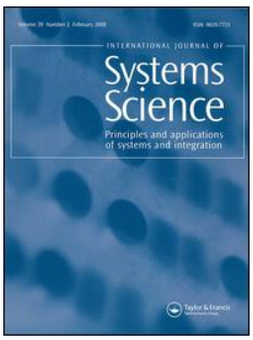

International Journal of Systems Science

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/tsys20

# Control for cooperative transport of a bar-shaped payload with rotorcraft UAVs including a landing stage on mobile robots

Javier Gimenez , Lucio R. Salinas , Daniel C. Gandolfo , Claudio D. Rosales & Ricardo Carelli

**To cite this article:** Javier Gimenez , Lucio R. Salinas , Daniel C. Gandolfo , Claudio D. Rosales & Ricardo Carelli (2020): Control for cooperative transport of a bar-shaped payload with rotorcraft UAVs including a landing stage on mobile robots, International Journal of Systems Science, DOI: 10.1080/00207721.2020.1815096

**To link to this article:** https://doi.org/10.1080/00207721.2020.1815096

Published online: 02 Sep 2020.

INTERNATIONAL JOURNAL OF SYSTEMS SCIENCE
https://doi.org/10.1080/00207721.2020.1815096

Taylor & Francis
Taylor & Francis Group



# Control for cooperative transport of a bar-shaped payload with rotorcraft UAVs including a landing stage on mobile robots

Javier Gimenez, Lucio R. Salinas, Daniel C. Gandolfo, Claudio D. Rosales and Ricardo Carelli

Instituto de Automática, Universidad Nacional de San Juan-CONICET, San Juan, Argentina

**ABSTRACT**

This paper considers the cooperative transport problem of a cable-suspended rigid bar with two rotorcraft UAVs considering collision avoidance, weight distribution, and wind perturbations. The proposal is based on null-space theory and includes a landing stage where the load is settled down on a formation of ground robots. Therefore, a variable formation is considered according to the task phase. In order to simulate the proposal in a realistic environment, very complete dynamic models for UAVs, ground robots, and load are considered. An adaptation stage is incorporated to link the control actions with the inputs required by the dynamic models. Theoretical bounds for the errors are studied under the assumption that perfect velocity tracking is not fulfilled.

**ARTICLE HISTORY**
Received 12 March 2019
Accepted 20 August 2020

**KEYWORDS**
Bar-shaped payload; cooperative transport; error bounds; multi-objective control; rotorcraft UAVs

## 1. Introduction

In recent years, payload transport using UAVs is having a great boom due to its high potential for several applications including supplying medicines and medical samples (Braun et al., 2019; Meier & Soesilo, 2014), food and package delivery (Wing Medium, 2020), assisting in diverse search and rescue scenarios (Bernard et al., 2011), fighting wildfires (Aydin et al., 2019; Saikin et al., 2020), and moving construction material (Augugliaro et al., 2014). There are companies that already offer UAVs specifically designed for load transport,[1] and the biggest package delivery companies are developing UAV-based systems (e.g. UPS, FedEx and DHL). Moreover, some studies suggest that delivering packages with drones may benefit the environment (Stolaroff et al., 2018). In Yoo et al. (2018) the authors analyses the benefits of delivery services in urban and rural areas using UAVs. However, the development of UAVs for load transport must still overcome several constraints: social (habits, thefts, sabotages, etc.), legislative (airspace use, assignment of responsibilities, etc.), structural (homes prepared to receive such orders), and technological (greater energy storage capacity (Gandolfo et al., 2017)).

The flight characteristics of the UAVs are modified by the load causing several difficulties that are being addressed in the literature. The resulting load-UAVs system has strong nonlinearities, complex couplings, and high levels of under-actuation. The difficulty is such that some authors subdivide the problem into three stages: takeoff, navigation and delivery (Cruz & Fierro, 2017), and for simplicity some authors have studied it under restrictive simplifications such as movement constrained to a vertical plane (Pizetta et al., 2016; Potter et al., 2015; Tang, 2014).

Payload transport using cables allows the generation of flexible, adaptable and agile multi-UAV systems (Liang et al., 2018; Palunko et al., 2012). An analysis of the optimal configuration for the transport system (cable length, relative positioning of the vehicles, use or non-use of a spreader bar, among others) can be found in Enciu and Horn (2017). For simplicity, the cables are commonly considered inflexible and inelastic (Almeida, 2014; Goodarzi et al., 2015; Guerrero et al., 2015). A more realistic alternative is to model them as a system of serially-connected links with spherical joints and mass concentrated at the end of each link (Gimenez et al., 2018; Kotaru et al., 2018; Salinas et al., 2018).

**CONTACT** Javier Gimenez jgimenez@inaut.unsj.edu.ar

Another frequent assumption is that the load is a point-mass, which simplifies the modelling and control of the system (Bernard et al., 2011; Gimenez et al., 2018; Raffo & de Almeida, 2016; Villa et al., 2018) but neglects the rotational dynamic effects of the payload. There are several papers on aerial transport without considering point-mass payloads (Collard & Cardou, 2013; Gassner et al., 2017; Tagliabue et al., 2017; Tognon et al., 2018). However, they do not consider the weight distribution between UAVs, which is a crucial factor taking into account the energy limitations of the UAVs, and the possibility of forming heterogeneous formations.

To the best of authors knowledge, there are no papers dealing the landing of a load transported by UAVs on a formation of land robots that move according to the current load pose.

Based on null-space theory, a control scheme was formulated in Gimenez et al. (2018) to transport cooperatively with two UAVs a cable-suspended and point-mass payload considering the properly weight distribution, wind perturbations, and collision avoidance. The present article extends what was done in Gimenez et al. (2018) considering a bar-shaped payload and modifying the priority scheme to improve the control performance. Besides, a landing stage is included, complementary to the navigation stage, in which the bar is placed on two ground robots. In this stage, the ground robots are incorporated to the formation, and they cooperate for the proper load delivery. As in the previous work, the kinematic controllers are based on the null-space theory, but they are adapted to the new system configuration. In order to test the proposal in a realistic scenario, the models used for the UAVs and cables are the same models implemented in Gimenez et al. (2018), while the bar model and MobileSim® (for the simulation of land vehicles) are included specifically for this article.

The document is organised in the following manner: Section 2 formulates the problem and Section 3 defines the variables that characterise the proposed tasks. The cooperative control law and control errors analysis are detailed in Section 4. The Simulation testbed and the implementation of the kinematic formation controller is explained in Section 5. A detailed and complete simulation of the proposal considering realistic dynamic models for the UAVs, ground robots and load is shown in Section 6. Finally, discussions are presented in Section 7 and conclusions are given in Section 8.

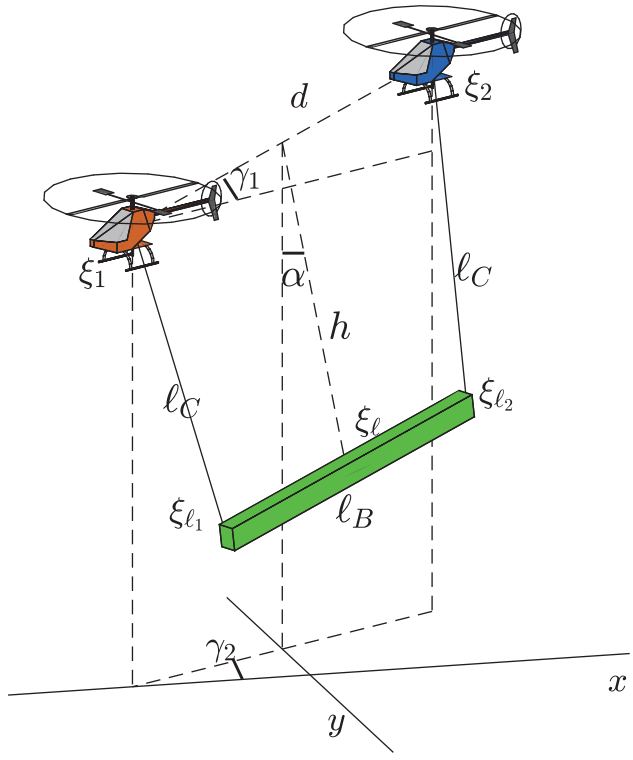


**Figure 1.** Bar-UAVs system: Geometric representation.

## 2. Problem formulation

Consider two rotorcraft UAVs with gravity centres (g.c.) in $\xi_i = [x_i, y_i, z_i]^T$ and yaw orientations $\psi_i$, which are kinematically modelled as a point-mass without roll and pitch movements. These rotational dynamics will be included in the adaptation stage explained in Section 5. Suppose both UAVs cooperatively transport a bar-shaped payload using cables of length $\ell_C$ that join the UAVs with the points $\xi_{\ell_1}$ and $\xi_{\ell_2}$ of the bar, respectively. The distance between these joint points is $\ell_B$ (see Figure 1). Without loss of generality, consider that the joint points $\xi_{\ell_i}$ are in the bar ends, although the same results could be achieved assuming that they are one on each side of the bar g.c. The formation must transport the load in a first phase of navigation, and later, in a landing phase, the load must be properly settled down on two ground robots.

In the navigation stage, the system configuration is given by $\mathbf{q} = [\xi_1^T, \psi_1, \xi_2^T, \psi_2]^T$, and the formation

must fulfill tasks sorted according to the following priorities that set which objectives can be omitted in critical situations:

(1) Collision avoidance.
(2) The secondary objectives are:
  (2.1) Safety distances between UAVs, which must be in a predetermined distance range to avoid collisions or undesirable separation.
  (2.2) The load weight must be properly distributed between the UAVs according to principles such as equity, on-board energies, or load capacities.
  (2.3) Path following in $z$ for the bar g.c. (the tasks 2.2 and 2.3 must have the same priority level to avoid conflicts between them).
(3) The tertiary objectives regulate the absolute and relative $x$-$y$ position of the UAVs to reduce oscillations caused by external factors such as wind:
  (3.1) The bar g.c. must follow a desired $x$-$y$ trajectory.
  (3.2) The bar yaw orientation must be tangential to this predetermined $x$-$y$ trajectory (unlike Gimenez et al., 2018, the same priority order is assigned to the tasks 3.1 and 3.2 in order to obtain better coordination).
(4) The lower priority objectives are the yaw orientations of the UAVs.

The landing stage begins when the UAVs detect the ground robots and there are no obstacles nearby. From that moment on, two ground robots with positions $\xi_{m_i} = [x_{m_i}, y_{m_i}]^T$ are incorporated into the formation, and the new system configuration is $[\mathbf{q}^T, \mathbf{q}_m^T]$, with $\mathbf{q}_m = [\xi_{m_1}^T, \xi_{m_2}^T]^T$. The following tasks are necessary to fulfill the landing stage satisfactorily:

(1) Safety distances between UAVs: Idem to the navigation stage.
(2) Path following for the midpoint between the ground robots.
(3) Distance between ground robots equal to $\ell_B$.
(4) The bar g.c. must follow a desired $x$-$y$-$z$ trajectory, where the $x$-$y$ trajectory must coincide with the current midpoint between the ground robots, and the $z$ trajectory must follow a decreasing and smooth function.
(5) The desired yaw orientation of the bar must be tangent to the desired $x$-$y$ trajectory.
(6) Relative orientation between ground robots equal to the yaw orientation of the bar.
(7) The desired pitch orientation of the bar must be null to reach a uniform contact with the ground robots (this task regulates partially the weight distribution considering low velocities).

All of these tasks can be fulfilled simultaneously in a free-obstacle environment.

This paper defines the kinematic velocities $\mathbf{v}_c$ required to fulfill as many tasks as possible, for then provide the dynamic control actions necessary to reach these velocities.

## 3. Task variables and their Jacobians

The task variables allows the straightforward definition of a kinematic controller $\mathbf{v}_c$ due to the fact they characterise the desired objectives and their time derivatives are linearly related to $\dot{\mathbf{q}}$ and $\dot{\mathbf{q}}_m$ through Jacobians. These variables are classified in five groups: UAVs formation, payload, collision avoidance, UAVs orientation, and variables of the ground robots formation. In practice, the bar performs several oscillations simultaneously (lateral, frontal and rotational), which must be reduced through coordinated movements between the UAVs. These movements are commanded by control actions based on the linear relation existing between the derivative of the task variables and $\dot{\mathbf{q}}$. The resulting Jacobians can be quite complex if certain simplifying assumptions are not made in order to generate consistent control actions without distorting the problem. In this paper, it is supposed that the bar performs only lateral pendular oscillations characterised by the angle $\alpha$ depicted in Figure 1. This implies that, the bar and the virtual segment that links the UAVs have the same yaw and pitch orientations, which is not valid in practice due to the deformation and elasticity of the cables. However, it allows for generating simple control actions in appropriate directions, since variations in the yaw and pitch orientation of the virtual segment imply rotations in the same direction for the bar (see Section 7).

For simplicity, denote respectively with $c_\beta$ and $s_\beta$ the functions $\cos\beta$ and $\sin\beta$, where $\beta$ represents each of the angles considered in the article.

### 3.1. UAVs formation variables

The UAVs formation is determined by (see Figure 1) the distance between the UAVs

$$d = \|\xi_1 - \xi_2\|,$$

and the pitch and yaw angles of the virtual segment that links them

$$\gamma_1 = \operatorname{asin}\left(\frac{z_2 - z_1}{d}\right), \quad \text{and}$$
$$\gamma_2 = \operatorname{atan}\left(\frac{y_2 - y_1}{x_2 - x_1}\right).$$

Their time derivatives are given by

$$\dot{d} = \mathbf{J}_d \dot{\mathbf{q}}, \tag{1}$$
$$\dot{\gamma}_1 = \mathbf{J}_{\gamma_1} \dot{\mathbf{q}}, \tag{2}$$
$$\dot{\gamma}_2 = \mathbf{J}_{\gamma_2} \dot{\mathbf{q}}, \tag{3}$$

where

$$\mathbf{J}_d = \mathbf{J}_d(\mathbf{q}) = \frac{1}{d}\begin{bmatrix} x_1 - x_2 & y_1 - y_2 & z_1 - z_2 & 0 \\ x_2 - x_1 & y_2 - y_1 & z_2 - z_1 & 0 \end{bmatrix},$$

$$\mathbf{J}_{\gamma_1} = \mathbf{J}_{\gamma_1}(\mathbf{q}) = \frac{1}{\sqrt{d^2 - (z_2 - z_1)^2}}\begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix} + \frac{z_1 - z_2}{d\sqrt{d^2 - (z_2 - z_1)^2}}\mathbf{J}_d,$$

$$\mathbf{J}_{\gamma_2} = \mathbf{J}_{\gamma_2}(\mathbf{q}) = \frac{1}{(x_2 - x_1)^2 + (y_2 - y_1)^2} \times \begin{bmatrix} y_2 - y_1 & x_1 - x_2 & 0 & 0 \\ y_1 - y_2 & x_2 - x_1 & 0 & 0 \end{bmatrix}.$$

### 3.2. Payload variables

Another variable of interest is the bar g.c. position $\xi_\ell = [x_\ell, y_\ell, z_\ell]^T$, which can be expressed as convex combination of the joint points, i.e. there is $0 < \lambda < 1$ such that $\xi_\ell = (1 - \lambda)\xi_{\ell_1} + \lambda\xi_{\ell_2}$. If both UAVs fly at the same altitude ($\gamma_1 = 0$), then the joint points are given by

$$\xi_{\ell_1} = \frac{\xi_1 + \xi_2}{2} + \frac{\ell_B}{2d}(\xi_1 - \xi_2) + h\begin{bmatrix} s_\alpha s_{\gamma_2} \\ -s_\alpha c_{\gamma_2} \\ -c_\alpha \end{bmatrix}, \tag{4}$$

$$\xi_{\ell_2} = \frac{\xi_1 + \xi_2}{2} + \frac{\ell_B}{2d}(\xi_2 - \xi_1) + h\begin{bmatrix} s_\alpha s_{\gamma_2} \\ -s_\alpha c_{\gamma_2} \\ -c_\alpha \end{bmatrix}, \tag{5}$$

with $h = h(\mathbf{q}) = \sqrt{\ell_C^2 - (d - \ell_B)^2/4}$ (see Figure 1). When $\gamma_1 \neq 0$, it is assumed that $\xi_{\ell_1}$ and $\xi_{\ell_2}$ are also calculated by using these expressions. This practical assumption prevents that any error in the load altitude is corrected by elevating only one UAV, and in this way, this task is not coming into conflict with the task of adequate weight distribution (see Gimenez et al., 2018 for more details).

The velocities of the joint points are given by

$$\dot{\xi}_{\ell_i} = \mathbf{J}_{\ell_i}\dot{\mathbf{q}} + \mathbf{A}_\ell, \tag{6}$$

where

$$\mathbf{J}_{\ell_1} = \begin{bmatrix} \mathbf{J}_{\ell_1,x} \\ \mathbf{J}_{\ell_1,y} \\ \mathbf{J}_{\ell_1,z} \end{bmatrix} = \mathbf{J}_{\ell_1}(\mathbf{q}, \alpha) = \frac{1}{2}[\mathbf{I}_3 \quad \mathbf{0}_{3\times 1} \quad \mathbf{I}_3 \quad \mathbf{0}_{3\times 1}] + \frac{\ell_B}{2d}[-\mathbf{I}_3 \quad \mathbf{0}_{3\times 1} \quad \mathbf{I}_3 \quad \mathbf{0}_{3\times 1}] + \frac{d}{4h}\begin{bmatrix} -s_\alpha s_{\gamma_2} \\ s_\alpha c_{\gamma_2} \\ c_\alpha \end{bmatrix}\mathbf{J}_d + hs_\alpha\begin{bmatrix} c_{\gamma_2} \\ s_{\gamma_2} \\ 0 \end{bmatrix}\mathbf{J}_{\gamma_2},$$

$$\mathbf{J}_{\ell_2} = \begin{bmatrix} \mathbf{J}_{\ell_2,x} \\ \mathbf{J}_{\ell_2,y} \\ \mathbf{J}_{\ell_2,z} \end{bmatrix} = \mathbf{J}_{\ell_2}(\mathbf{q}, \alpha) = \frac{1}{2}[\mathbf{I}_3 \quad \mathbf{0}_{3\times 1} \quad \mathbf{I}_3 \quad \mathbf{0}_{3\times 1}] + \frac{\ell_B}{2d}[\mathbf{I}_3 \quad \mathbf{0}_{3\times 1} \quad -\mathbf{I}_3 \quad \mathbf{0}_{3\times 1}] + \frac{d}{4h}\begin{bmatrix} -s_\alpha s_{\gamma_2} \\ s_\alpha c_{\gamma_2} \\ c_\alpha \end{bmatrix}\mathbf{J}_d + hs_\alpha\begin{bmatrix} c_{\gamma_2} \\ s_{\gamma_2} \\ 0 \end{bmatrix}\mathbf{J}_{\gamma_2},$$

$$\mathbf{A}_\ell = \begin{bmatrix} A_{\ell,x} \\ A_{\ell,y} \\ A_{\ell,z} \end{bmatrix} = \mathbf{A}_\ell(\mathbf{q}, \alpha, \dot{\alpha}) = h\begin{bmatrix} c_\alpha s_{\gamma_2} \\ -c_\alpha c_{\gamma_2} \\ s_\alpha \end{bmatrix}\dot{\alpha}.$$

Then, the velocity of the bar g.c. is given by

$$\dot{\xi}_\ell = \mathbf{J}_\ell\dot{\mathbf{q}} + \mathbf{A}_\ell, \tag{7}$$

where

$$\mathbf{J}_\ell = \begin{bmatrix} \mathbf{J}_{\ell,x} \\ \mathbf{J}_{\ell,y} \\ \mathbf{J}_{\ell,z} \end{bmatrix} = \mathbf{J}_\ell(\mathbf{q}, \alpha) = (1 - \lambda)\mathbf{J}_{\ell_1} + \lambda\mathbf{J}_{\ell_2}.$$

### 3.3. Variable of obstacle avoidance

Consider $N$ dynamic obstacles in the environment with positions

$$\xi_{o,i} = \xi_{o,i}(t) = [x_{o,i}(t), y_{o,i}(t), z_{o,i}(t)]^T, \quad 1 \leq i \leq N.$$

The obstacle avoidance is done analogously to Gimenez et al. (2018) but considering as formation centre

$$\xi_c = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \frac{\xi_1 + \xi_2 + \xi_\ell}{3},$$

whose time derivative is given by

$$\begin{aligned} \dot{\xi}_c &= \frac{\dot{\xi}_1 + \dot{\xi}_2 + \dot{\xi}_\ell}{3} \\ &= \frac{1}{3}\left(\mathbf{J}_\ell + \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3\times 1} & \mathbf{I}_3 & \mathbf{0}_{3\times 1} \end{bmatrix}\right)\dot{\mathbf{q}} + \frac{1}{3}\mathbf{A}_\ell. \end{aligned}$$

The task variable for collision avoidance is

$$V = V(\mathbf{q}, \alpha, \{\xi_{o,i}\}) = \sum_{i=1}^{N} V_i,$$

where each $V_i$ is the potential field generated by each obstacle, which is defined by

$$\begin{aligned} V_i &= V_i(\mathbf{q}, \alpha, \xi_{o,i}) \\ &= \exp\left\{ -\frac{(x_c - x_{o,i})^n}{a_x} - \frac{(y_c - y_{o,i})^n}{a_y} \right. \\ &\quad \left. - \frac{(z_c - z_{o,i})^n}{a_z} \right\}, \end{aligned} \tag{8}$$

with $a_x > 0$, $a_y > 0$, $a_z > 0$ and $n > 0$ (even number) design parameters. In practice, it is considered $V_i = 0$ when the obstacle is not close, i.e. $V_i < \varsigma$ with $\varsigma > 0$. Then

$$\dot{V} = \mathbf{J}_V \dot{\mathbf{q}} + \mathbf{A}_V, \tag{9}$$

where

$$\mathbf{J}_V = \mathbf{J}_V(\mathbf{q}, \alpha, \{\xi_{o,i}\}) = \sum_{i=1}^{N} \mathbf{J}_{V_i},$$

and

$$\mathbf{A}_V = \mathbf{A}_V(\mathbf{q}, \alpha, \dot{\alpha}, \{\xi_{o,i}\}, \{\dot{\xi}_{o,i}\}) = \sum_{i=1}^{N} \mathbf{A}_{V_i},$$

with

$$\begin{aligned} \mathbf{J}_{V_i} &= \mathbf{J}_{V_i}(\mathbf{q}, \alpha, \xi_{o,i}) \\ &= -\frac{1}{3}\mathbf{a}_i^T \left(\mathbf{J}_\ell + \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_{3\times 1} & \mathbf{I}_3 & \mathbf{0}_{3\times 1} \end{bmatrix}\right), \end{aligned}$$

$$\mathbf{A}_{V_i} = \mathbf{A}_{V_i}(\mathbf{q}, \alpha, \dot{\alpha}, \xi_{o,i}, \dot{\xi}_{o,i}) = \mathbf{a}_i^T \left(\dot{\xi}_{o,i} - \frac{1}{3}\mathbf{A}_\ell\right),$$

$$\mathbf{a}_i = \mathbf{a}_i(\mathbf{q}, \alpha, \xi_{o,i}) = nV_i \begin{bmatrix} \dfrac{(x_c - x_{o,i})^{n-1}}{a_x} \\ \dfrac{(y_c - y_{o,i})^{n-1}}{a_y} \\ \dfrac{(z_c - z_{o,i})^{n-1}}{a_z} \end{bmatrix}.$$

### 3.4. UAVs orientation variables

The orientation variables are the yaw orientations $\psi_1$ and $\psi_2$ of the UAVs, whose respective Jacobians are

$$\mathbf{J}_{\psi_1} = [0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0 \quad 0], \tag{1}$$

$$\mathbf{J}_{\psi_2} = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1]. \tag{10}$$

### 3.5. Variables of the ground robots formation

Finally, the ground robots formation is determined by (see Figure 2) the midpoint

$$\xi_m = \frac{\xi_{m_1} + \xi_{m_2}}{2},$$

the distance between robots

$$d_m = \|\xi_{m_1} - \xi_{m_2}\|,$$

and the relative orientation between them

$$\gamma_m = \operatorname{atan}\left(\frac{y_{m_2} - y_{m_1}}{x_{m_2} - x_{m_1}}\right).$$

Their time derivatives are given by

$$\dot{\xi}_m = \mathbf{J}_m \dot{\mathbf{q}}_m, \quad \dot{d}_m = \mathbf{J}_{d_m} \dot{\mathbf{q}}_m, \quad \dot{\gamma}_m = \mathbf{J}_{\gamma_m} \dot{\mathbf{q}}_m.$$

where

$$\begin{aligned} \mathbf{J}_m &= \frac{1}{2}\begin{bmatrix} \mathbf{I}_2 & \mathbf{I}_2 \end{bmatrix}, \\ \mathbf{J}_{d_m} &= \frac{1}{d_m}\begin{bmatrix} x_{m_1} - x_{m_2} & y_{m_1} - y_{m_2} \\ x_{m_2} - x_{m_1} & y_{m_2} - y_{m_1} \end{bmatrix}, \\ \mathbf{J}_{\gamma_m} &= \frac{1}{d_m^2}\begin{bmatrix} y_{m_2} - y_{m_1} & x_{m_1} - x_{m_2} \\ y_{m_1} - y_{m_2} & x_{m_2} - x_{m_1} \end{bmatrix}. \end{aligned}$$

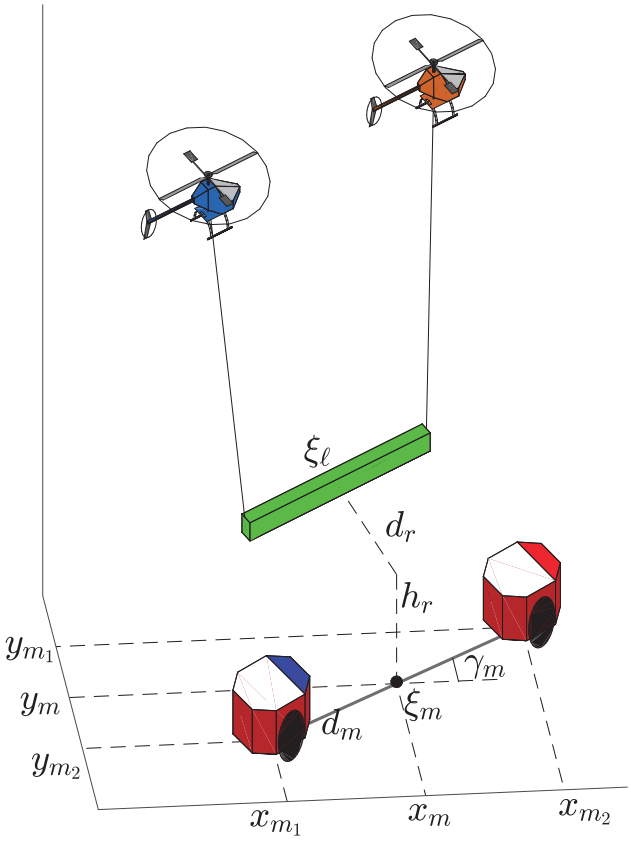


**Figure 2.** Variables of the ground robots formation and the landing stage.

## 4. Kinematic formation controller

The kinematic velocities $\mathbf{v}_c$ and $\mathbf{v}_{m,c}$ for the state vectors $\mathbf{q}$ and $\mathbf{q}_m$ required to fulfill the desired objectives with their priorities are given in this Section. These velocities are based on null-space theory, which works projecting desired velocities of lower priority tasks on the null-space of the Jacobian of the higher priority tasks. This scheme keeps the system stable respecting priorities when there are obstacles nearby, and allows the simultaneous fulfillment of all tasks in a free-obstacle environment.

### 4.1. Control objectives and their priorities in the navigation stage

Navigating by free-obstacle zones ($V = 0$) is the first objective for the navigation stage. However, it is also desired that $\dot{V} = 0$ when entering in an obstacle zone ($V_i \geq \varsigma$ for any $i$) to avoid collisions by navigating through the level curves of $V$. Consider the control action

$$\mathbf{v}_c^{(1)} = \mathbf{J}_1^{\dagger}(-k_V V - \mathbf{A}_V), \tag{11}$$

where $\mathbf{J}_1 = \mathbf{J}_1(\mathbf{q}, \alpha, \{\xi_{o,i}\}) = \mathbf{J}_V$ is the Jacobian of task 1, $k_V > 0$ is a design constant, and the superscript $^{\dagger}$ represents the pseudoinverse matrix. Then, if $\mathbf{v}_c^{(1)} = \dot{\mathbf{q}}$ (perfect velocity tracking), then from (9),

$$\dot{V} = \mathbf{J}_V \dot{\mathbf{q}} + \mathbf{A}_V = \mathbf{J}_V \mathbf{v}_c^{(1)} + \mathbf{A}_V = -k_V V,$$

and therefore, $V(t) \to 0$. Besides, any velocity vector of the form

$$\mathbf{v}_c = \mathbf{v}_c^{(1)} + (\mathbf{I}_8 - \mathbf{J}_1^{\dagger}\mathbf{J}_1)\mathbf{w}, \quad \mathbf{w} \in \mathbb{R}^8, \tag{12}$$

satisfies the task since $\mathbf{J}_1 \mathbf{v}_c = \mathbf{J}_1 \mathbf{v}_c^{(1)}$. Taking advantage of this, a second task with lower priority and associated Jacobian $\mathbf{J}_2$ can be simultaneously satisfied by considering $\mathbf{w} = \mathbf{v}_c^{(2)}$ when both tasks are not conflictive ($\mathbf{J}_2 \mathbf{J}_1^{\dagger} = \mathbf{0}$), since $\mathbf{J}_2 \mathbf{v}_c = \mathbf{J}_2 \mathbf{v}_c^{(2)}$. Otherwise, task 2 will be performed in the best possible way without conditioning task 1. This idea is recursively extended to satisfy more than two tasks sorted by priorities. Section 4.3 analyses error bounds when not assuming a perfect velocity tracking.

The second objective for the navigation stage is a proper weigh distribution maintaining a safety distance $d_m \leq d \leq d_M$ between the UAVs while the bar g.c. remains at a preset altitude.

For pulling effects caused by the load, the UAV behind makes a lesser effort if both UAVs navigate at the same altitude ($\gamma_1 = 0$). Then, the UAVs should regulate the altitude difference ($\gamma_1$) to compensate for this situation, or more generally, to transport weight proportions $(\rho, 1-\rho)$, $\rho \in (0,1)$. Then, the desired elevation angle is defined by

$$\gamma_1^* = \gamma_1 + k_{\gamma_1,1} \tanh\left(k_{\gamma_1,2}\left(\frac{\breve{f}_1}{\rho} - \frac{\breve{f}_2}{1-\rho}\right)\right),$$

where $k_{\gamma_1,1}, k_{\gamma_1,2} > 0$ are design parameters, and $\breve{f}_i$ is a smoothed version of $f_i = \|F_i\|$, with $F_i$ the force generated by the bar on the $i$th UAV. Thus, the control objective is $\tilde{\gamma}_1(t) = \gamma_1^*(t) - \gamma_1(t) \to 0$. The weight proportions can be defined on-line proportional to the on-board energy availability or load capacities of the UAVs. This fact represents an important advantage of the proposed control approach.

In order to navigate maintaining a safety distance and generating control actions only when $d \notin$

$[d_m, d_M]$, it is defined

$$\tilde{d} = \begin{cases} d_m - d & \text{if } d < d_m, \\ 0 & \text{if } d_m \leq d \leq d_M, \\ d_M - d & \text{if } d > d_M. \end{cases} \quad (13)$$

The objective of desired altitude $z^*(t)$ for the bar g.c. is satisfied by requiring that $\tilde{z}_\ell(t) = z^*(t) - z_\ell(t) \to 0$.

Note that from (1), (2) and (7), it results

$$\begin{bmatrix} \dot{d} \\ \dot{\gamma}_1 \\ \dot{z}_\ell \end{bmatrix} = \mathbf{J}_2 \dot{\mathbf{q}} + \mathbf{A}_2, \quad (14)$$

where

$$\mathbf{J}_2 = \mathbf{J}_2(\mathbf{q}, \alpha) = \begin{bmatrix} \mathbf{J}_d \\ \mathbf{J}_{\gamma_1} \\ \mathbf{J}_{\ell,z} \end{bmatrix}, \quad \text{and}$$

$$\mathbf{A}_2 = \mathbf{A}_2(\mathbf{q}, \alpha, \dot{\alpha}) = \begin{bmatrix} 0 \\ 0 \\ A_{\ell,z} \end{bmatrix}.$$

Then, the second task is fulfilled by using

$$\mathbf{v}_c^{(2)} = \mathbf{J}_2^\dagger \begin{bmatrix} k_{d,1} \tanh(k_{d,2}\tilde{d}) \\ \dot{\gamma}_1^* + \tilde{\gamma}_1 \\ \dot{z}^* + k_{\ell,z,1} \tanh(k_{\ell,z,2}\tilde{z}_\ell) - A_{\ell,z} \end{bmatrix}, \quad (15)$$

with $k_{d,1}, k_{d,2}, k_{\ell,z,1}, k_{\ell,z,2} > 0$ design constants.

The third objective requires that the bar g.c. tracks a desired $x$-$y$ trajectory, and that the bar is always tangential to it. For them, the UAVs must navigate one behind the other in such a way that $\tilde{x}_\ell(t) = x^*(t) - x_\ell(t) \to 0$, $\tilde{y}_\ell(t) = y^*(t) - y_\ell(t) \to 0$, and $\tilde{\gamma}_2(t) = \gamma_2^*(t) - \gamma_2(t) \to 0$, with $\gamma_2^* = \text{atan2}(\dot{y}_\ell^*, \dot{x}_\ell^*)$. In practice, for better bar controllability, $\tilde{\gamma}_2$ is calculated assuming that the yaw orientation of the bar is $\gamma_2$. From (3) and (7), it results

$$\begin{bmatrix} \dot{x}_\ell \\ \dot{y}_\ell \\ \dot{\gamma}_2 \end{bmatrix} = \mathbf{J}_3 \dot{\mathbf{q}} + \mathbf{A}_3, \quad (16)$$

where

$$\mathbf{J}_3 = \mathbf{J}_3(\mathbf{q}, \alpha) = \begin{bmatrix} \mathbf{J}_{\ell,x} \\ \mathbf{J}_{\ell,y} \\ \mathbf{J}_{\gamma_2} \end{bmatrix}, \quad \text{and}$$

$$\mathbf{A}_3 = \mathbf{A}_3(\mathbf{q}, \alpha, \dot{\alpha}) = \begin{bmatrix} A_{\ell,x} \\ A_{\ell,y} \\ 0 \end{bmatrix}.$$

Then, the third task is fulfilled by using

$$\mathbf{v}_c^{(3)} = \mathbf{J}_3^\dagger \begin{bmatrix} \dot{x}^* + k_{\ell,x,1} \tanh(k_{\ell,x,2}\tilde{x}_\ell) - A_{\ell,x} \\ \dot{y}^* + k_{\ell,y,1} \tanh(k_{\ell,y,2}\tilde{y}_\ell) - A_{\ell,y} \\ \dot{\gamma}_2^* + k_{\gamma_2,1} \tanh(k_{\gamma_2,2}\tilde{\gamma}_2) \end{bmatrix}, \quad (17)$$

with $k_{\ell,x,1}, k_{\ell,x,2}, k_{\ell,y,1}, k_{\ell,y,2}, k_{\gamma_2,1}, k_{\gamma_2,2} > 0$ design constants.

The desired yaw orientation for each UAV is defined in the fourth control objective and does not conflict with the other tasks. These desired orientations must be tangential to the respective desired $x$-$y$ trajectories, i.e.

$$\psi_1^*(t) = \text{atan2}(\dot{y}_1^*(t), \dot{x}_1^*(t)), \quad \text{and}$$
$$\psi_2^*(t) = \text{atan2}(\dot{y}_2^*(t), \dot{x}_2^*(t)),$$

where

$$\begin{bmatrix} x_1^*(t) \\ y_1^*(t) \end{bmatrix} = \begin{bmatrix} x_\ell^*(t) \\ y_\ell^*(t) \end{bmatrix} + \frac{d}{2\sqrt{(\dot{x}_\ell^*(t))^2 + (\dot{y}_\ell^*(t))^2}} \begin{bmatrix} \dot{x}_\ell^*(t) \\ \dot{y}_\ell^*(t) \end{bmatrix},$$

$$\begin{bmatrix} x_2^*(t) \\ y_2^*(t) \end{bmatrix} = \begin{bmatrix} x_\ell^*(t) \\ y_\ell^*(t) \end{bmatrix} - \frac{d}{2\sqrt{(\dot{x}_\ell^*(t))^2 + (\dot{y}_\ell^*(t))^2}} \begin{bmatrix} \dot{x}_\ell^*(t) \\ \dot{y}_\ell^*(t) \end{bmatrix},$$

are the desired $x$-$y$ trajectories for the UAVs considering $d = (d_m + d_M)/2$ for simplicity.

From these definitions and from (10), the fourth objective is fulfilled by using

$$\mathbf{v}_c^{(4)} = \mathbf{J}_4^\dagger \begin{bmatrix} \dot{\psi}_1^* + k_{\psi_1,1} \tanh(k_{\psi_1,2}\tilde{\psi}_1) \\ \dot{\psi}_2^* + k_{\psi_2,1} \tanh(k_{\psi_2,2}\tilde{\psi}_2) \end{bmatrix}, \quad (18)$$

where $\tilde{\psi}_1 = \psi_1^* - \psi_1$ and $\tilde{\psi}_2 = \psi_2^* - \psi_2$ are the task errors, $k_{\psi_1,1}, k_{\psi_1,2}, k_{\psi_2,1}, k_{\psi_2,2} > 0$ are design constants, and

$$\mathbf{J}_4 = \begin{bmatrix} \mathbf{J}_{\psi_1} \\ \mathbf{J}_{\psi_2} \end{bmatrix}.$$

Thus, the null-space based control law that satisfies as many objectives as possible respecting priorities is

given by

$$\mathbf{v}_c = \mathbf{v}_c^{(1)} + \left(\mathbf{I}_8 - \mathbf{J}_1^{\dagger}\mathbf{J}_1\right) \times \left(\mathbf{v}_c^{(2)} + \left(\mathbf{I}_8 - \mathbf{J}_2^{\dagger}\mathbf{J}_2\right) \times \left(\mathbf{v}_c^{(3)} + \left(\mathbf{I}_8 - \mathbf{J}_3^{\dagger}\mathbf{J}_3\right)\mathbf{v}_c^{(4)}\right)\right). \quad (19)$$

A detailed analysis of task compatibility and the tuning of the parameters can be found in Gimenez et al. (2018).

### 4.2. Control objectives for the landing stage

The objective of safety distance between UAVs is solved as in the navigation stage. As regards the path following objective for the midpoint between the ground robots, consider the desired path $\xi_m^*(s) = [x_m^*(s), y_m^*(s)]^T$. The orientation of the tangent to the path is given by

$$\theta^*(s) := \operatorname{atan2}\left(\frac{\partial \xi_m^*}{\partial y^*}(s), \frac{\partial \xi_m^*}{\partial x^*}(s)\right).$$

Then, there is a function $s = s(t)$ such as

$$s(t) := \arg\min_s \|\xi_m(t) - \xi_m^*(s)\|.$$

That is, $\xi_m^*(s(t))$ is the point on the curve $\xi_m^*(s)$ closest to $\xi_m(t)$ at each time $t$. Given a desired velocity for the formation $v_m^* > 0$, the control objectives is $\|\xi_m(t) - \xi_m^*(s(t))\| \to 0$.

In the landing stage, the reference trajectory for the bar g.c. is redefined considering $\xi_m(t)$ as

$$\xi^*(t) = \begin{bmatrix} \xi_m(t) \\ z_m^*(t) \end{bmatrix},$$

where $z_m^*$ is defined in order to reach a smooth and precise payload landing. Given a landing altitude $h_r$ and the distance $d_r = \|\xi_\ell - [\xi_m^T, h_r]^T\|$ to the landing point, this paper considers

$$z_m^* = \max\left\{h_r + \frac{z_\ell - h_r}{1 + d_r}, h_r, z_\ell - 1\right\},$$

which is the maximum between three terms (see Figure 2). The first term is considered to slow down the descent when the payload is close to landing, the second term is a lower bound, and the third term avoids distant altitude reference.

The coordination between the yaw orientation of the bar and the position of the ground robots can take place in two different ways:

(1) The UAVs orient the bar according to $\theta^*(s)$ while the ground robots adjust its relative orientation $\gamma_m$ according to the yaw orientation of the bar.
(2) The UAVs orient the bar according to $\gamma_m$ while the ground robots follow their desired path with relative orientation $\gamma_m = \theta^*(s)$.

In this paper, the first option is chosen since in this way a cooperative task between the UAVs and the ground robots is achieved. Besides, it is easier to control the relative yaw orientation $\gamma_m$ between the ground robots than the yaw orientation of the bar, due to potential oscillations of the bar.

Finally, it is considered $\gamma_1^* = 0$ for reaching a uniform landing of the bar on the ground robots. In practice, in the landing stage, $\tilde{\gamma}_1 = \gamma_1^* - \gamma_1$ is also calculated using the pitch orientation of the bar as $\gamma_1$.

All the objectives proposed for the landing stage can be fulfilled simultaneously considering the following kinematic controller,

$$\begin{bmatrix} \mathbf{v}_c \\ \mathbf{v}_{c,m} \end{bmatrix} = \mathbf{J}_\lambda^{\dagger} \begin{bmatrix} k_{d,1}\tanh(k_{d,2}\tilde{d}) \\ k_{\gamma_1,1}\tanh(-k_{\gamma_1,2}\gamma_1) \\ \dot{z}_m^* + k_{\ell,z,1}\tanh(k_{\ell,z,2}(z_m^* - z_\ell)) - A_{\ell,z} \\ \dot{x}_m^* + k_{\ell,x,1}\tanh(k_{\ell,x,2}(x_m^* - x_\ell)) - A_{\ell,x} \\ \dot{y}_m^* + k_{\ell,y,1}\tanh(k_{\ell,y,2}(y_m^* - y_\ell)) - A_{\ell,y} \\ \dot{\theta}^*(s) + k_{\gamma_2,1}\tanh(k_{\gamma_2,2}(\theta^*(s) - \gamma_2)) \\ \dot{\psi}_1^* + k_{\psi_1,1}\tanh(k_{\psi_1,2}\tilde{\psi}_1) \\ \dot{\psi}_2^* + k_{\psi_2,1}\tanh(k_{\psi_2,2}\tilde{\psi}_2) \\ \mathbf{v}^* + \mathbf{K}_{m,1}\tanh(\mathbf{K}_{m,2}(\xi_m^* - \xi_m)) \\ k_{d_m,1}\tanh(k_{d_m,2}(\ell_B - d_m)) \\ \dot{\gamma}_2 + k_{\gamma_m,1}\tanh(k_{\gamma_m,2}(\gamma_2 - \gamma_m)) \end{bmatrix}, \quad (20)$$

where

$$\mathbf{J}_\lambda = \mathbf{J}_\lambda(\mathbf{q}, \mathbf{q}_m, \alpha) = \begin{bmatrix} \mathbf{J}_2 \\ \mathbf{J}_3 \\ \mathbf{J}_4 \\ \mathbf{J}_{\lambda,m} \end{bmatrix}, \quad \mathbf{J}_{\lambda,m} = \begin{bmatrix} \mathbf{J}_m \\ \mathbf{J}_{d_m} \\ \mathbf{J}_{\gamma_m} \end{bmatrix},$$

$k_{d_m,1}, k_{d_m,2}, k_{\gamma_m,1}, k_{\gamma_m,2} > 0$ are design constants, $\mathbf{K}_{m,1}$, $\mathbf{K}_{m,2} > 0$ are design matrices, and $\mathbf{v}^* = v_m^*[\cos\theta^*, \sin\theta^*]^T$.

### 4.3. Error bounds

Generally $\mathbf{v}_c \neq \dot{\mathbf{q}}$ and $\mathbf{v}_{m,c} \neq \dot{\mathbf{q}}_m$ since each vehicle has its dynamics. Then, the stability of the error system requires an appropriate velocity tracking. This Section found error bounds assuming that $\|\tilde{\mathbf{v}}\|_\infty < \infty$ and $\|\tilde{\mathbf{v}}_m\|_\infty < \infty$, where $\tilde{\mathbf{v}} := \mathbf{v}_c - \dot{\mathbf{q}}$ and $\tilde{\mathbf{v}}_m := \mathbf{v}_{m,c} - \dot{\mathbf{q}}_m$ are the velocity tracking errors.

Consider the collision avoidance, which is the primary objective in the navigation stage. From (9), (11) and (19), $\dot{V} = \mathbf{J}_1(\mathbf{v}_c - \tilde{\mathbf{v}}) + \mathbf{A}_V = -k_V V - \mathbf{J}_1\tilde{\mathbf{v}}$. As the potential functions and the obstacle observations are bounded, it follows that $\|\mathbf{J}_1\|_\infty < \infty$, and thus, $\|\mathbf{J}_1\tilde{\mathbf{v}}\|_\infty < \infty$. Let $\mathcal{V} = \frac{1}{2}V^2$ be an energy function with time derivative $\dot{\mathcal{V}} = V\dot{V} = -k_V V^2 - V\mathbf{J}_1\tilde{\mathbf{v}}$. If $k_V V > \|\mathbf{J}_1\tilde{\mathbf{v}}\|_\infty$ then $\dot{\mathcal{V}} < 0$, and thus, $V$ is ultimately bounded by

$$V \leq \frac{\|\mathbf{J}_1\tilde{\mathbf{v}}\|_\infty}{k_V}.$$

The velocity tracking error $\tilde{\mathbf{v}}$ and the design constant $k_V$ regulate the error bound and the obstacle repulsion.

Regarding task 2, a requirement for task stability is that it is not conflicting with the higher priority tasks. A bound for $\tilde{d}$ is only analysed since the other cases are similar. From $\mathbf{J}_d\mathbf{J}_1^\dagger = 0$, it results $\mathbf{J}_d\mathbf{v}_c = \mathbf{J}_d\mathbf{v}_c^{(2)} = k_{d,1}\tanh(k_{d,2}\tilde{d})$. Besides, $\|\mathbf{J}_d\|_\infty = 1$ from the definition of $\mathbf{J}_d$. Let $\mathcal{V} = \frac{k_{d,1}}{k_{d,2}}\ln(\cosh(k_{d,2}\tilde{d}))$ be an energy function, for which

$$\begin{aligned}\dot{\mathcal{V}} &= k_{d,1}\tanh(k_{d,2}\tilde{d})\dot{\tilde{d}} = -k_{d,1}\tanh(k_{d,2}\tilde{d})\dot{d}\\ &= -k_{d,1}\tanh(k_{d,2}\tilde{d})\mathbf{J}_d(\mathbf{v}_c - \tilde{\mathbf{v}})\\ &= -k_{d,1}\tanh(k_{d,2}\tilde{d})(k_{d,1}\tanh(k_{d,2}\tilde{d}) - \mathbf{J}_d\tilde{\mathbf{v}}).\end{aligned}$$

If $|k_{d,1}\tanh(k_{d,2}\tilde{d})| > \|\mathbf{J}_d\tilde{\mathbf{v}}\|_\infty$ then $\dot{\mathcal{V}} < -k_{d,1}^2\tanh^2(k_{d,2}\tilde{d}) < 0$. Thus, $\tilde{d}$ is ultimately bounded with bound regulated by $k_{d,1}$ and $k_{d,2}$, since

$$|\tanh(k_{d,2}\tilde{d})| \leq \frac{\|\mathbf{J}_d\tilde{\mathbf{v}}\|_\infty}{k_{d,1}}.$$

The bound for the other tasks are similarly found if they are not conflictive with higher-priority tasks. In particular, for path or trajectory tracking tasks, it is not verified that $\mathbf{J}_\ell\mathbf{J}_1^\dagger = 0$. That is, the obstacle avoidance is conflictive with following a trajectory or path. However, it defines $\mathbf{J}_1 = \mathbf{0}$ when $V = 0$, and thus, $\mathbf{J}_\ell\mathbf{J}_1^\dagger = 0$ for free-obstacle environment. The same procedure can be used to find the error bounds for the tasks corresponding to the landing stage. Thus, ultimately bounded stability of the system depends on the correct setting of the UAV on-board control and the correct parameters selection.

## 5. Simulation testbed and implementation of the kinematic formation controller

The main simulation code is developed in MATLAB® (The MathWorks Inc., Massachusetts, USA), and a custom C++ code is design to communicate by shared memory the main code with the C++ codes used to simulate the dynamics of UAVs, cables, and bar. The models used for the UAVs and cables are the same models implemented in Gimenez et al. (2018), while the bar model and MobileSim® (for the simulation of land vehicles) are included specifically for this article.

The UAV model used is a 6DoF dynamic model of a mini-helicopter named MIT's X-Cell .60 extracted from Gavrilets (2003) with nominal parameters. It adequately represents the mini-helicopter dynamics in both hovering and low-speed flight envelope (up to 20 [m/s] forward flight). The model considers wind effects and non-ideal dynamics such as flapping, drag, and actuator dynamics (see Gavrilets, 2003; Gimenez et al., 2018 for more details). For each UAV, the connection between the input commands for the dynamic model $C_{md,i}$ and the kinematic formation control law $\mathbf{v}_{c,i} = [\dot{x}_{i,c}, \dot{y}_{i,c}, \dot{z}_{i,c}, \dot{\psi}_{i,c}]^T$ is made via an adaptation stage conformed by two stages: a velocity frame change and a PID-based cascade control (see details in Gandolfo et al., 2016; Gimenez et al., 2018; Salinas et al., 2014). An important advantage of this proposal is that the same controller can be used for other types of miniature rotorcraft (such as quadrotors (Gandolfo et al., 2017)) by only modifying the adaptation stage.

The cables are simulated with forty point-like masses joined by springs and dampers (Kelvin-Voigt models) for the swinging and waving motion and simple 3D solid objects for the drag. The masses are affected by elastic and viscous forces (caused by the surrounding springs and dampers), drag forces, air friction forces, gravitational forces, and other ground related forces (absorption, friction and repulsion). The first mass of each cable is attached to a spherical joint below the corresponding helicopter g.c, and the last links of the cables are joined between them to simulate

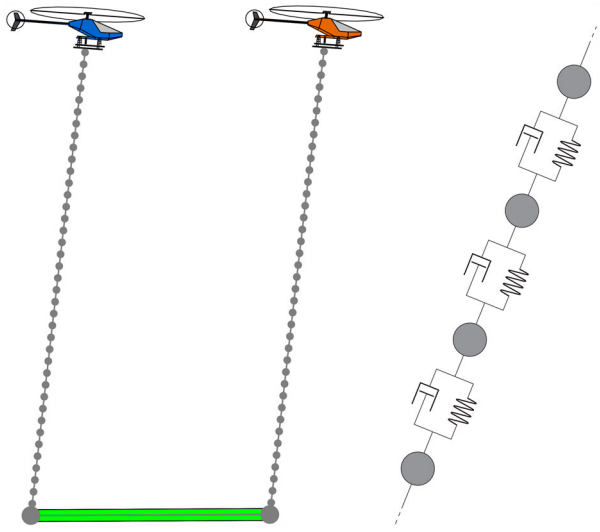

**Figure 3.** Mini-helicopters transporting a cable-suspended bar (left) and cable section represented by mass-spring-damper systems (right).

**Table 1.** Parameters of the cables and payload.

| Parameters | Value |
|---|---|
| Number of cables | 2 |
| Number of masses per cable | 39 |
| Cable diameter | 0.004 [m] |
| Cable link mass | 0.0033 [kg] |
| Cable drag coefficient | 1.0 |
| Bar mass | 4.75 [kg] |
| Bar length | 0.2 [m] |
| Bar drag coefficient | 0.82 |
| Spring length | 0.1 [m] |
| Spring friction | 1 [N·s/m] |
| Spring rate | 40,000 [N/m] |
| Gravitational acceleration | 9.7917 [m/s$^2$] |
| Air friction | 0.002 [N·s/m] |
| Air density | 1.151 [kg/m$^3$] |
| Ground absorption | 0.2 [N·s/m] |
| Ground friction | 0.02 [N·s/m] |
| Ground repulsion | 10 [N/m] |

the bar-shaped payload using a spring with high constant (see Figure 3). The bar mass is distributed equally among the masses of the bar ends. A list of all the parameters used in the simulation is given in Table 1. For more details see Gimenez et al. (2018). A detailed video of the dynamic evolution of the cable-bar system can be found in https://youtu.be/7gE_n6b5-LM.

Finally, the dynamics of the ground robots is included by using MobileSim® with default setting, which also communicates (receiving speed commands and sending vehicle states) with the main code by shared memory. The commands $\mathbf{v}_{m,c,i}$ for the ground robots are transformed in the linear and angular reference velocities required by MobileSim.

## 6. Simulation results

It is known that following an eight-shaped trajectory is challenging for any controller, even more for a system with a high degree of under-actuation. For this reason, the trajectory to be followed by the bar g.c. during the navigation stage has been selected as $\xi_\ell^*(t) = [12\cos(\pi t/T), -12\sin(2\pi t/T), 14 - 8t/T]^T$, with $T = 90$ s. Meanwhile, the ground robots are controlled so that the midpoint between them $\xi_m$ follows a circular path of radius 9m at a constant velocity of 0.5 m/s. The landing stage begins when the distance between the bar g.c. and the landing point (with $h_r = 0.4$ m) is less than 10 m. During this stage, the ground robots continue with the path following task using the same reference values. To incorporate the wind effects, a constant wind with intensity 5.5 m/s is introduced from $t = 30$ s to $t = 90$ s. Besides, the obstacles $\xi_{o,1} = [0, 0, 10]^T$ (static) and $\xi_{o,2} = \xi_{o,2}(t) = [2\cos(16\pi\, t/T)+10{,}2\sin(16\pi t/T) - 5{,}14]^T$ (dynamic) are considered. Each helicopter must carry half the load weight, except for the first 45 s where helicopter 1 transports 3/5 of the weight. The design parameters are set to: $k_V = 0.03$, $a_x = a_y = 2.5$, $a_z = 4$, $n = 2$, $\varsigma = 0.001$, $k_{d,1} = 0.5$, $k_{d,2} = 0.75$, $k_{\gamma_2,1} = 0.5$, $k_{\gamma_2,2} = 0.75$, $k_{\ell,x,1} = k_{\ell,y,1} = k_{\ell,z,1} = 0.5$, $k_{\ell,x,2} = k_{\ell,y,2} = k_{\ell,z,2} = 0.375$, $k_{\psi_1,1} = k_{\psi_2,1} = 0.5$, $k_{\psi_1,2} = k_{\psi_2,2} = 0.75$, $k_{d_m,1} = k_{d_m,2} = k_{\gamma_m,1} = k_{\gamma_m,2} = 1$, $\mathbf{K}_{m,1} = 0.5\mathbf{I}_3$, and $\mathbf{K}_{m,2} = 1.5\mathbf{I}_3$. Besides, it is considered $k_{\gamma_1,1} = 0.02$ and $k_{\gamma_1,2} = 0.2$ for navigation stage, and $k_{\gamma_1,1} = 0.5$ and $k_{\gamma_1,2} = 0.75$ for landing stage due to the sensibility difference between controlling the pitch orientation of the bar and controlling the difference of forces performed by the UAVs.

Figure 4 shows different views of the simulated results with the system state at times $t = 0$ s (start of simulation), $t = 7$ s (detection of the dynamic obstacle), $t = 30$ s (wind introduction), $t = 45$ s (change in the weight proportion lifted by each UAV and avoidance of the static obstacle), $t = 90$ s (end of the wind), $t = 134.1$ s (switch to landing stage), and $t = 167.2$ s (the bar is settled down on the formation of ground robots). A video of the simulation can be seen in https://youtu.be/NRGLAvV47Hk. Figure 5 shows the norm of the forces exerted by each UAV and their filtered versions. Figure 6 shows the error evolution for each task grouped by navigation priorities. Finally, Figure 7 shows a comparison between control and actual velocities in the global frame. The fact that the

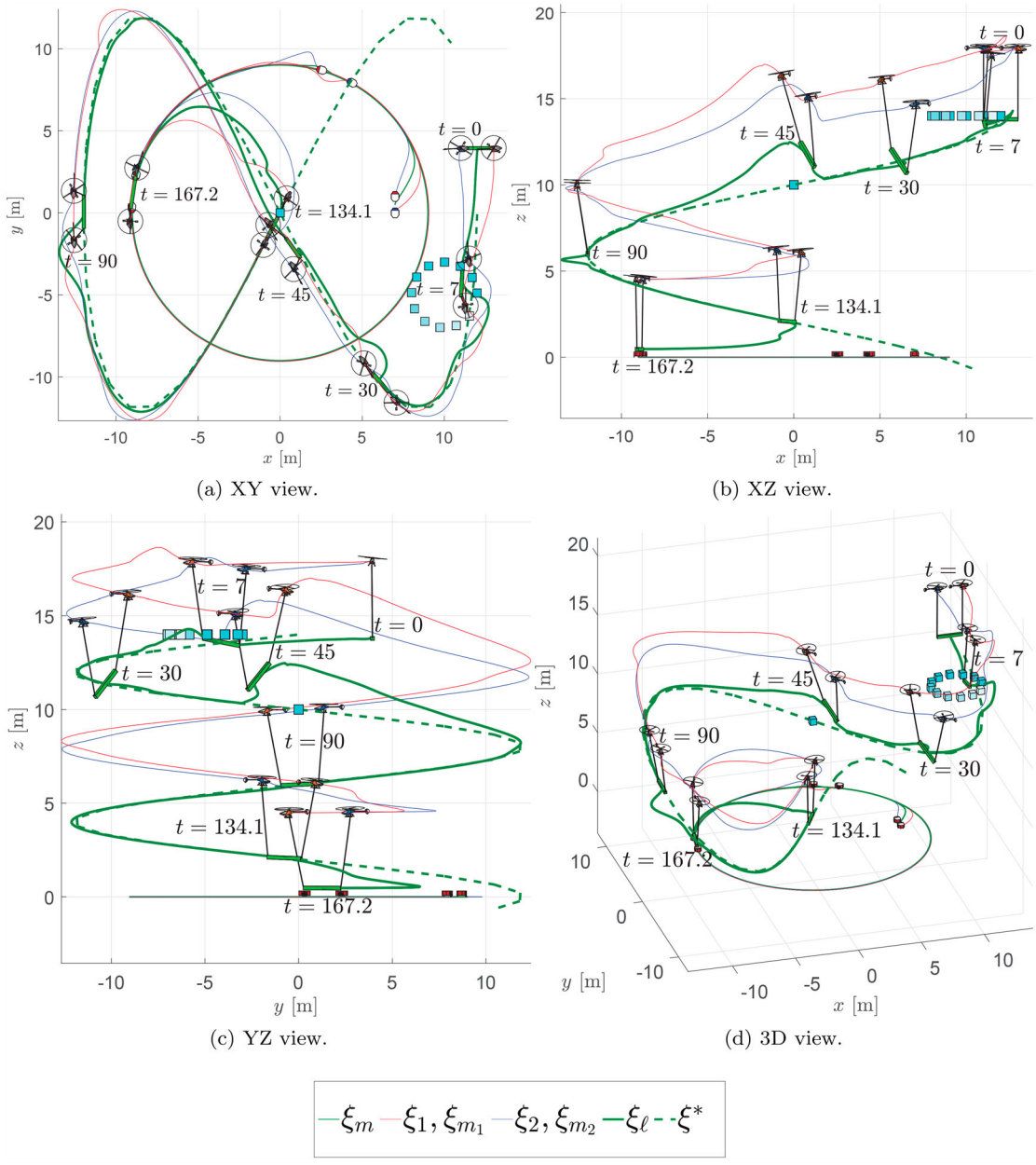


**Figure 4.** Simulation results considering wind perturbation and collision avoidance. (a) XY view. (b) XZ view. (c) YZ view and (d) 3D view.

velocity tracking errors are small is crucial to keep the control errors bounded. In these figures, the wind presence (shaded zone), change in the weight proportion assigned for each UAV (red dashed line), and the switch between stages (blue dashed line) are highlighted. While after the stage switch at time $t = 134.1$ there are task redefinitions (and the respective errors), for simplicity, the error curves continue to be plotted in the same graphs of Figure 6. References changes produce discontinuities in the error curves (see blue dashed line).

The curve of $V$ in Figure 6 discriminates the obstacle avoidance stages in which, respectively, a dynamic (whose movement is sketched with cubes coloured in shades of blue) and a static obstacle are avoided. A collision is highlighted under these circumstances when $V = e^0 = 1$, which is well above the observed $V$-values.

The noise level of the force exerted by the UAVs and the need to filter them can be shown in Figure 5. This noise also requires small gains $k_{\gamma_1,i}$, since otherwise the oscillatory effects are transferred to the force

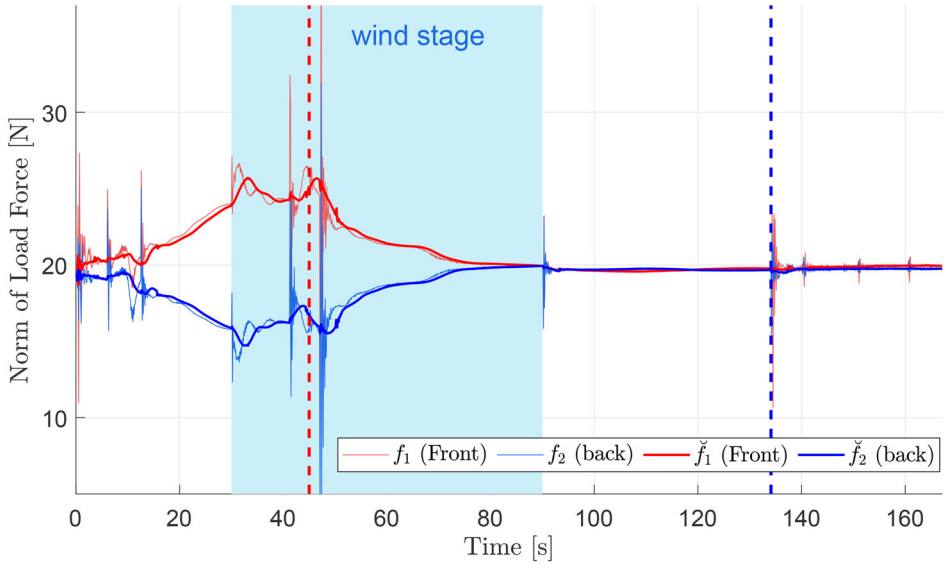


**Figure 5.** Forces exerted by each UAV and their filtered versions.

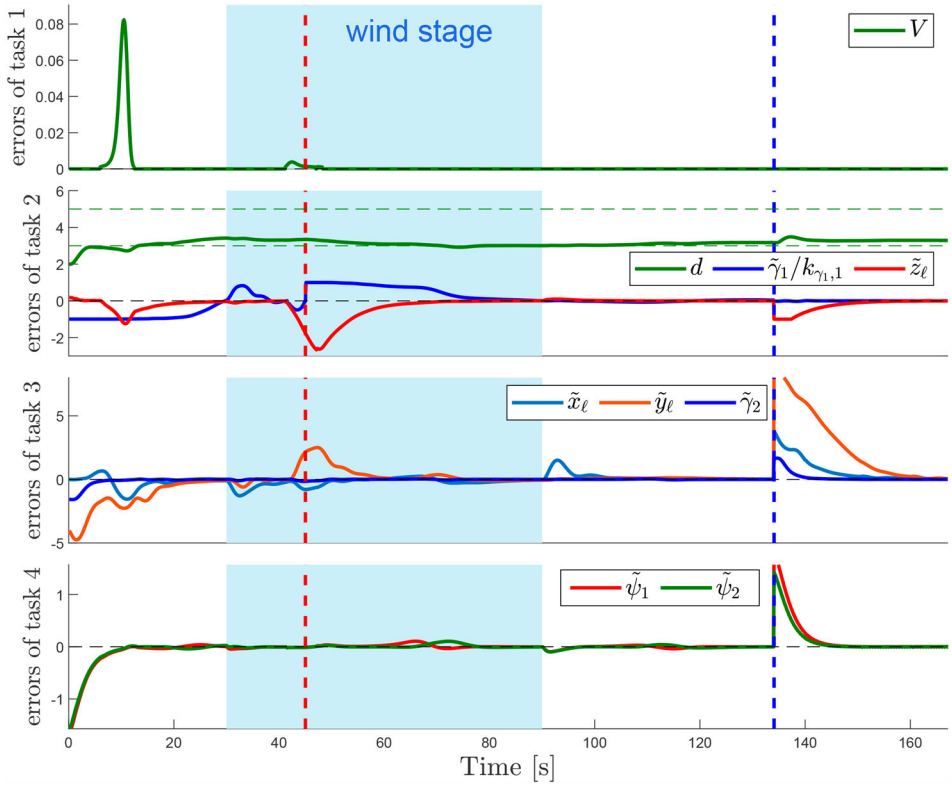


**Figure 6.** Task errors grouped according to the priority levels in the navigation stage.

distribution control. That is why a slow evolution of the weight distribution is observed when a reference change is made (see Figure 5). Besides, the obstacle avoidance (priority task) and the wind do not allow the complete reaching of the proportions $(3/5, 2/5)$ proposed. However, during the first 45 s, the bar is carried inclined in order to reach the desired weight distribution (see Figure 4). Note that the weight distribution during the landing stage is indirectly controlled by the pitch control for the virtual segment that links the UAVs.

The distance between UAVs remains within the specified security range $[3, 5]$ except for small time intervals in which the controller acts quickly to

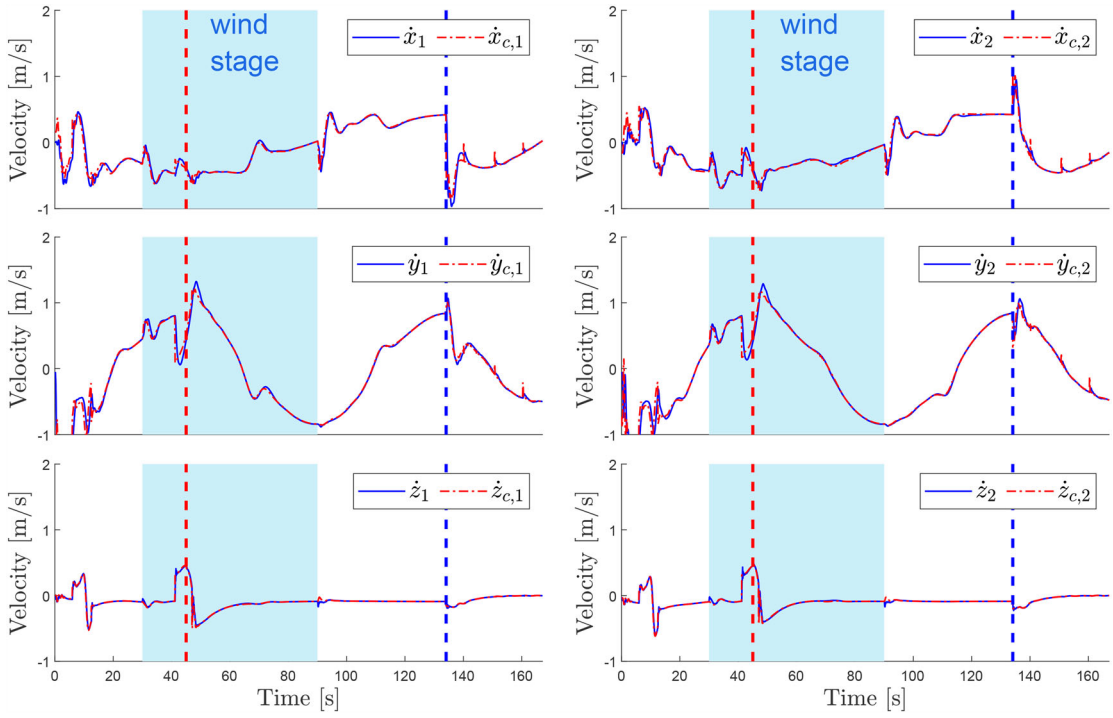


**Figure 7.** Velocity tracking: Comparison between control and actual velocities in the global frame.

improve errors (see green dashed lines in the second plot of Figure 6).

Figures 4 and 6 show that the trajectory tracking task for the load is fulfilled very satisfactorily except in the obstacle avoidance stages. The yaw orientation error of the bar is minimal even during the obstacle avoidance, remaining almost immutable to the wind effects. Besides, these errors converge rapidly during the landing stage achieving the desired positioning for the bar on the ground robots. The yaw orientation errors of the UAVs are also small despite being tasks of lower priority.

## 7. Discussions

The control law was calculated assuming that the bar and the virtual segment that joins both UAVs have the same yaw and pitch orientations. The assumption allowed to generate a simple controller that fulfills adequately the objectives, since the desired directions of movement for the UAVs to control the position and orientation of the bar are practically independent of the level of flexibility of the cables. On the other hand, the adaptation stage has not a model-based controller and allows facing the unmodeled dynamics of the UAVs and the UAVs-load interaction (such as wind and pulling effects).

Regarding the generality of the presented results, other UAV architectures can be used with just modifying the adaptation stage since this is a kinematic-based approach. Besides, the task variables defined in Section 3 can be reused for a wide range of applications of mobile robots.

## 8. Conclusions

The paper presents a kinematic controller based on the null-space theory to transport with two UAVs a bar-shaped payload (navigation stage) and positioning it on two ground robots (landing stage). To simulate the proposal in a realistic environment, complete dynamic models for the UAVs, ground robots and the cable-bar system are considered. The simulations show mobility in the load-UAVs system with a realism superior to that found in the literature. An adaptation stage is incorporated to link the control actions with the inputs required by the dynamic models. This stage is conformed by change of frames and PID-based cascade controls. The velocity tracking errors are small enough to earn acceptable task errors for any practical application. The realistic simulation shows the excellent performance of the proposed control law and how this approach allows considering a change in objectives and priorities as part of a complex global task.

Besides, this strategy allows using different types of rotorcraft machines (e.g. quadcopters) by changing only the adaptation stage.

## Note

1. For example: https://wingcopter.com, https://velosuav.com and https://swissdrones.com.

## Disclosure statement

No potential conflict of interest was reported by the author(s).

## Notes on contributors

***Javier Gimenez*** received the B. Sc. degree in Mathematics from the National University of San Juan (UNSJ), Argentina in 2009, and the Ph. D. degree in Mathematics from the National University of Córdoba (UNC), Argentina in 2014. Currently, he is an Assistant Researcher of the National Council for Scientific Research (CONICET) Argentina, and an Adjunct Professor in the Institute of Automatics, UNSJ-CONICET, Argentina. His research interests is the applied mathematic to robotics.

***Lucio R. Salinas*** received the B.Eng. degree in electronic engineering and the Ph.D. degree in control systems engineering from the National University of San Juan (UNSJ), San Juan, Argentina, in 2008 and 2013, respectively. He is an associate researcher at the National Scientific and Technical Research Council (CONICET) and an assistant professor at the Instituto de Automática, UNSJ- CONICET. His research interests include robotics, teleoperation systems, human-machine systems, unmanned aerial vehicles and software development.

***Daniel C. Gandolfo*** graduated in Electronic Engineering with orientation in Control Systems from the National University of San Juan (UNSJ), Argentina in 2006 and has been working as automation engineer in the industry until 2010. He obtained the Ph.D. degree in Control Systems Engineering at the Instituto de Automática, National University of San Juan in 2014. He is currently Professor at the National University of San Juan and Researcher of the National Council for Scientific and Technical Research (CONICET). His research interest included algorithms for management energy systems and optimal control strategies with application in unmanned aerial vehicles (UAV).

***Claudio D. Rosales*** studied Electronic Engineering at the National University of San Juan (UNSJ), Argentine obtaining the degree Electronic Engineer in 2009, and the Ph.D. in Control Systems Engineering from the UNSJ in 2014, and the Ph.D. in Electric Engineering from the Federal University of Espírito Santo, Brazil, in 2018. Currently, he is an assistant researcher of the Council for Scientific and Technological Research, Argentina, and an associate professor in the Institute of Automatic, UNSJ-CONICET. His main interests included algorithms for multi-robot systems, nonlinear control, artificial intelligence, and aerial robotic.

***Ricardo Carelli*** received B. Sc. degree in Engineering from the National University of San Juan, Argentina in 1976, and the Ph.D. degree in Electrical Engineering from the National University of Mexico (UNAM), Mexico in 1989. He is a full professor at the National University of San Juan and a Superior Researcher of the National Council for Scientific and Technical Research, Argentina. He is a Senior Member of IEEE and a Member of Argentine Association of Automatic Control – International Federation of Automatic Control (AADECA-IFAC). He was Director of the Automatics Institute, National University of San Juan, Argentina.

## ORCID

*Javier Gimenez* http://orcid.org/0000-0002-1731-8520

## References

Almeida, M. M. (2014). *Control strategies of a tilt-rotor UAV for load transportation* [Master's thesis, Universidade Federal De Minas Gerais Escola De Engenharia]. http://hdl.handle.net/1843/BUOS-9Q6GFQ

Augugliaro, F., Lupashin, S., Hamer, M., Male, C., Hehn, M., Mueller, M. W., Willmann, J. S., Gramazio, F., Kohler, M., & D'Andrea, R. (2014). The flight assembled architecture installation: Cooperative construction with flying machines. *IEEE Control Systems Magazine*, *34*(4), 46–64. https://doi.org/10.1109/MCS.2014.2320359

Aydin, B., Selvi, E., Tao, J., & Starek, M. J. (2019). Use of fire-extinguishing balls for a conceptual system of drone-assisted wildfire fighting. *Drones*, *3*(1), 17. https://doi.org/10.3390/drones3010017

Bernard, M., Kondak, K., Maza, I., & Ollero, A. (2011). Autonomous transportation and deployment with aerial robots for search and rescue missions. *Journal of Field Robotics*, *28*(6), 914–931. https://doi.org/10.1002/rob.v28.6

Braun, J., Gertz, S. D., Furer, A., Bader, T., Frenkel, H., Chen, J., Glassberg, E., & Nachman, D. (2019). The promising future of drones in prehospital medical care and its application to battlefield medicine. *Journal of Trauma and Acute Care Surgery*, *87*(1S), S28–S34. https://doi.org/10.1097/TA.0000000000002221

Collard, J. F., & Cardou, P. (2013, September). Computing the lowest equilibrium pose of a cable-suspended rigid body. *Optimization and Engineering*, *14*, 457–476. https://doi.org/10.1007/s11081-012-9191-5

Cruz, P. J., & Fierro, R. (2017). Cable-suspended load lifting by a quadrotor UAV: Hybrid model, trajectory generation, and control. *Autonomous Robots*, *41*(8), 1629–1643. https://doi.org/10.1007/s10514-017-9632-2

Enciu, J., & Horn, J. F. (2017, January). *Flight performance optimization of a multi-lift rotorcraft formation*. AIAA atmospheric flight mechanics conference. American Institute of Aeronautics and Astronautics Inc, AIAA.

Gandolfo, D., Salinas, L., Brandão, A., & Toibero, J. (2017). Stable path-following control for a quadrotor helicopter considering energy consumption. *IEEE Transactions on Control Systems Technology*, *25*(4), 1423–1430. https://doi.org/10.1109/TCST.2016.2601288

Gandolfo, D., Salinas, L., Brandão, A., & Toibero, M. (2016). Path following for unmanned rotorcraft – an approach on energy autonomy improvement. *Information Technology and Control*, *45*(1), 87–98. https://doi.org/10.5755/j01.itc.45.1.12413

Gassner, M., Cieslewski, T., & Scaramuzza, D. (2017, May). *Dynamic collaboration without communication: Vision-based cable-suspended load transport with two quadrotors*. IEEE international conference on robotics and automation (pp. 5196–5202). IEEE.

Gavrilets, V. (2003). *Autonomous aerobatic maneuvering of miniature helicopters* [Doctoral dissertation, Massachusetts Institute of Technology. Dept. of Aeronautics and Astronautics]. https://dspace.mit.edu/handle/1721.1/17609

Gimenez, J., Gandolfo, D., Salinas, L., Rosales, C., & Carelli, R. (2018). Multi-objective control for cooperative payload transport with rotorcraft UAVs. *ISA Transactions*, *80*, 491–502. https://doi.org/10.1016/j.isatra.2018.05.022

Goodarzi, F. A., Lee, D., & Lee, T. (2015, December). Geometric control of a quadrotor UAV transporting a payload connected via flexible cable. *International Journal of Control, Automation and Systems*, *13*(6), 1486–1498. https://doi.org/10.1007/s12555-014-0304-0

Guerrero, M. E., Mercado, D. A., Lozano, R., & García, C. D. (2015, December). Passivity based control for a quadrotor UAV transporting a cable-suspended payload with minimum swing. IEEE annual conference on decision and control. IEEE.

Kotaru, P., Wu, G., & Sreenath, K. (2018). *Differential-flatness and control of quadrotor(s) with a payload suspended through flexible cable(s)*. Indian control conference (ICC) (pp. 352–357). IEEE.

Liang, X., Fang, Y., Sun, N., & Lin, H. (2018). Dynamics analysis and time-optimal motion planning for unmanned quadrotor transportation systems. *Mechatronics*, *50*, 16–29. https://doi.org/10.1016/j.mechatronics.2018.01.009

Meier, P., & Soesilo, D. (Eds.) (2014). *Drones in humanitarian action case study No 2: Using drones for medical payload delivery in Papua New Guinea*. Médecins Sans Frontières. https://europa.eu/!HM67uY

Palunko, I., Cruz, P., & Fierro, R. (2012, September). Agile load transportation: Safe and efficient load manipulation with aerial robots. *IEEE Robotics & Automation Magazine*, *19*(3), 69–79. https://doi.org/10.1109/MRA.2012.2205617

Pizetta, I., Brandão, A., & Sarcinelli-Filho, M. (2016, June). *Cooperative quadrotors carrying a suspended load*. Proceedings of the international conference on unmanned aircraft systems (ICUAS). IEEE.

Potter, J. J., Adams, C. J., & Singhose, W. (2015, October). A planar experimental remote-controlled helicopter with a suspended load. *IEEE/ASME Transactions on Mechatronics*, *20*(5), 235–247. https://doi.org/10.1109/TMECH.2014.2386801

Raffo, G. V., & de Almeida, M. M. (2016, July). *Nonlinear robust control of a quadrotor UAV for load transportation with swing improvement*. Proceedings of the American control conference (ACC). IEEE.

Saikin, D. A., Baca, T., Gurtner, M., & Saska, M. (2020). Wildfire fighting by unmanned aerial system exploiting its time-varying mass. *IEEE Robotics and Automation Letters*, *5*(2), 2673–2680. https://doi.org/10.1109/LSP.2016

Salinas, L., Gimenez, J., Rosales, C., & Gandolfo, D. (2018, June). *Null-space-based path-following control for cooperative payload transport using multiple rotorcraft UAVs*. Proceedings of the international conference on unmanned aircraft systems. IEEE.

Salinas, L., Slawiński, E., & Mut, V. (2014). Kinematic nonlinear controller for a miniature helicopter via Lyapunov techniques. *Asian Journal of Control*, *16*(3), 856–870. https://doi.org/10.1002/asjc.2014.16.issue-3

Stolaroff, J. K., Samaras, C., O´Neill, E. R., Lubers, A., Mitchell, A. S., & Ceperley, D. (2018). Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery. *Nature Communications*, *9*(409).

Tagliabue, A., Kamel, M., Verling, S., Siegwart, R., & Nieto, J. (2017, May). *Collaborative transportation using MAVs via passive force control*. IEEE international conference on robotics and automation (ICRA) (pp. 5766–5773). IEEE.

Tang, S. (2014). *Aggressive maneuvering of a quadrotor with a cable-suspended payload* [Ph.D. dissertation]. Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, PA.

Tognon, M., Gabellieri, C., Pallottino, L., & Franchi, A. (2018). Aerial co-manipulation with cables: The role of internal force for equilibria, stability, and passivity. *IEEE Robotics and Automation Letters*, *3*, 2577–2583. https://doi.org/10.1109/LRA.2018.2803811

Villa, D. K., Brandão, A. S., & Sarcinelli-Filho, M. (2018). *Load transportation using quadrotors: A survey of experimental results*. Proceedings of the international conference on unmanned aircraft systems (pp. 84–93). IEEE.

Wing Medium (2020). *Thank you Canberra – one year of air delivery*. Retrieved 2020-04-08, from https://link.medium.com/Q860hpidx5

Yoo, W., Yu, E., & Jung, J. (2018). Drone delivery: Factors affecting the public's attitude and intention to adopt. *Telematics and Informatics*, *35*(6), 1687–1700. https://doi.org/10.1016/j.tele.2018.04.014