3D Formation Control of Autonomous Vehicles Based on Null-Space

Claudio Rosales, Paulo Leica, Mario Sarcinelli-Filho, Gustavo Scaglia & Ricardo Carelli

Journal of Intelligent & Robotic Systems
with a special section on Unmanned Systems

ISSN 0921-0296

J Intell Robot Syst
DOI 10.1007/s10846-015-0329-5

ONLINE FIRST
February 2016 Vol. 81 · No. 2
ISSN: 0921-0296
Journal of Intelligent & Robotic Systems
with a special section devoted to Unmanned Systems
Editor-in-Chief: Kimon P. Valavanis
Special Issue: Current Developments in Robotics and Mobile Machines
Guest Editors: Massimo Callegari and Primo Zingaretti
Springer

J Intell Robot Syst
DOI 10.1007/s10846-015-0329-5

3D Formation Control of Autonomous Vehicles Based on Null-Space

Claudio Rosales · Paulo Leica · Mario Sarcinelli-Filho · Gustavo Scaglia · Ricardo Carelli

Received: 16 December 2014 / Accepted: 28 December 2015

Abstract This paper proposes a new algorithm for controlling a formation of multiple autonomous aerial vehicles based on multiple control objectives. The strategy includes using the null space of a Jacobian matrix to achieve the different control objectives in a non-conflicting way. The mission is split into two elementary tasks, with suitable command references generated for each robot. The commands for each task are combined through a hierarchical method by using the projection of commands onto the null space. The incorporation of ground vehicles in the control scheme is also considered, thus extending the proposed scheme for controlling heterogeneous formations. The stability analysis of the control system shows that such a system is asymptotically stable, and experimental results validate the proposed control system.

Keywords Aerial formation control · Null space · Quadrotor

C. Rosales (✉) · P. Leica · G. Scaglia · R. Carelli
Instituto de Automatica, Universidad Nacional de San Juan,
Av. Libertador San Martin, 1112 (O), San Juan, Argentine
e-mail: crosales@inaut.unsj.edu.ar

P. Leica
e-mail: pleica@inaut.unsj.edu.ar

G. Scaglia
e-mail: gscaglia@inaut.unsj.edu.ar

R. Carelli
e-mail: rcarelli@inaut.unsj.edu.ar

M. Sarcinelli-Filho
Universidade Federal do Espirito Santo, Av. Fernando
Ferrari, 514, Goiabeiras, Vitória-ES, Brazil
e-mail: mario.sarcinelli@ufes.br

1 Introduction

Formation control of aerial vehicles has become an area of great interest in the robotics community. This new field is motivated by the possibility to accomplish tasks that a single vehicle is not able to perform, by means of replacing it by a group of aerial robots of smaller capability. Effectively, the use of multiple robots has several advantages, such as cost reduction, increased strength, better performance and efficiency [18]. Instead of designing a single powerful robot, a multi-robot system (MRS) can be designed, generally, in a simpler and more economical way [12]. The ability to control vehicles working cooperatively is a main challenge for researchers of robotics and artificial intelligence areas. Interesting results have been published in the literature [19].

An aerial formation can be defined as a set of two or more aerial vehicles flying together, whose dynamic states are coupled through a common control law. This coupling can be expressed in terms of translational and/or rotational degrees of freedom, as well as position or velocity [16].

Published online: 14 January 2016

A preliminary classification of multi-robot systems based on the type of control establishes them as centralized or decentralized systems. The main characteristic of a centralized system is that a single processing unit makes decisions and communicates with all vehicles in the team. Since the central unit stores a lot of information, it is more efficient for obtaining solutions. A disadvantage, however, lies on the fact that the central unit will be continuously receiving data from the robots, with the possibility of causing congestion by imputing large data volume, therefore reducing the speed of decision making. In the worst case, if the central unit fails, the whole system will stop operating, because the robots are not capable of making any decision by themselves [8]. On the other hand, in a decentralized system, each UAV (Unmanned Aerial Vehicle) is capable of communicating and sharing information among themselves. However, and in general, since each vehicle is assigned to do a specific portion of the whole objective, each aircraft will be able to accomplish only that part of the control objective, with no knowledge about the other robots' tasks. This type of control system is more dynamic and faster without needing to send out or to store information. Other advantages are greater robustness in case of an UAV failure, system scalability, fewer communication requirements and distributed computational effort.

Basically, in the literature, there are three structures for controlling multi-robot systems, namely leader-follower structure, behaviour-based methods and virtual structures, each of them with their respective advantages and disadvantages. For example, in the leader-follower structure, a robot is considered the leader while the other ones are followers [9, 10]. In this structure each follower deals its information with the leader, but knows nothing about the other followers' information. The leader has no information about the followers either. Therefore, if the leader fails there will be no way to ensure that the control objective be reached. However, this structure is easy to understand and implement. In the second case, the behaviour-based structure, the behaviour of the UAV formation is defined as a combination of individual behaviour of each member [6]. The main question of this approach is how to formalize it mathematically. In difficult setting, it may not be ease to ensure the convergence of the formation to the desired objectives. In the third approach, the virtual structures imply establishing geometric relationships that will remain rigid between the robots and the referential system, which can be a virtual point or a virtual robot. Nevertheless, an advantage of this method is that the virtual leader never fails. Therefore, the formation will be kept during the entire task.

A control scheme based on a virtual structure is presented in [11], called cluster space control. The position control (or trajectory-tracking control) is performed by considering the centroid of the geometric structure (a triangle) correspondent to a formation of three robots on the plane. In [1], such a control scheme is extended to non-planar robots, but considering a formation of just two robots. In [7], an inherent problem of centralized control systems, i.e. its scalability, is addressed. More specifically a technique was developed to allow a generalization, which not discussed in [11], by extending the control approach to a team of $n > 3$ robots. Obstacle avoidance by the formation was also considered, by enabling the structure be momentarily modified, thus allowing an flexible behaviour of the formation while moving.

Nowadays, robots tasks require a large volume of real-time computation, generally involving multiple subtasks, such as manipulation, exploration, obstacle avoidance, etc. This means that different control objectives should be achieved at the same time, sometimes causing conflicts of interest among them and the pre-assigned priority order. In [2], a number of control schemes are discussed, which split the control problem into several sub-problems that are eventually solved individually. Therefore, the primary and most important objective will be considered a minimum norm solution obtained by the pseudo-inverse of the Jacobian associated to the problem, whereas the secondary objectives are projected onto the null space of such Jacobian. The main advantage of this control scheme is that it guarantees that the main objective is obtained, while the lower hierarchy objectives are projected onto the null space thus not generating any conflict with the primary objective [4]. This concept was presented in [3] for generic robotic control systems, and in [5] to the control of multi-robot systems.

The present paper proposes to develop a centralized position and trajectory-tracking controller for an $3D$ UAV formation, based on the null space approach. The contribution of the work lies on using the null space-based control technique for controlling a $3D$ virtual structure formation, with the mission being partitioned into several non conflicting tasks for better formation task performance.

2 Null-space Based Control

Generally a mission may involve one or more robots, who individually or collectively achieve a certain amount of tasks at a given instant of time. Several control schemes have been proposed (with their advantages and disadvantages) in the literature. One of them is the behavioural-based control, which proposes split the control objective into several sub-objectives and then solve each of them individually, to finally combine the outputs of each controller to get the command to be sent to the robots.

The null space-based control is a behavioural-based control, in the sense that it considers different control subsystems. Such an approach is derived from the inverse kinematics solution of redundant industrial manipulators. For a redundant system, there are infinite solutions for a single solution, and this fact is used to introduce secondary objectives to be achieved in the null space of the primary task.

Defining $q \in \Re^m$ as the set of variables to be controlled and $x \in \Re^{3n}$ as the positions of each one of the n aerial robots of the formation, the relationship between both sets of variables is defined as

$$q = f_{(x)}, \tag{1}$$

with the corresponding differential relationship

$$\dot{q} = \frac{\partial f_{(x)}}{\partial x}\dot{x} = J_{(x)}\dot{x}, \tag{2}$$

where $J_{(x)} \in \Re^{m\times 3n}$ is the Jacobian matrix associated to such mapping, which relates the robots velocities with the variations of the task variables.

An effective way to generate a reference motion for each robot $x_{d,i(t)}$ from a desired value of the task variables $q_{d,i(t)}$ is to invert the kinematic relationship (2). A typical requirement is to pursue the minimum norm speed, leading to the least-squares solution. For matrices with more columns than rows, which is common in formation control, where $3n > m$, it is defined the right pseudo-inverse matrix

$$\dot{x}_d = J^\dagger \dot{q}_d = J^T(JJ^T)^{-1}\dot{q}_d, \tag{3}$$

where $J^\dagger$ is the right pseudo-inverse of the Jacobian matrix $J_{(x)}$. The reference position can be obtained by integrating $\dot{x}_d$ in (3) and using the closed loop inverse kinematics algorithm

$$\dot{x}_d = J^\dagger(\dot{q}_d + \lambda\tilde{q}), \tag{4}$$

where λ is a positive definite gain matrix used to adjust the response, and $\tilde{q} = q_d - q$ is the error associated to the task.

Projecting each speed onto the null space created by the Jacobian matrix of the higher-level task, the competition between the sub-tasks becomes a solved problem. However, one needs to calculate the desired velocity for each task, given by

$$\dot{x}_{id} = J_i^\dagger(\dot{q}_{id} + \lambda_i\tilde{q}_i), \tag{5}$$

where i denotes the $i - th$ task. The index can also indicate the task priority, so the task number 1 has higher priority, in comparison with the other tasks. Solving each task individually it is possible to combine them to obtain a general solution, similarly to the behavioural-based approach. For the case of two tasks, results

$$\dot{x}_d = J_1^\dagger(\dot{q}_{1d}+\lambda_1\tilde{q}_1)+(I-J_1^\dagger J_1)J_2^\dagger(\dot{q}_{2d}+\lambda_2\tilde{q}_2), \tag{6}$$

where I is an identity matrix of appropriate dimension. Each sub-task is computed individually, then, its contribution to the overall system speed is added. The velocities of the lower-level tasks are projected onto the null space of the immediate superior task, thus removing the component which can have conflict with this. Therefore, the high priority task is always achieved, and the lower ones are met if they do not conflict with the task of higher priority.

3 Formation Representation

This section presents the direct and inverse transformations that describe a three UAV formation. These relationships allow defining the characteristics of the multi-robot system, such as position, shape and orientation, as a function of the robots' positions or vice versa.

In [11],a representation of a multi-robot system and a control structure called "cluster space control" are presented. This approach considers a ground-based triangular formation of three mobile robots.

3.1 Direct Kinematic Transformation

Figure 1 shows the referential frames that represent an aerial UAV formation, specifically, the inertial and the mobile frames related to the formation triangle. All measurements are referred to the inertial frame $< I >$, corresponding to axes ${}^I x$, ${}^I y$ and ${}^I z$ (such as the formation variables and robot poses). The origin of the formation frame $< F >$ (whose axes are ${}^F x$, ${}^F y$ and ${}^F z$) is chosen to coincide with the centroid of the triangle, and this moving frame allows determining the relative orientation respective the fixed frame.

The formation variables were chosen to represent a three-robot system in $3D$, in such a way that the main characteristics of the formation can be distinguished clearly. To determine the MRS position P_F, the centroid of the triangle was chosen as the representative reference of the system, and can be expressed as

$$P_F = \begin{bmatrix} x_F \\ y_F \\ z_F \end{bmatrix} = \begin{bmatrix} \dfrac{x_1 + x_2 + x_3}{3} \\ \dfrac{y_1 + y_2 + y_3}{3} \\ \dfrac{z_1 + z_2 + z_3}{3} \end{bmatrix}. \tag{7}$$

The shape variables of the formation are defined as S_F, and corresponds to the distance between robots R_1 and R_2; the distance between robots R_1 and R_3, and the angle $\widehat{R_2 R_1 R_3}$, which is equivalent to the distance $R_2 - R_3$, following the cosine theorem.

$$S_F = \begin{bmatrix} d_1 \\ d_2 \\ \beta_F \end{bmatrix} = \begin{bmatrix} \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \\ \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2 + (z_1 - z_3)^2} \\ \arccos \dfrac{d_1^2 + d_2^2 - d_3^2}{2 d_1 d_2} \end{bmatrix}, \tag{8}$$

where $d_3 = \sqrt{(x_2 - x_3)^2 + (y_2 - y_3)^2 + (z_2 - z_3)^2}$ represents the distance $R_2 - R_3$.

Finally, the triangle orientation with respect to the inertial frame needs to be calculated, but by firstly defining the formation frames. The axis ${}^F x$ is considered as extending from the centroid of the triangle out to the robot position R_1. Axis ${}^F z$ is perpendicular to the three UAV plane, and the third one ${}^F y$ is consequently defined in order to complete the formation frame.

$${}^F x = R_1 - P_F = [x_1\ y_1\ z_1]^T - [x_F\ y_F\ z_F]^T \tag{9}$$

$${}^F z = (R_1 - R_2) \times (R_1 - R_3) \tag{10}$$

$${}^F y = {}^F z \times {}^F x. \tag{11}$$

According to [17], the Tait-Bryan angles (pitch, roll and yaw) relating both reference frames are stated as

$$\begin{aligned} \phi_F &= \arctan \frac{{}^F_C\mathcal{R}_{21}}{{}^F_C\mathcal{R}_{11}} = \arctan \frac{{}^F x_{(y)}}{{}^F x_{(x)}} \\ \theta_F &= -\arcsin {}^F_C\mathcal{R}_{31} = -\arcsin {}^F x_{(z)} \\ \psi_F &= \arctan \frac{{}^F_C\mathcal{R}_{32}}{{}^F_C\mathcal{R}_{33}} = \arctan \frac{{}^F y_{(z)}}{{}^F z_{(z)}}. \end{aligned} \tag{12}$$

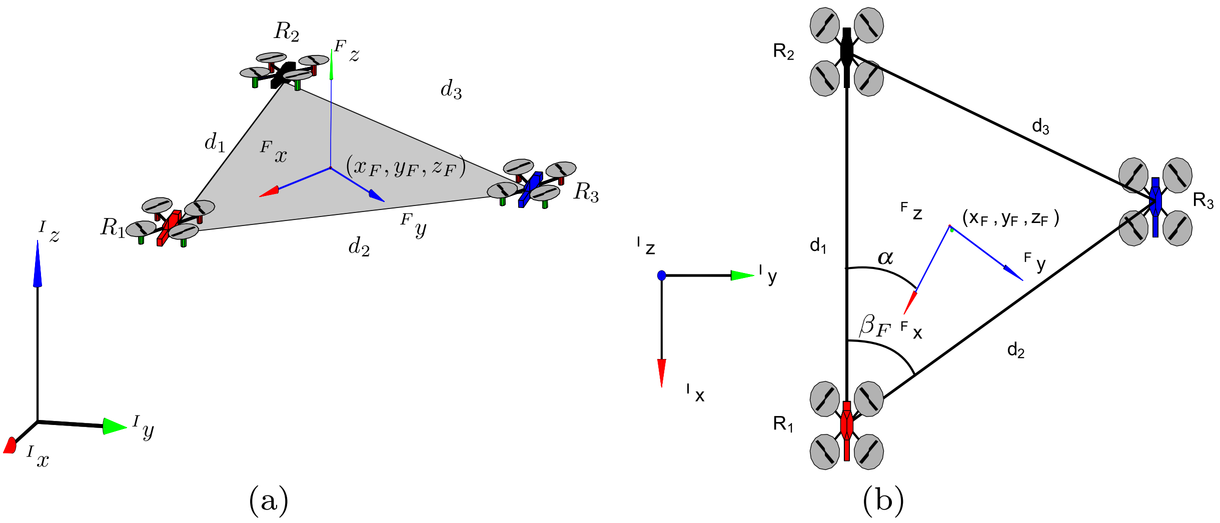


Fig. 1 Formation variables. **a** Normal view of the formation. **b** Top view of the formation

where $\mathcal{R}$ is the rotation matrix that relates both frames. (7), (8) and (12) represent the direct kinematics of the system, which allows characterizing the formation from the positions of each robot.

By differentiating these equations, the direct differential kinematics is obtained with the Jacobian matrix,

$$\dot{q} = J_{(x)}\dot{x} \tag{13}$$

Note that (13) allows relating the time variations (velocities) of the formation variables respecting the robot variables, in a way of resembling a mapping between two spaces.

3.2 Inverse Kinematic Transformation

The previous subsection presented a $3D$ characterization of a 3 UAV formation, starting from the positions of each robots and their differential relation. However, sometimes it is necessary to calculated the inverse relationship, i.e. to determine the positions of the individual robots by starting from the formation variables.

By expressing the positions of robots R_2 and R_3 referred to robot R_1, and the position of robot R_1 with respect to the origin, it results

$$\begin{aligned}
x_1 &= x_F + \frac{2}{3}h_F \cos\theta_F \cos\psi_F \\
y_1 &= y_F + \frac{2}{3}h_F (\sin\phi_F \sin\theta_F \cos\psi_F - \cos\phi_F \sin\psi_F) \\
z_1 &= z_F - \frac{2}{3}h_F (\cos\phi_F \sin\theta_F \cos\psi_F + \sin\phi_F \sin\psi_F) \\
x_2 &= x_1 - d_1 \cos\theta_F \cos(\alpha + \psi_F) \\
y_2 &= y_1 - d_1 \cos\phi_F \sin(\alpha + \psi_F) \\
&\quad - d_2 \sin\phi_F \sin\theta_F \cos(\alpha + \psi_F) \\
z_2 &= z_1 + d_1 \sin\phi_F \sin(\alpha + \psi_F) \\
&\quad + d_2 \cos\phi_F \sin\theta_F \cos(\alpha + \psi_F) \\
x_3 &= x_1 - d_2 \cos\theta_F \cos(\beta_F - \alpha - \psi_F) \\
y_3 &= y_1 + d_2 \cos\phi_F \sin(\beta_F - \alpha - \psi_F) \\
&\quad - d_2 \sin\phi_F \sin\theta_F \cos(\beta_F - \alpha - \psi_F) \\
z_3 &= z_1 - d_2 \sin\phi_F \sin(\beta_F - \alpha - \psi_F) \\
&\quad + d_2 \cos\phi_F \sin\theta_F \cos(\beta_F - \alpha - \psi_F)
\end{aligned} \tag{14}$$

where $h_F = \sqrt{\frac{1}{2}\left(d_1^2 + d_2^2 - \frac{1}{2}d_3^2\right)}$ is the distance between R_1 and the central point of the segment $\overline{R_2R_3}$ passing through the point (x_F, y_F, z_F), and $\alpha = \arccos \dfrac{d_1^2 + d_2^2 - \dfrac{d_3^2}{4}}{2d_1d_2}$.

If $\phi_F = 0$ and $\theta_F = 0$ is complied, which is the case for ground-based robot formation, (14) turns to be equal to the corresponding one of [11]. This means that the equations presented in [11] represents a particular case of (14).

By differentiating the above equations, the direct differential kinematics is obtained

$$\dot{x} = J_{(q)}^{-1}\dot{q}, \tag{15}$$

where J^{-1} is the inverse Jacobian matrix, which allows relating the variations from of the formation variables respecting each robot velocity.

3.3 Constraints in Formation Variables

The above equations represent a transformation $T : \Re^9 \rightarrow \Re^9$ in which (7), (8) and (12) characterize the forward direction, whereas (14) characterizes the inverse direction. However, there are natural constraints in the formation because, necessarily, the altitude of each robot must be $z_i \geq 0 \; \forall t > 0$, where $i = 1, 2, 3$. These constraints can be expressed as

$$\begin{aligned}
z_1 &\geq z_F - \frac{2}{3}h_F (c\phi_F s\theta_F c\psi_F + s\phi_F s\psi_F) \\
z_2 &\geq z_1 + d_1 \sin\phi_F \sin(\alpha + \psi_F) \\
&\quad + d_2 \cos\phi_F \sin\theta_F \cos(\alpha + \psi_F) \\
z_3 &\geq z_1 - d_2 \sin\phi_F \sin(\beta_F - \alpha - \psi_F) \\
&\quad + d_2 \cos\phi_F \sin\theta_F \cos(\beta_F - \alpha - \psi_F)
\end{aligned} \tag{16}$$

The inequality becomes an equality for ground robots. If the MRS is formed with aerial robots, $z_i > 0$ $\forall t > 0$, for $i = 1, 2, 3$. The references' planner should take into account said restrictions to prevent any accident because, if these constraints are not met, the altitude of at least one robot could be negative in its attempt to reach the formation references.

4 Formation Control

The interest of adopting a null-space approach is motivated by its capability of treating separately the

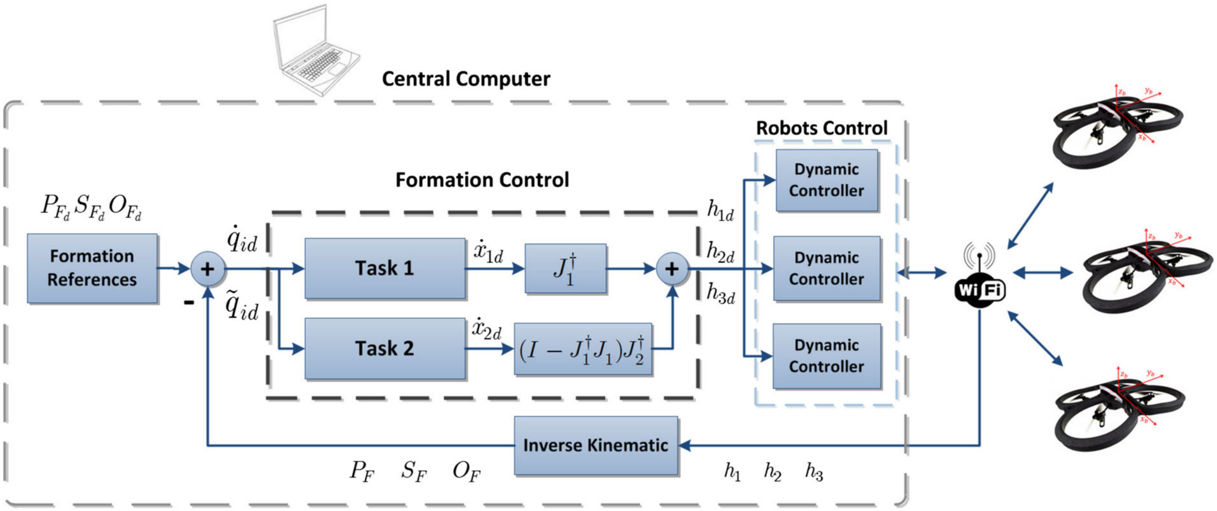


Fig. 2 Structure of the control scheme used in the experiments run

set of variables that define the formation's orientation and shape from those that determine its position in $3D$. Note that these separated calculations give rise to two different control systems. Finally, both task are combined to generate the corresponding control actions. An advantage of such splitting-plus-combination approach is that it allows obtaining different behaviours, depending on the order of the partial tasks to be executed by the controller structure.

4.1 Controlling the Shape and Orientation

The proposed shape and orientation controller is used to define the geometric shape and relative orientation of the multi-robot system.

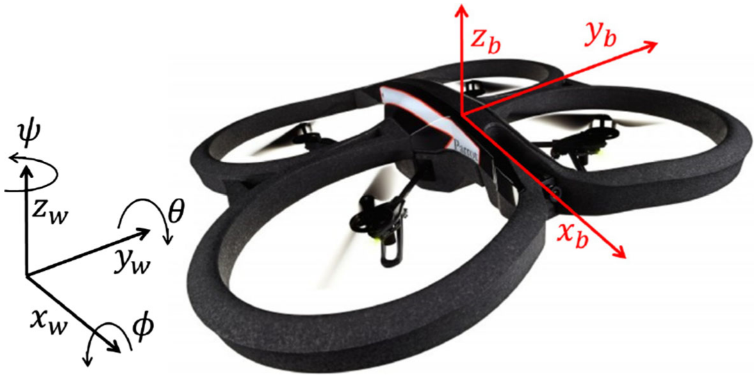
Next, the shape and orientation variables are analysed and the Jacobian matrix in (13) is split, by taking just those rows that relate shape and orientation variables, with the robot formation speed

$$\dot{q}_1 = J_1 \dot{x}, \tag{17}$$

where $q_1 = [d_1\ d_2\ \beta_F\ \phi_F\ \theta_F\ \psi_F]^T$ and

$$J_{(x)} = \begin{bmatrix} \frac{\partial d_1}{\partial x_1} & \frac{\partial d_1}{\partial y_1} & \frac{\partial d_1}{\partial z_1} & \cdots & \frac{\partial d_1}{\partial x_3} & \frac{\partial d_1}{\partial y_3} & \frac{\partial d_1}{\partial z_3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ \frac{\partial \psi_F}{\partial x_1} & \frac{\partial \psi_F}{\partial y_1} & \frac{\partial \psi_F}{\partial z_1} & \cdots & \frac{\partial \psi_F}{\partial x_3} & \frac{\partial \psi_F}{\partial y_3} & \frac{\partial \psi_F}{\partial z_3} \end{bmatrix}. \tag{18}$$

Fig. 3 The AR.Drone 2.0 quadrotor and the coordinate systems adopted ($\{w\}$ and $\{b\}$ are the global and the body coordinate systems, respectively)



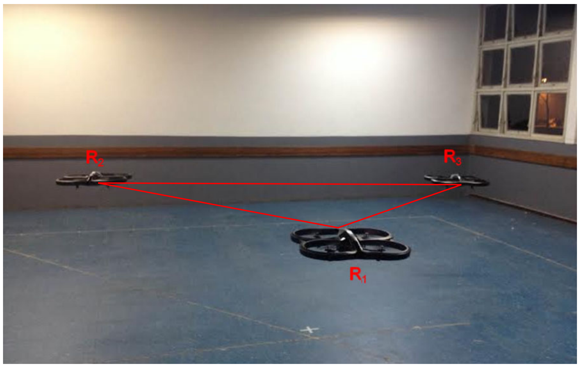


Fig. 4 A snapshot of the 3 Ar.Drones during the experiment

The controller proposed for q_1 variables is

$$\dot{x}_d = J_1^{\dagger}\left(\dot{q}_{1d} + L_1 \tanh\left(k_1 L_1^{-1}\tilde{q}_1\right)\right), \tag{19}$$

where L_1 and k_1 are positive diagonal matrices that allow tuning the system response. The set of variables $\dot{q}_{1d}$ represents the desired time variation of shape and orientation variables, whereas $\tilde{q}_1$ stands for the control error of these variables and $\dot{x}_d$ are the velocity references of each robot. The term tanh is used to ensure the saturation of the velocity calculated by the controller, so as to guarantee that the vehicles can perform the computed control actions.

Table 1 References used in the experiment for the trajectory to be tracked and the formation variables

Formation variable	$0 - 30[sec]$
x_F	$\frac{4t}{15}\,[m]$
y_F	$\sin\left(\frac{t}{15}\right)\,[m]$
z_F	$2\,[m]$
p_F	$3.5\,[m]$
q_F	$3.5\,[m]$
β_F	$\pi/3$ [rad]
ϕ_F	0 [rad]
θ_F	$-\frac{\pi}{18}$ [rad]
ψ_F	0 [rad]

4.2 Controlling the Position of the Centroid

This controller allows heading the formation's centroid to its reference, in positioning or trajectory-tracking tasks.

The 3 rows of the Jacobian matrix of (13) not included in J_1 relates the time variation of the centroid position to the velocity of each robot. This sub-matrix is a new Jacobian matrix, namely J_2, for which just the centroid variables are considered

$$\dot{q}_2 = J_2\dot{x}, \tag{20}$$

Fig. 5 Time evolution for the shape variables

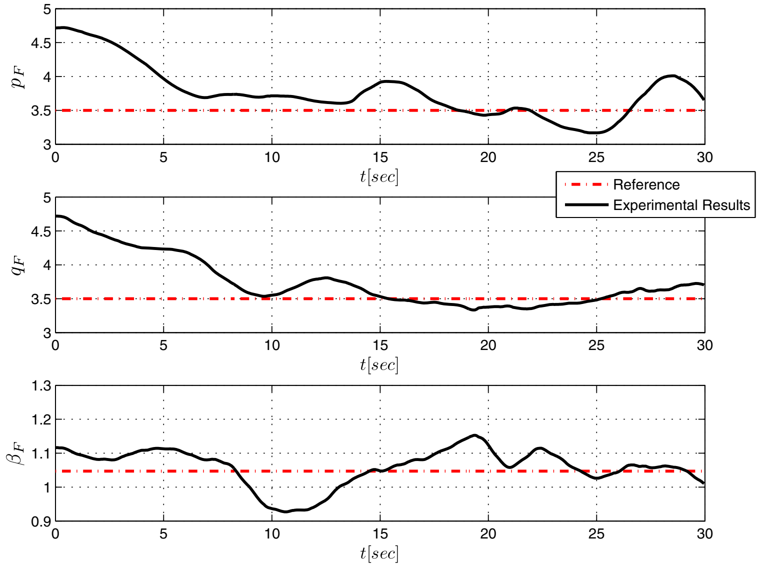
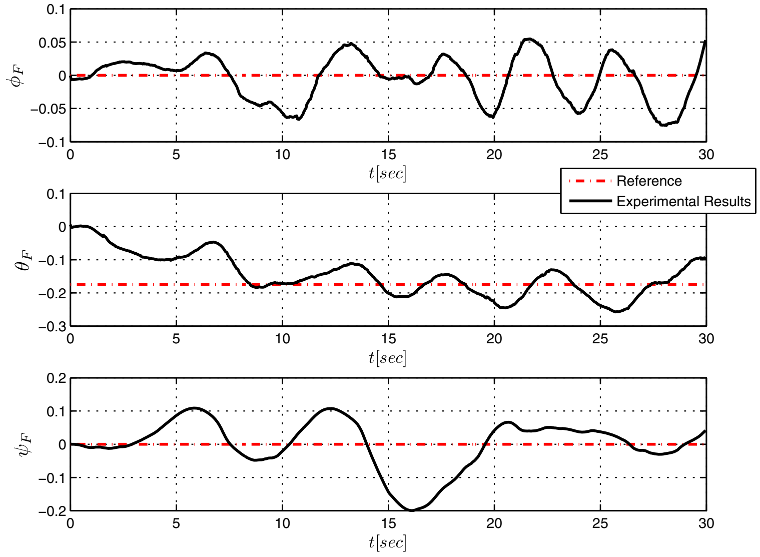


Fig. 6 Time evolution for the orientation variables



where $q_2 = [x_F \ y_F \ z_F]^T$. The proposed controller for the trajectory tracking and positioning task for the centroid is

$$\dot{x}_d = J_2^{\dagger}\left(\dot{q}_{2d} + L_2 \tanh\left(k_2 L_2^{-1} \tilde{q}_2\right)\right), \tag{21}$$

where L_2 and k_2 are positive diagonal matrices that allow tuning the system response. The variable $\dot{q}_{2d}$ represents the desired time variation of shape and orientation variables, whereas $\tilde{q}_2$ represents the control error for these variables and $\dot{x}_d$ are the velocity references of each robot in the formation. The term tanh is used once more to guarantee the saturation of velocities calculated by the controller.

4.3 Formation Controller

Once both formation tasks are solved independently, it is necessary to combine them. As proposed here, the null-space approach will be used in order to prevent any conflict between the solutions from both controllers. Sometimes it is more important to control the shape and orientation of the formation instead

Fig. 7 Time evolution for the position variables

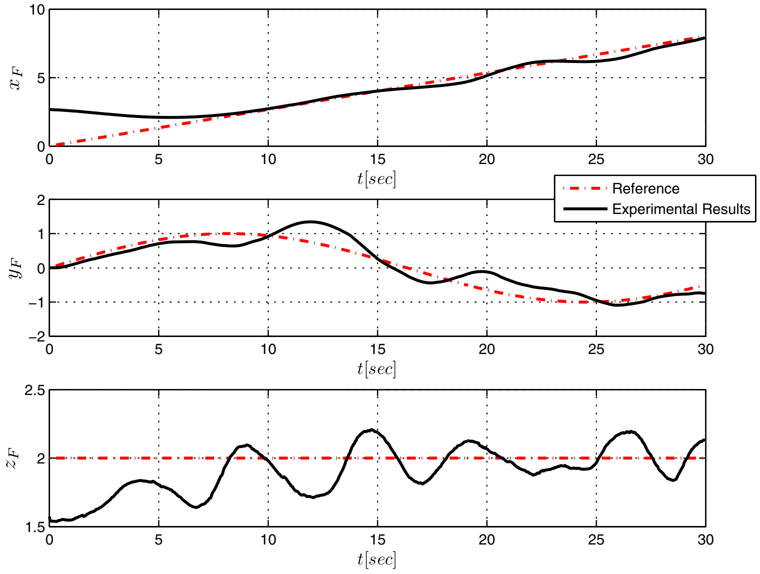
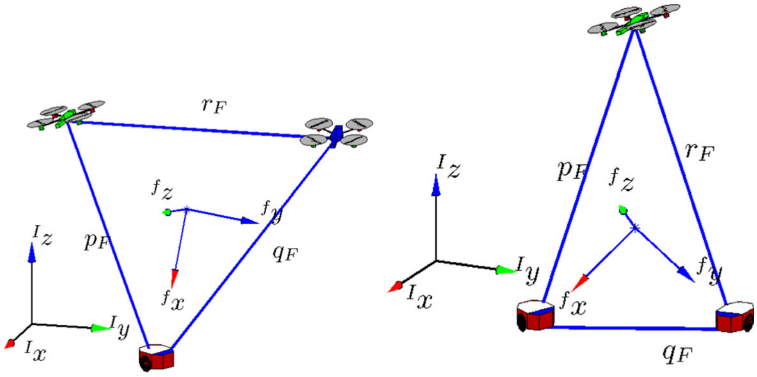


Fig. 8 The two cases of UAV-UVG formations dealt with in this work



(a)

(b)

of its position. Or vice versa. In this paper, the control of shape and orientation is considered the most important task, while position control is the secondary task.

To consider the entire control of the formation, the solution of the higher priority task, i.e. the shape and orientation control, will be projected on the row space of the Jacobian matrix of (13), whereas the centroid position control task will be projected into the null space of such matrix. Notice that both tasks can be fulfilled simultaneously, but the null-space based control does not guarantee that the lower priority task (the secondary one in this case) is accomplished with the optimal velocity [13].

Table 2 References adopted for controlling the 2UAV-1UGV formation

Formation variable	$0 - 40[sec]$
x_F	$4\sin(0.5t)$ [m]
y_F	$4\cos(0.25t)$ [m]
z_F	$\frac{2h_F}{3}(\cos\phi_F \sin\theta_F \cos\psi_F + \sin\phi_F \sin\psi_F)$ $\cong 1.3$ [m]
p_F	4 [m]
q_F	4 [m]
β_F	$\frac{\pi}{2}$ [rad]
ϕ_F	$\pi/32$ [rad]
θ_F	$-\frac{\pi}{3}$ [rad]
ψ_F	$\frac{\pi}{16}$ [rad]

Thus, the complete controller proposed to accomplish both task simultaneously will be

$$\dot{x}_d = J_1^{\dagger}\left(\dot{q}_{1d} + L_1 \tanh\left(k_1 L_1^{-1}\tilde{q}_1\right)\right) + \left(I - J_1^{\dagger} J_1\right) J_2^{\dagger}\left(\dot{q}_{2d} + L_2 \tanh\left(k_2 L_2^{-1}\tilde{q}_2\right)\right), \qquad (22)$$

which represents the two controllers dealt with in the two previous subsections.

4.4 Stability Analysis

In this section the stability of the whole system, when using the proposed controller, is discussed. First of all, lets recall that for any matrix $A \in R^{m\times n}$, the null space and row space are orthogonal subspaces of R^m. Analogously, the left null space and the column space are also orthogonal subspaces of R^n. The following analysis also assumes perfect velocity tracking ($\dot{x} = \dot{x}_d$) for each individual robot in the formation.

The first step is an analysis of the primary objective, i.e. shape and orientation control. By multiplying both members of (22) by J_1, which is supposed to be full rank, and by noting that $J_1\left(I - J_1^{\dagger}J_1\right) = 0$, i.e. the column space and the null space are orthogonal, it results

$$\dot{q}_1 = \dot{q}_{1d} + L_1 \tanh\left(k_1 L_1^{-1}\tilde{q}_1\right), \qquad (23)$$

where $\tilde{q}_1 = q_{1d} - q_1$. Consider the Lyapunov candidate function

$$V_{(q_1)} = \frac{1}{2}\tilde{q}_1^T\tilde{q}_1, \qquad (24)$$

whose time derivative, after some operations will be

$$\dot{V}_{(q_1)} = \tilde{q}_1^T \dot{\tilde{q}}_1 = -\tilde{q}_1^T L_1 \tanh\left(k_1 L_1^{-1} \tilde{q}_1\right). \tag{25}$$

Since L_1 and k_1 are positive definite matrices, $\dot{V}$ will be negative definite, therefore resulting that $\tilde{q}_1 \rightarrow 0$ with $t \rightarrow \infty$.

The lower level task can be accomplished only if it does not pose any conflict with the higher level task. In order to analyse the secondary task, i.e. the centroid position control, lets multiply by J_2 both members of (22), resulting in

$$\begin{aligned} \dot{x}_d &= J_2 J_1^{\dagger}\left(\dot{q}_{1d} + L_1 \tanh\left(k_1 L_1^{-1} \tilde{q}_1\right)\right) \\ &\quad + J_2\left(I - J_1^{\dagger} J_1\right) J_2^{\dagger}\left(\dot{q}_{2d} + L_2 \tanh\left(k_2 L_2^{-1} \tilde{q}_2\right)\right). \end{aligned} \tag{26}$$

Because both task are not conflictive one another, the range of J_1 is orthogonal to the range of J_2 and $J_2 J_1^{\dagger} = 0$, or

$$R\left(J_2^{\dagger}\right) \perp R\left(J_1^{\dagger}\right). \tag{27}$$

By substituting this result in (26),

$$\dot{q}_2 = \dot{q}_{2d} + L_2 \tanh\left(k_2 L_2^{-1} \tilde{q}_2\right). \tag{28}$$

Consider the Lyapunov candidate function

$$V_{(q_2)} = \frac{1}{2} \tilde{q}_2^T \tilde{q}_2, \tag{29}$$

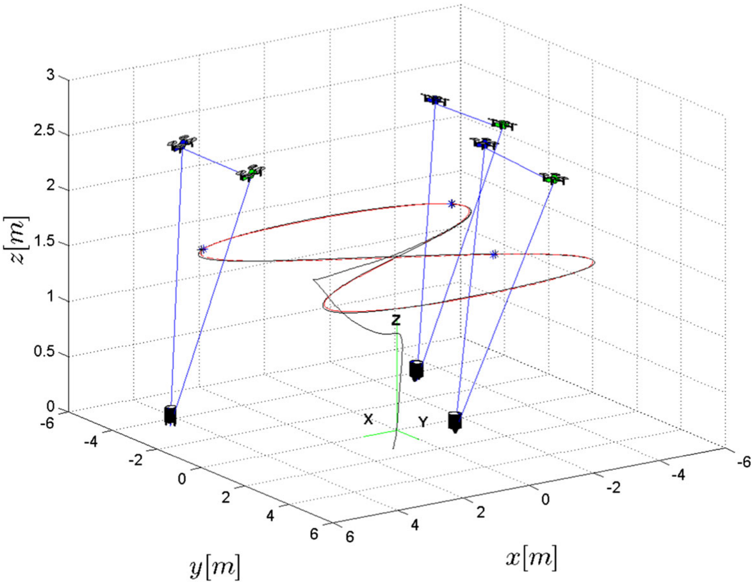
whose time derivative, after some operations, allows concluding that $\tilde{q}_2 \rightarrow 0$ with $t \rightarrow \infty$.

5 Experimental Results

The control strategy here proposed was implemented in laboratory experiences in order to validate it. The structure of the control system it is shown in Fig. 2, where the two control levels, i.e. the formation control level and the individual robots control level, are clearly seen. With the first one, the references for position and velocity of each robot are generated, depending on the error in formation variables. In the second control level, each robot in the formation has a dynamic controller associated to it, in order to enable them to reach the references generated by the first control level. In the experiments, the dynamic controller based on linear algebra, proposed in [14], was adopted as the controller associated to each robot.

The experimental platform chosen in this work is the AR.Drone quadrotor, from Parrot Inc., version 2.0, shown in Fig. 3, altogether with the adopted coordinate systems. Such a platform is an autonomous aerial vehicle (a rotor-craft) commercialized as a hi-tech toy, originally designed to be controlled through smartphones or tablets via Wi-Fi network, with specific communication protocols [15]. Therefore, in our experiments we used a Wi-Fi network with three

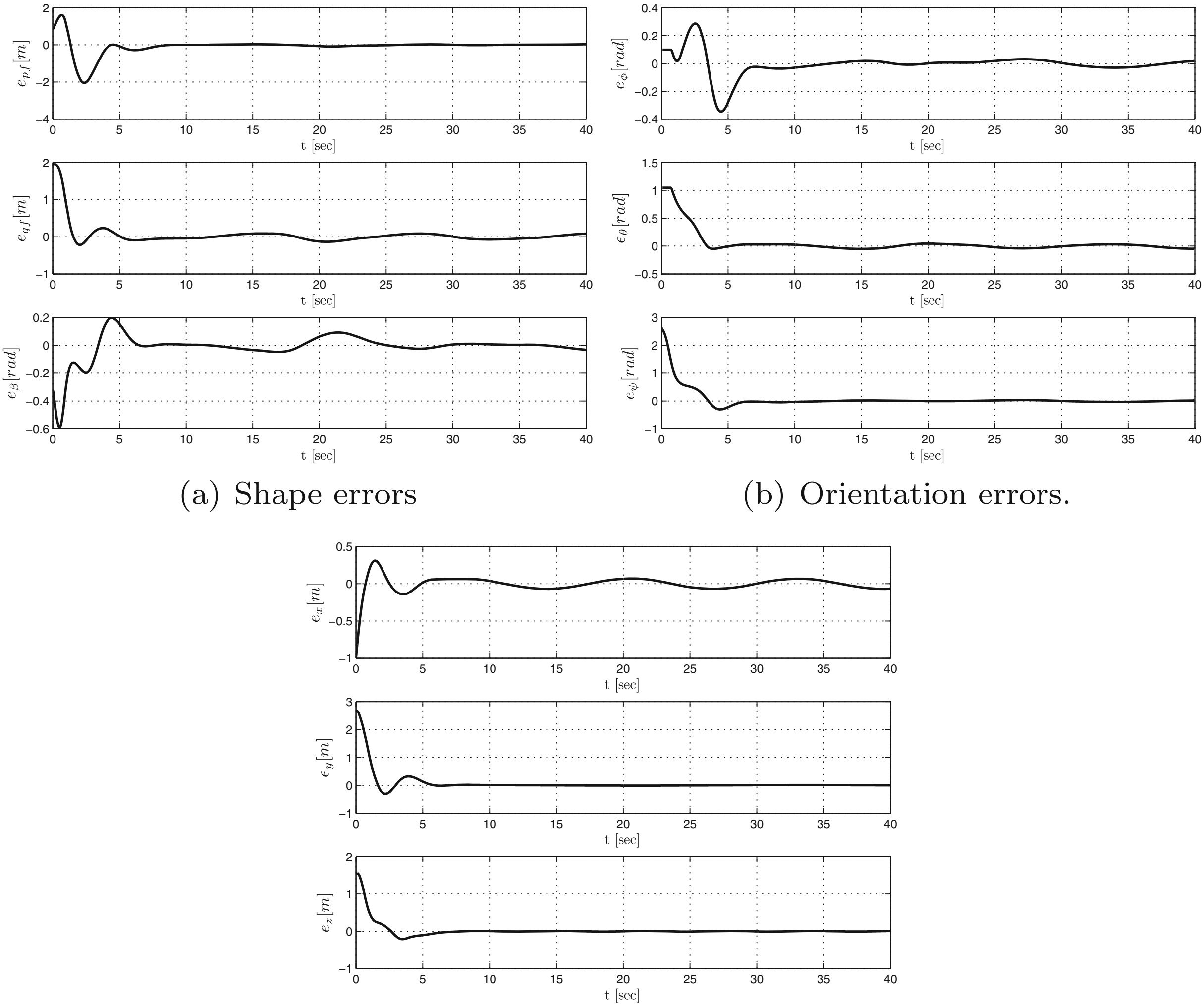
Fig. 9 Evolution of the formation composed by 1 *UGV* and 2 *UAV*s in the 3*D* space



AR.Drones and a central computer. Once the three robots send their position to the central computer, it calculates the inverse kinematics (the formation variables based on the positions of the robots), compares it with the formation references and, finally, the control error is obtained. Next, the controllers associated to each robot generate the control actions to be Wi-Fi transmitted to each corresponding robot in the formation.

Figure 4 shows the aerial multi-robot system operating during the experiment. The experience was a trajectory-tracking one. Table 1 describe the trajectory. Since the laboratory where the experiment was run is a small place, the flight times were short, limited to 30 s, for safety reason.

Figure 5 shows the time evolution of both the desired and the real shape variables, whereas Fig. 6 shows the time evolution of the orientation variables. Figure 7 shows the time evolution of the position variables. It can be noted in all graphics that all real values tend to the reference, with small error, which is due to the differences between model parameters used in the dynamic controller (see [14]) and those ones of the real model. Nevertheless, is always verified that the control system is stable, thus guarantee that all errors are bounded.



(a) Shape errors

(b) Orientation errors.

(c) Barycentre position errors.

Fig. 10 Control errors for a 2UAV-1UGV formation

Table 3 References adopted for controlling the 1UAV-2UGV formation

Formation variable	$0-40[sec]$
x_F	$4\sin(0.5t)$ [m]
y_F	$4\cos(0.5t)$ [m]
z_F	$-\frac{h_F}{3}\sin\theta_F\cos\psi_F \cong 0.8$ [m]
p_F	4 [m]
q_F	4 [m]
β_F	$\frac{\pi}{2}$ [rad]
ϕ_F	0 [rad]
θ_F	$-\frac{\pi}{3}$ [rad]
ψ_F	$\frac{\pi}{16}$ [rad]

6 UAV-UGV Formation Control

In the present section the control strategy presented in Section 4 is expanded to consider the case in which one or more robot(s) of the formation is/are UGV (Unmanned Ground Vehicle) robot(s). Section 3.3 presents the constraints for the formation references so as to ensure that $z_i \geq 0$ along time. If one of the vehicles is a UGV robot, heavier restrictions should be imposed, because its altitude must always be $z_i = 0$. For every UGV vehicle in the formation, one column in the Jacobian matrix (2) is lost, thus imposing an additional restriction to the formation references.

Taking into account the above considerations, there are two possibilities to include UGVs in the three-robots system and thus build a heterogeneous formation. Specifically the first formation is arranged with 2 UAVs and 1 UGV, and the second formation with 1 UAV and 2 UGVs (at least one UAV is kept). Figure 8 shows the two UAV-UGV formations considered below in this work.

In the first one, assuming that R_1 is a ground robot, it implies that $z_1 = 0$. By substituting into (14) , the restrictions

$$z_F = \frac{2h_F}{3}(\cos\phi_F\sin\theta_F\cos\psi_F + \sin\phi_F\sin\psi_F)$$

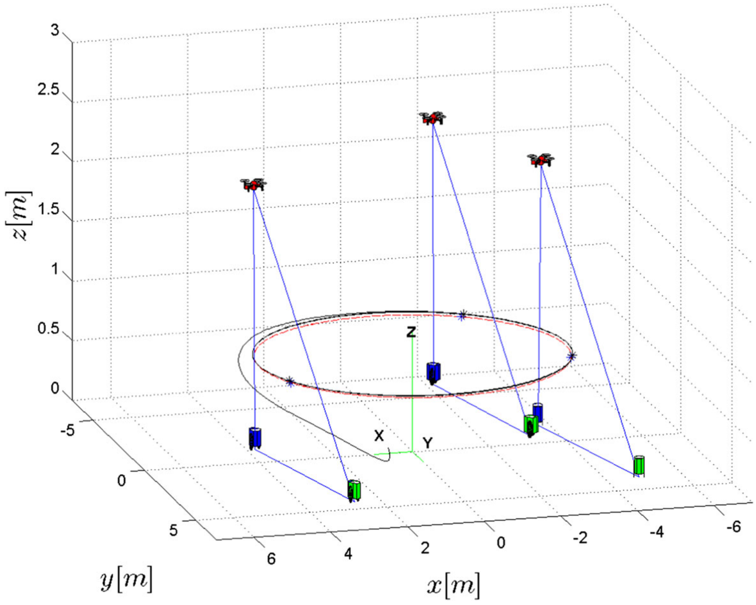
$$0 < \theta_F < \pi$$

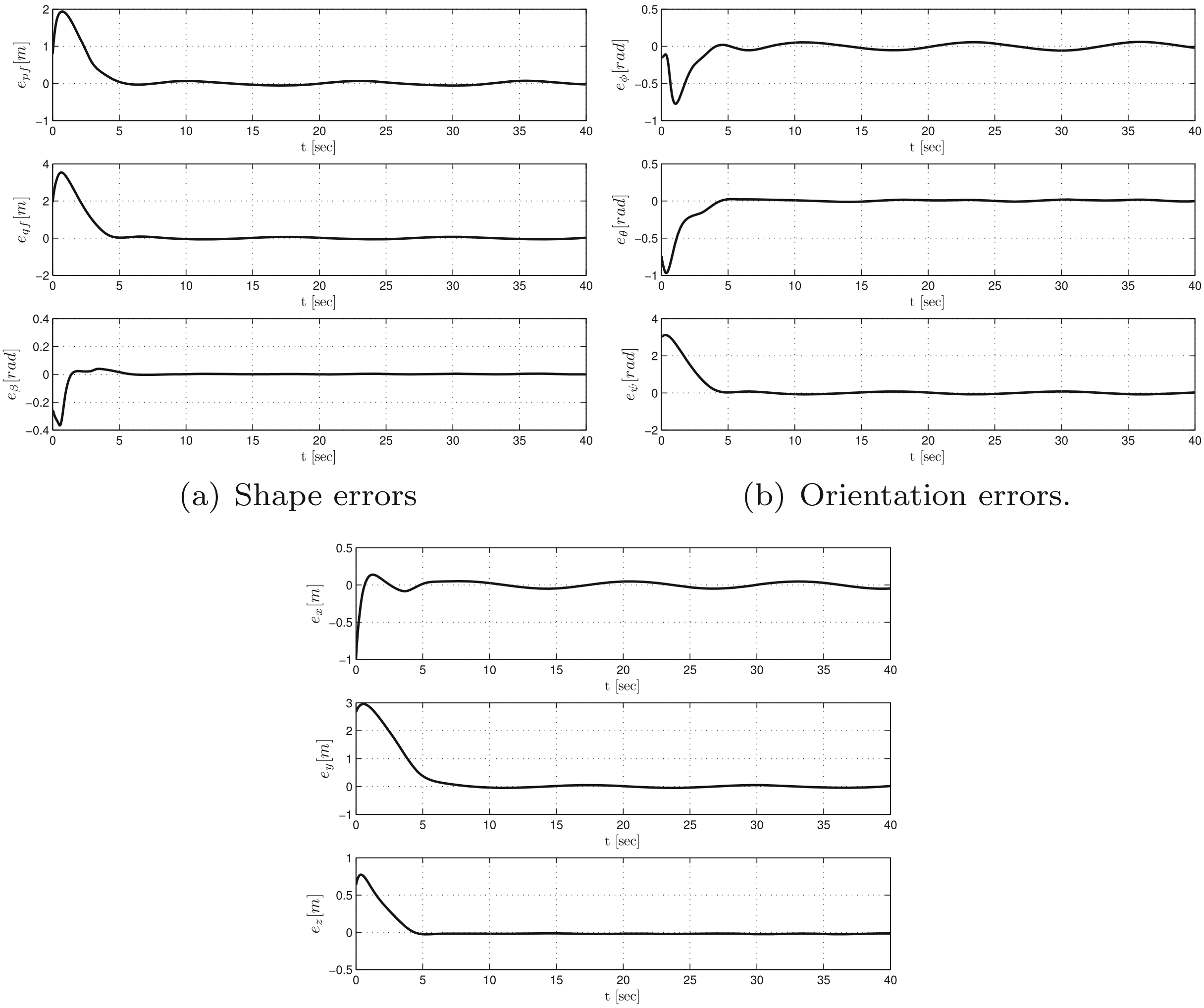
over the references for the formation variables are obtained.

Taking into account these restrictions, the control strategies presented in Section 4 can be used. Indeed, we have run simulations considering two UAVs and one UGV. The references used in the simulation of trajectory-tracking task are detailed in Table 2.

Figure 9 shows the 3D evolution of the UAV-UGV formation, whereas Fig. 10a, b and c present the control errors for shape, orientation and centroid position. In all cases, it is demonstrated that the control errors and bounded and the control system is stable.

Fig. 11 Evolution of the formation composed by 2 *UGV* and 1 *UAV*s in the 3D space





(a) Shape errors

(b) Orientation errors.

(c) Barycentre position errors .

Fig. 12 Control errors by a 1UAV-2UGV formation control

Finally, when R_2 and R_3 are UGVs, it should be guaranteed that $z_2 = 0$ and $z_3 = 0$. By substituting into (14), the restrictions

$$z_F = \frac{h_F}{3} \sin\theta_F \cos\psi_F$$

$$\phi_F = 0$$

are obtained.

For this last case, a trajectory-tracking task is also simulated, in order to check the effectiveness of the control system. References adopted for the formation variables are detailed in Table 3. Figure 11 shows the time evolution of the position of multi-robot system in $3D$, whereas Fig. 12a, b and c show the errors corresponding to formation shape, orientation and centroid position, respectively. From such figures it can be verified that the formation variables effectively converge to their reference values. This means that the control errors are bounded and the control system is effectively stable.

7 Conclusions

A novel controller capable of controlling an aerial multi-robot system, based on multiple objective control and null-space, is proposed in this work. As part of this proposal, a representation of a triangular formation available in the literature was extended to $3D$, the setting from which this work was conceived. The whole task was split in two sub-tasks, the control of the formation shape and orientation, and the control of the position of the barycentre of the formation, which defines the position of the formation centroid in $3D$. Then, the two controllers designed for the two sub-tasks were combined, by projecting one of them into the null space of the other, in order to prevent any conflict. Experiences with the proposed controller gave acceptable results, which validate the proposed approach, as they confirm as well the system stability, in concordance with its theoretical analysis. As an additional contribution, the proposed controller was extended to take into account heterogeneous formations, specifically for the cases in which one or two robots in the triangular formation are ground robots.

References

1. Agnew, M.S., Canto, P.D., Kitts, C.A., Li, S.: Cluster space control of aerial robots. In: International Conference on Advanced Intelligent Mechatronics (IEEE/ASME), pp. 1305–1310, Montral, Canada (2010)
2. Antonelli, G., Arrichiello, F., Chiaverini, S.: The null-space-based behavioral control for mobile robots. In: IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA), pp. 15–20 (2005)
3. Antonelli, G., Arrichiello, F., Chiaverini, S.: The null-space-based behavioral control for autonomous robotic systems. Intell. Serv. Robot. **1**(1), 27–39 (2008)
4. Antonelli, G., Arrichiello, F., Chiaverini, S.: Experiments of formation control with multirobot systems using the null-space-based behavioral control. IEEE Trans. Control Syst. Technol. **17**(5), 1173–1182 (2009)
5. Antonelli, G., Chiaverini, S.: Kinematic control of platoons of autonomous vehicles. IEEE Trans. Robot. **22**(6), 1285–1292 (2006)
6. Balch, T., Arkin, R.: Behavior-based formation control for multirobot teams. IEEE Trans. Robot. Autom. **14**(6), 926–939 (1998)
7. Brandao, A.S., Rampinelli, V.T.L., Martins, F.N., Sarcinelli-Filho, M., Carelli, R.: The multilayer control scheme: A strategy to guide n-robots formations with obstacle avoidance. J. Control Autom. Electr. Syst. **26**(3), 201–214 (2015)
8. Carelli, R.O., Roberti, F., Tosetti, S.: Direct visual tracking control of remote cellular robots. Robot. Auton. Syst. **54**(10), 805–814 (2006)
9. Chen, J., Sun, D., Yang, J., Chen, H.: Leader-follower formation control of multiple non-holonomic mobile robots incorporating a receding-horizon scheme. Int. J. Robot. Res. **29**(6), 727–747 (2010)
10. Consolini, L., Morbidi, F., Prattichizzo, D., Tosques, M.: Leader-follower formation control of nonholonomic mobile robots with input constraints. Automatica **40**(5), 1343–1349 (2008)
11. Kitts, C.A., Mas, I.: Cluster space specification and control of mobile multirobot systems. IEEE/ASME Trans. Mechatron. **14**(2), 207–218 (2009)
12. Liu, J., Wu, J.: Multiagent Robotic Systems. CRC Press (2001)
13. Park, J., Choi, Y., Chung, W.K., Youm, Y.: Multiple tasks kinematics using weighted pseudo-inverse for kinematically redundant manipulators. In: Proceedings of the 2001 IEEE International Conference on Robotics and Automation (ICRA'01), pp. 4041–4047, Seoul, Korea (2001)
14. Rosales, C., Gandolfo, D., Scaglia, G., Jordan, M., Carelli, R.: Trajectory tracking of a mini four-rotor helicopter in dynamic environments - a linear algebra approach. Robotica **FirstView**, 1–25 (2014). doi:10.1017/S0263574714000952
15. Santana, L.V., Brandao, A.S., Sarcinelli-Filho, M., Carelli, R.: A trajectory tracking and 3d positioning controller for the ar.drone quadrotor. In: 2014 International Conference on Unmanned Aircraft Systems (ICUAS'14), pp. 756–767 (2014)
16. Scharf, D., Hadaegh, F., Ploen, S.: A survey of spacecraft formation flying guidance and control (part ii): Control. In: American Control Conference, pp. 2976–2985, Massachusetts, USA (2004)
17. Slabaugh, G.G.: Computing euler angles from a rotation matrix. Technical Reports (1999)
18. Tanner, H., Pappas, G., Kumar, V.: Leader-to-formation stability. IEEE Trans. Robot. Autom. **20**(3), 443–455 (2004)
19. Zhang, Y., Mehrjerdi, H.: A survey on multiple unmanned vehicles formation control and coordination: normal and fault situations. In: International Conference on Unmanned Aircraft Systems (ICUAS), pp. 1087–1096, GA, USA (2013)

Claudio Rosales studied Electronic Engineering at UNSJ (Argentine) obtaining the degree Electronic Engineering in 2009, and the PhD in Control Systems from the National University of San Juan, in 2014. He is a postdoctoral researcher of the Council for Scientific and Technological Research, Argentina, since 2015. His main interests included algorithms for multirobots systems, nonlinear control and aerial robotic.

Paulo Leica was born in Quito, Ecuador in 1978. He graduated in Electronics & Control Engineering and obtained the master degree in Control Systems from the National Polytechnic School, Ecuador. He received the Ph.D. degree in Electrical Engineering from the National University of San Juan, Argentina. He is currently full Professor at the National Polytechnic School. He is the coordinator of the Program in Electronics & Control Engineering at the same university. His research interests are robotics, Control Systems and artificial intelligence applied to automatic control.

Mario Sarcinelli-Filho received the B.S. degree in Electrical Engineering from Federal University of Espírito Santo, Brazil, in 1979, and the M. Sc. and Ph. D. degrees, both in Electrical Engineering, from Federal University of Rio de Janeiro, Brazil, in 1983 and 1990, respectively. He is currently a full professor at the Department of Electrical Engineering, Federal University of Espírito Santo, Brazil, and a researcher of the Brazilian National Council for Scientific and Technological Development (CNPq). His research interests are control of unmanned aerial vehicles, including formations of unmanned aerial vehicles, coordinated control of mobile robots, mobile robot navigation, signal and image processing, and sensing systems. He has co-authored more than 40 journal papers, more than 300 conference papers, and 14 book chapters, and advised 17 PhD students and 19 MSc students.

Gustavo Scaglia received the Dipl.-Ing. degree in Electronic Engineering with orientation in Control Systems from the National University of San Juan, Argentina, in 1999, and the Ph.D in Control Systems from the Institute of Automatic Control at the National University of San Juan, Argentina in 2006, his main works are related to new algorithms for trajectory tracking. He is a Research Fellow of the Council for Scientific and Technological Research, Argentina, since 2011. He leads different technological projects and his current scientific research is at the Engineering Chemical Institute from National University of San Juan. His main interests are algorithms for tracking trajectories, nonlinear and adaptive control theory, and mechanical and chemical process.

Ricardo Carelli (M'76 - SM'98) was born in San Juan, Argentina. He graduated in Engineering from the National University of San Juan, Argentina, and obtained a Ph.D degree in Electrical Engineering from the National University of Mexico (UNAM). He is full professor at the National University of San Juan and senior researcher of the National Council for Scientific and Technical Research (CONICET, Argentina). Prof. Carelli is Director of the Instituto de Automática, National University of San Juan (Argentina). His research interests are on Robotics, Manufacturing Systems, Adaptive Control and Artificial Intelligence Applied to Automatic Control. Prof. Carelli is a senior member of IEEE and a member of AADECA-IFAC.