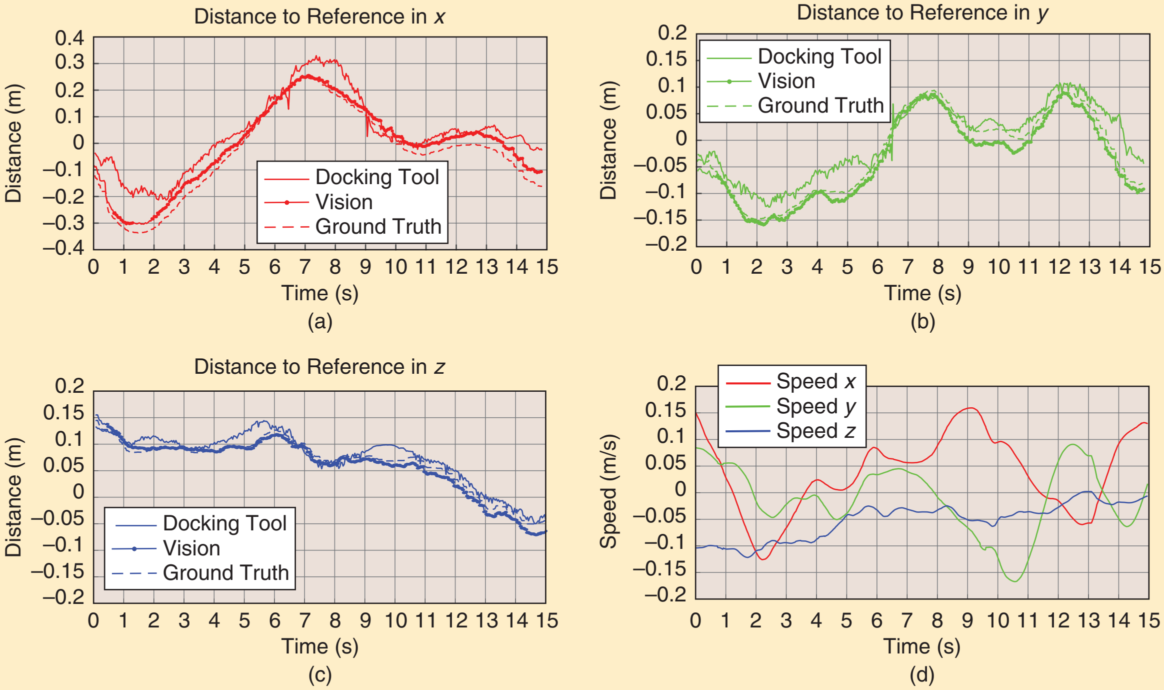


Figure 13. (a)–(c) A comparison of the errors between the current UAV position and a reference position measured from the docking tool (solid line) and the total station (dashed line) and the vision system (solid line with dots) in the three axes. (d) The speed control generated for the outer loop of the cascade controller.

Table 3. The errors and standard deviations during the test experiments.

	GPS		Vision		Docking Tool	
	μ	σ	μ	σ	μ	σ
x (m)	0.164	0.079	0.022	0.027	0.034	0.032
y (m)	0.153	0.123	0.012	0.032	0.036	0.028
z (m)	0.179	0.085	0.023	0.026	0.040	0.039
Average speed (Hz)	10		25		1200	

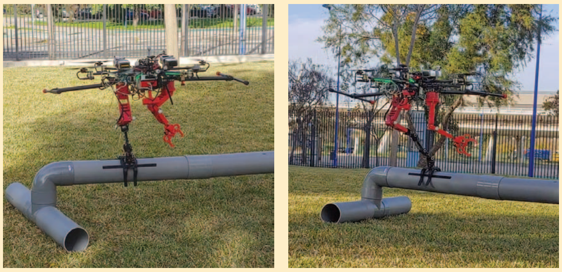


Figure 14. Two snapshots showing experiments with the PBS of the manipulator using the position given by the docking tool.

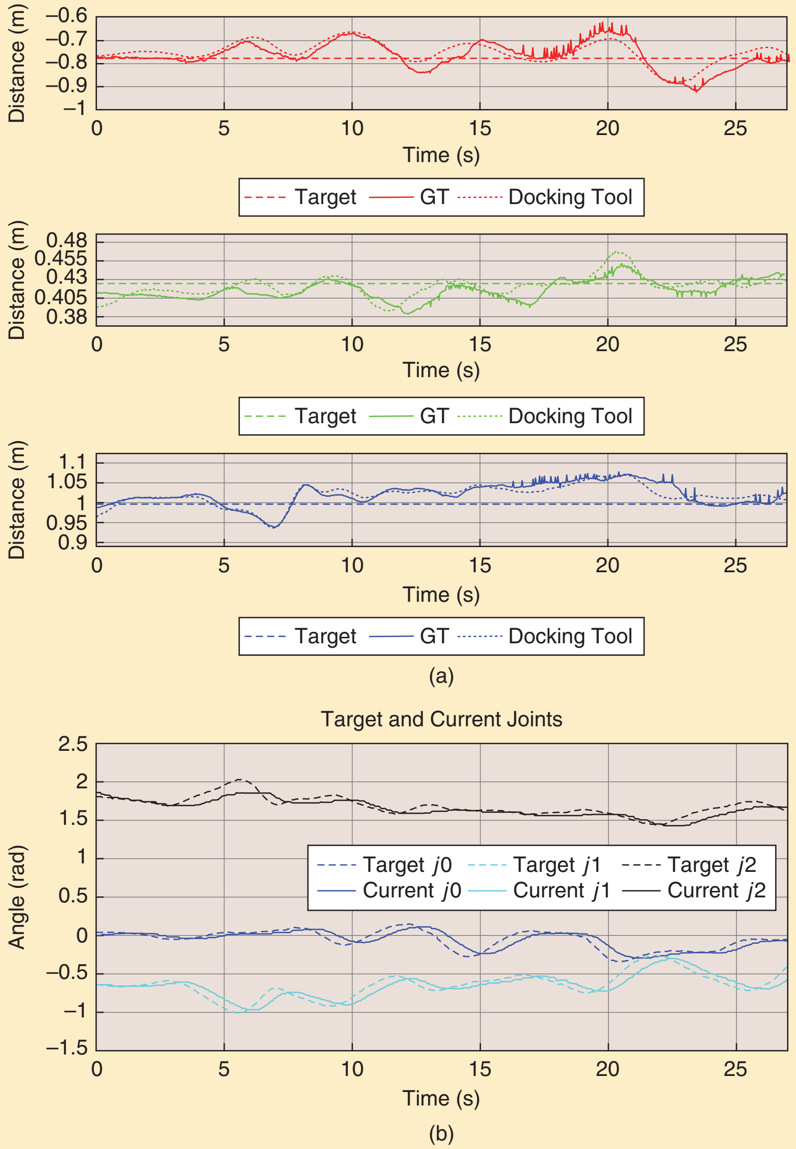


Figure 15. The results of the PBS algorithm. (a) The end effector's position is indicated by the solid lines and the target location by the dashed lines. (b) The values of the joints during the operation.

second manipulator is commanded to place its end effector to a predefined location (Figure 14). The goal is to perform PBS to minimize the error between the end effector's real position and the target position.

To measure the end effector's real position, a reflector prism, detected by the total station, has been added. Using the docking tool and the control loop proposed in the "Control Loop" section, the arm tries to fix the position of the end effector in the target pose.

The expected relative position from the attachment point computed by the docking tool has been compared against the reference position obtained by the total station. Figure 15(a) shows the positions obtained. Additionally, the results of the manipulator's joint state, based on the PBS algorithm to move the arm toward the target point, are shown in Figure 15(b). The dashed lines represent the target joints of the manipulator, and the solid lines are the actual joints at each instant.

Conclusions and Future Work

A low-cost and low-weight in-flight docking system was proposed for a dual-arm aerial manipulator. It was shown that the tool provides a high rate of accurate measurements for the aerial robot's relative position from a target zone. This tool allows the robot to operate even in GPS-denied conditions. Autonomous control of the UAV in contact with a fixed object was achieved. Moreover, the tool is cheap and easy to replace.

The system is still a prototype, and two aspects are identified for future work. First, the optimization of the number of joints and their placement will be studied. As mentioned in the "Docking Tool Model" section, the positioner has 5 DoF, lacking the one that corresponds to the roll of the drone. This extra DoF has been skipped to reduce the weight of the tool and simplify its construction. However, it is considered for later versions. Second, the lengths of the bars are chosen to cover a wider range of space and reduce, as possible, the deadlocks of the tool. Nonetheless, these lengths can be mathematically optimized to fit specific applications and better avoid critical angles of the joints. Furthermore, the accuracy of joint angle measurements can be improved by using more expensive devices, such as high-resolution encoders.

Currently, we are working on the active placement of the docking tool aided by the second manipulator. Additionally, during the experiments, the arm that holds the docking tool remained at the same position. In future work, the arm will move actively to increase the working volume of the robot.

Acknowledgments

We thank other members of the Group of Robotics, Vision, and Control for supporting us during this study. This work was developed in the framework of the European Union-funded projects AEROARMS (SI-1439/2015) and HYFLIER (SI-1762/23/2017). We also thank Manuel Perez Jimenez, Javier Prada Delgado, Jose Maria Aguilar, and the company DroneTools for their collaboration and assistance on the development of the hexacopter platforms.

References

[1] S. Montambault, J. Beaudry, K. Toussaint, and N. Pouliot, "On the application of vtol uavs to the inspection of power utility assets," in *Proc. 1st Int. Conf. Applied Robotics for the Power Industry*, 2010, pp. 1–7.

[2] M. Stokkeland, K. Klausen, and T. A. Johansen, "Autonomous visual navigation of unmanned aerial vehicle for wind turbine inspection," in *Proc. 2015 Int. Conf. Unmanned Aircraft Systems (ICUAS)*, pp. 998–1007.

[3] P. Addabbo et al., "A uav infrared measurement approach for defect detection in photovoltaic plants," in *Proc. 2017 IEEE Int. Workshop on Metrology for AeroSpace (MetroAeroSpace)*, pp. 345–350.

[4] M. Orsag, C. Korpela, and P. Oh, "Modeling and control of mm-uav: Mobile manipulating unmanned aerial vehicle," *J. Intelligent Robotic Syst.*, vol. 69, no. 1, pp. 227–240, Jan. 2013.

[5] V. Lippiello and F. Ruggiero, "Cartesian impedance control of a uav with a robotic arm," *IFAC Proc. Volumes*, vol. 45, no. 22, pp. 704–709, 2012.

[6] F. Ruggiero et al., "A multilayer control for multirotor uavs equipped with a servo robot arm," in *Proc. 2015 IEEE Int. Conf. Robotics and Automation (ICRA)*, pp. 4014–4020.

[7] S. Kim, S. Choi, and H. J. Kim, "Aerial manipulation using a quadrotor with a two dof robotic arm," in *Proc. 2013 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 4990–4995.

[8] C. Korpela, M. Orsag, and P. Oh, "Towards valve turning using a dual-arm aerial manipulator," in *Proc. 2014 IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pp. 3411–3416.

[9] A. Suarez et al., "Lightweight and human-size dual arm aerial manipulator," in *Proc. 2017 Int. Conf. Unmanned Aircraft Systems (ICUAS)*, pp. 1778–1784.

[10] T. W. Danko, K. P. Chaney, and P. Y. Oh, "A parallel manipulator for mobile manipulating uavs," in *Proc. 2015 IEEE Int. Conf. Technologies for Practical Robot Applications (TePRA)*, pp. 1–6.

[11] V. Parra-Vega, A. Sanchez, and C. Izaguirre, "Toward force control of a quadrotor uav in se(3)," in *Proc. 2012 IEEE 51st IEEE Conf. Decision and Control (CDC)*, pp. 1802–1809.

[12] T. Bartelds, A. Capra, S. Hamaza, S. Stramigioli, and M. Fumagalli, "Compliant aerial manipulators: Toward a new generation of aerial robotic workers," *IEEE Robot. Autom. Lett.*, vol. 1, no. 1, pp. 477–483, Jan. 2016.

[13] R. Mur-Artal and J. D. Tardós, "ORB-SLAM2: An open-source SLAM system for monocular, stereo and RGB-D cameras," *IEEE Trans. Robot.*, vol. 33, no. 5, pp. 1255–1262, Oct. 2017.

[14] M. Labb and F. Michaud, "Online global loop closure detection for large-scale multi-session graph-based SLAM," *in Proc. 2014 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pp. 2661–2666.

[15] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," in *Proc. Robotics: Science and Systems,* 2014. doi: 10.15607/RSS.2014.X.007.

[16] C. Forster, Z. Zhang, M. Gassner, M. Werlberger, and D. Scaramuzza, "Svo: Semidirect visual odometry for monocular and multicamera systems," *IEEE Trans. Robot.*, vol. 33, no. 2, pp. 249–265, Apr. 2017.

[17] K. Sun et al., "Robust stereo visual inertial odometry for fast autonomous flight," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 965–972, Apr. 2018.

[18] L. Meier, P. Tanskanen, F. Fraundorfer, and M. Pollefeys, "Pixhawk: A system for autonomous flight using onboard computer vision," in *Proc. 2011 IEEE Int. Conf. Robotics and Automation*, pp. 2992–2997.

[19] L. Meier, D. Honegger, and M. Pollefeys, "Px4: A node-based multithreaded open source robotics framework for deeply embedded platforms," in *Proc. 2015 IEEE Int. Conf. Robotics and Automation (ICRA)*, pp. 6235–6240.

[20] A. Barrientos, L. Pen, C. Balaguer, and R. Aracil, *Fundamentos de Robtica*. New York: McGraw-Hill, 1997.

[21] A. Goldenberg, B. Benhabib, and R. Fenton, "A complete generalized solution to the inverse kinematics of robots," *IEEE J. Robot. Autom.*, vol. 1, no. 1, pp. 14–20, Mar. 1985.

[22] N. Courty and E. Arnaud, "Inverse kinematics using sequential Monte Carlo methods," in *Articulated Motion and Deformable Objects*, F. J. Perales and R. B. Fisher, Eds. Berlin: Springer-Verlag, 2008, pp. 1–10.

[23] C. W. Wampler and L. J. Leifer, "Applications of damped least-squares methods to resolved-rate and resolved-acceleration control of manipulators," *J. Dyn. Syst. Meas. Control*, vol. 110, no. 1, pp. 31–38, Mar. 1988.

[24] S. R. Buss, "Introduction to inverse kinematics with Jacobian transpose, pseudoinverse and damped least squares methods," *IEEE Trans. Robot. Autom.*, vol. 17, no. 1, pp. 1–19, 2004.

[25] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. Cambridge, MA: MIT Press, 2005.

Pablo Ramon Soria, Group of Robotics, Vision, and Control, University of Seville, Spain. Email: prs@us.es.

B.C. Arrue, Group of Robotics, Vision, and Control, University of Seville, Spain. Email: barrue@us.es.

Anibal Ollero, Group of Robotics, Vision, and Control, University of Seville, Spain. Email: aollero@us.es.