Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Cylindrical Bounded Quaternion Control for Tracking and Surrounding a Ground Target Using UAVs ⋆

Hernan Abaunza * **Pedro Castillo** * **Didier Theilliol** **
Adel Belkadi **,*** **Laurent Ciarletta** ***

* *Sorbonne Universites, Universite de Technologie de Compiègne, CNRS UMR 7253, Compiègne, France (e-mail: (castillo,habaunza)@hds.utc.fr).*
** *Universite de Lorraine, CRAN, CNRS, UMR 7039, BP 70239, 54506 Vandoeuvre Cedex, France (e-mail: adel.belkadi@univ-reims.fr) (e-mail: didier.theilliol@univ-lorraine.fr).*
*** *Mines Nancy, Universite de Lorraine, Loria, UMR 7503, France (e-mail: laurent.ciarletta@loria.fr).*

Abstract: A cooperative tracking algorithm for multiple quadrotors autonomously tracking and surrounding a target ground vehicle is presented in this paper. A nonlinear bounded controller is proposed using geometrical functions to stabilize the translational and rotational dynamics using bounded 3-dimensional control inputs, employing quaternion properties and operators in its design, Lyapunov theory is used to prove system stability. The navigation challenge is tackled by proposing a cost function based on the desired behavior of the aerial vehicles for tracking and surrounding the target, which is solved by finding the optimal solution with a Particle Swarm Optimization algorithm. Simulations and experimental results corroborate the good performance of the scheme.

Keywords: Flying robots; Guidance navigation and control; Autonomous robotic systems; Bounded Control; Quadrotors; Quaternions; Multi-robot systems

1. INTRODUCTION

Heterogeneous multi-robot systems (those consisting on more than one kind of robot) have gained an increasing interest in the last years. Many applications have been conceived on which autonomous or semi-autonomous robots collaborate to accomplish a (common) mission. A particular case is the cooperation between Unmanned Aerial and Ground Vehicles (UAVs and UGVs).

Some advances have been presented in the literature, for instance, Cognetti et al. [2014] proposed a cooperative navigation scheme, on which a quadrotor follows multiple UGVs using a camera, Cantelli et al. [2013] introduced an heterogeneous system based on a UAV autonomously following a land robot for application on Humanitarian de-mining operations. Brandao et al. [2012], developed a leader-follower navigation algorithm where a miniature helicopter follows a ground robot, while Tanner [2007] proposed a coordinated controller for groups of UAVs and UGVs using a decentralized flocking algorithm.

A recurrent problem found in multi-robot systems is computing trajectories for each agent that achieve the flight objectives while ensuring collision avoidance, see Pang et al. [2011]. The solutions for this problem can be classified in three categories: centralized (for example Kempker et al. [2012], and Meskin and Khorasani [2011]), decentralized (see Tanner et al. [2005]) and distributed (refer to Belkadi et al. [2016]). The choice of a strategy depends mainly on the available resources, the working environment and the mission objective, see Worthmann et al. [2015].

In this work, a distributed trajectory generation algorithm is proposed to coordinate a group of UAVs that track a moving ground vehicle. The objective is to arrive at a desired formation around the target, and then surround it with a continuous coordinated motion while avoiding collisions. The formulation of this algorithm is based on a Particle Swarm Optimization (PSO) problem, see Kennedy [2010] and Rendón and Martins [2017].

In order to robustly follow the mission trajectories, a bounded quaterion-based controller is also introduced. The algorithm is based on a cylindrical formulation that ensures a consistent bounded control input for any direction on the x-y plane, while maintaining an independent bound on the z axis.

Bounded controllers have been studied in the literature for their robustness towards disturbances and model uncertainties, see Bateman and Lin [2003] and Ran et al. [2016]. Some works have explored the application of saturated controllers on UAVs. Castillo et al. [2004] and Castillo et al.

⋆ Research Supported by the Ministére de L'Education Nationale, de l'Enseignement Supérieur et de La Recherche, the National Network of Robotics Platforms (ROBOTEX), the Conseil Regional de Lorraine, from France, and the CONACYT from Mexico.

[2005] proposed a position controller for a quadrotor based on nested saturations. In Castillo et al. [2006], bounded functions are combined with a backstepping technique to design an attitude controller for a quadrotor.

Quaternions have also been studied for developing UAV controllers, due to their singularity-free and continuous representations of rotations. Some of these works can be found in Abaunza and Castillo [2019], Diao et al. [2013], Liu et al. [2015] and Izaguirre-Espinosa et al. [2016].

The outline of the paper is organized as follows: In Section 2 the challenge of this work is stated. The nonlinear cylindrical bounded quaternion-based controller is developed in Section 3. Section 4 introduces the coordinated circular target tracking strategy for the quadrotors. Real-time simulations are detailed in Section 5, while experimental validations are presented in Section 6.

2. OBJECTIVE

Our goal is to give a solution for a scenario where a mobile target is tracked by multiple UAVs that follow a desired configuration. Once the target is reached, it is surrounded by performing a desired coordinated path while avoiding collisions, see Figure 1.

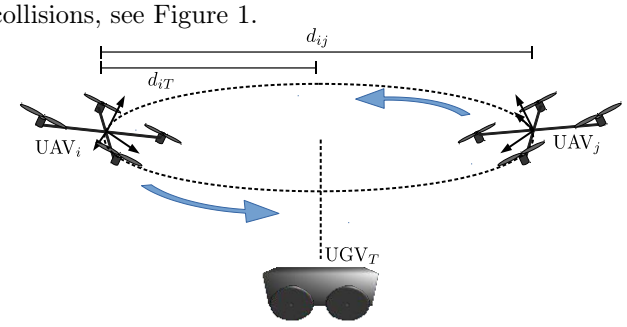

Figure 1. Proposed scenario; two quadrotors autonomously tracking a UGV, while facing it.

In this case study, the target will be a ground vehicle performing random movements and several aerial vehicles fly to surround it in a symmetrical formation (imposed by the user) while avoiding collisions.

The quadrotor dynamic model based on Newton's equations and using quaternion rotations is given by

$$\frac{d}{dt}\begin{bmatrix}\boldsymbol{p}\\ \dot{\boldsymbol{p}}\\ \boldsymbol{q}\\ \boldsymbol{\Omega}\end{bmatrix} = \begin{bmatrix}\dot{\boldsymbol{p}}\\ \boldsymbol{q}\otimes\frac{1}{m}\boldsymbol{F}_{th}\otimes\boldsymbol{q}^* + \boldsymbol{g}\\ \frac{1}{2}\boldsymbol{q}\otimes\boldsymbol{\Omega}\\ J^{-1}(\boldsymbol{\tau}-\boldsymbol{\Omega}\times J\boldsymbol{\Omega})\end{bmatrix}, \tag{1}$$

where $p = [x\ y\ z]^T$ denotes the vehicle position with respect to the inertial frame $\mathcal{I}$, and $\boldsymbol{q}$ is the unit quaternion which defines the body rotation. $\boldsymbol{F}_{th}$, $\boldsymbol{\tau} \in \mathbb{R}^3$ represent the vertical thrust force and torque vectors acting on the vehicle with respect to the body frame. J is the inertia matrix and $\boldsymbol{\Omega} \in \mathbb{R}^3$ represents the angular speed vector of the aerial vehicle.

The expression $\boldsymbol{q}\otimes\frac{\boldsymbol{F}_{th}}{m}\otimes\boldsymbol{q}^* = \boldsymbol{F}_{th}^{\mathcal{I}}$ means that $\boldsymbol{F}_{th}$ is rotated to the inertial frame $\mathcal{I}$ using quaternion operations.

3. CYLINDRICAL BOUNDED QUATERNION CONTROL

A controller for robustly tracking trajectories is developed in this section by introducing a bounding function for the magnitude of any given vector inside a cylinder. The idea is that its output limits the final action of the control forces and torques.

Define a *cylindrical bounded function* $\boldsymbol{\sigma}_d(\boldsymbol{a}) : \mathbb{R}^3 \to \mathbb{R}^3$, with arguments $\boldsymbol{a} = [a_x, a_y, a_z]^T$ and bounding limits $b = [b_{xy}, b_z]^T$, as

$$\boldsymbol{\sigma}_d(\boldsymbol{a}) \coloneqq \begin{bmatrix}\sigma_{xy}\\ \sigma_z\end{bmatrix},\ \sigma_{xy}\ \in \mathbb{R}^2, \sigma_z \in \mathbb{R}\ ; \tag{2}$$

$$\sigma_{xy} \coloneqq \begin{cases}[a_x, a_y]^T & \text{for } ||[a_x, a_y]|| < b_{xy}\,,\\ b_{xy}\dfrac{[a_x, a_y]^T}{||[a_x, a_y]^T||} & \text{for } ||[a_x, a_y]|| \geq b_{xy}\,,\end{cases} \tag{3}$$

$$\sigma_z \coloneqq \begin{cases}a_z & \text{for } |a_z| < b_z\,,\\ b_z\,\text{sign}(a_z) & \text{for } |a_z| \geq b_z\,.\end{cases} \tag{4}$$

Note that vector $\boldsymbol{\sigma}_d(\boldsymbol{a})$ is contained inside a cylinder centered in the origin with radius b_{xy} and height $2b_z$, this ensures a symmetrical behavior in the xy plane, and independent bounds in the z axis, as Figure 2 illustrates.

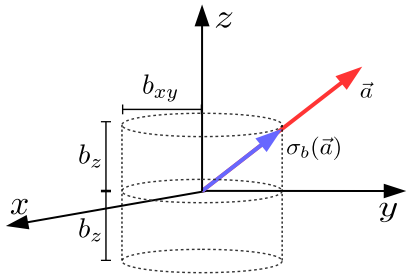

Figure 2. Bounding a vector inside a cylinder.

The control strategy will be based on a function which bounds a vector inside of a cylinder such that a symmetrical behavior is ensured in any direction of the $x-y$ plane while setting a different bound for the vertical axis.

3.1 Translational Bounded Control Law

***Lemma** 1. The position subsystem of* (1) *converges asymptotically to a desired reference* $\boldsymbol{p}_d$ *by means of a control force given by*

$$\begin{aligned}\boldsymbol{F}_u = &-mK_t\boldsymbol{\sigma}_a\Big(K_a\Big(\dot{\boldsymbol{p}}-\dot{\boldsymbol{p}}_d\Big) + K_b\boldsymbol{\sigma}_b\Big(K_tK_a\Big(\boldsymbol{p}-\boldsymbol{p}_d\Big)\\ &+\Big(\dot{\boldsymbol{p}}-\dot{\boldsymbol{p}}_d\Big)\Big)\Big) - m\boldsymbol{g},\end{aligned} \tag{5}$$

where K_tK_a, K_b *denote positive diagonal matrices, and* $\boldsymbol{\sigma}_a, \boldsymbol{\sigma}_b$ *are bounding functions, such that the position* $\boldsymbol{p}$ *converges to a desired reference* $\boldsymbol{p}_d$.

***Assumption** 1.* Given any desired force F_u, a quaternion $\boldsymbol{q}_d$ exists such that the quadrotor vertical thrust vector $\boldsymbol{F}_{th}$ can be rotated into $\boldsymbol{F}_u$. Therefore $\boldsymbol{q}_d \otimes \boldsymbol{F}_{th} \otimes \boldsymbol{q}_d^* = \boldsymbol{F}_u$

***Assumption** 2.* The rotational dynamics of system (1) are controlled such that the vehicle attitude $\boldsymbol{q}$ can follow at any reference as $\boldsymbol{q} \to \boldsymbol{q}_d$, implying that $\boldsymbol{q}\otimes\boldsymbol{F}_{th}\otimes\boldsymbol{q}^* \equiv \boldsymbol{F}_u$.

***Proof** 1.* Define a positive-definite function $V_1 : \mathbb{R}^3 \to \mathbb{R}^+$ as $V_1 \coloneqq \frac{1}{2}\dot{\boldsymbol{p}}_e \cdot \dot{\boldsymbol{p}}_e$ and its derivative as $\dot{V}_1 = \dot{\boldsymbol{p}}_e \cdot \big(m^{-1}\boldsymbol{F}_u + \boldsymbol{g}\big)$, where $\boldsymbol{p}_e = \boldsymbol{p} - \boldsymbol{p}_d$ describes the position error. Propose

$$\boldsymbol{F}_u = -mK_t\boldsymbol{\sigma}_a\Big(K_a\Big(\dot{\boldsymbol{p}}-\dot{\boldsymbol{p}}_d\Big) + K_b\boldsymbol{\varphi}_b\Big) - m\boldsymbol{g}\,, \tag{6}$$

where $\boldsymbol{\varphi}_b = [\varphi_{bx}, \varphi_{by}, \varphi_{bz}]^T$ denotes a vectorial bounded function with limits b_{xy} and b_z such that $|||[\varphi_{bx}, \varphi_{by}]||| <$

b_{xy}, $|\varphi_{bz}| < b_z$. $K_t, K_a, K_b \in \mathbb{R}^{3\times3}$ are constant positive-definite diagonal matrices, therefore

$$\dot{V}_1 = -\Big(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d\Big) \cdot K_t \boldsymbol{\sigma}_a \Big(K_a\Big(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d\Big) + K_b \boldsymbol{\varphi}_b\Big) \ . \quad (7)$$

Note that (7) can be analyzed in two cases as follows, since $\boldsymbol{\sigma}_a(K_a(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d) + K_b\boldsymbol{\varphi}_b)$ is bounded, then $\dot{V}_1 < 0$ if

$$\text{sign}(k_{ai}\dot{p}_{e_i} + k_{bi}\varphi_{bi}) = \text{sign}(\dot{p}_{e_i}) \Rightarrow |k_{ai}\dot{p}_{e_i}| > k_{bi}b_i \ , \quad (8)$$

where $k_{ai}\,, k_{bi}\,, i : x, y, z$ represent the diagonal entries of K_a and K_b, implying that there exists T_0, such that for any time $t > T_0$, $|k_{ai}\dot{p}_{e_i} + k_{bi}\varphi_{bi}| \leq 2k_{bi}b_i$. Therefore, bounds and gains have to be chosen as $a_{xy} > 2k_{bx}b_{xy}$, $a_{xy} > 2k_{by}b_{xy}$, and $a_z > k_{bz}2b_z$ to ensure that $\dot{V}_1 < 0$. Notice that for all $t > T_0$, a second case arises where $\boldsymbol{\sigma}_a(K_a(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d) + K_b\boldsymbol{\varphi}_b) = K_a(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d) + K_b\boldsymbol{\varphi}_b$, and (6) can be considered as

$$\boldsymbol{F}_u = -mK_t\Big(K_a\Big(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d\Big) + K_b\boldsymbol{\varphi}_b\Big) - m\boldsymbol{g} \, . \quad (9)$$

Proposing an auxiliar vector $\boldsymbol{\nu}_1 \in \mathbb{R}^3$ as

$$\boldsymbol{\nu}_1 \coloneqq K_tK_a(\boldsymbol{p} - \boldsymbol{p}_d) + (\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d), \quad (10)$$

$$\dot{\boldsymbol{\nu}}_1 = K_tK_a(\dot{\boldsymbol{p}} - \dot{\boldsymbol{p}}_d) + \big(m^{-1}\boldsymbol{F}_u + \boldsymbol{g}\big) \, , \quad (11)$$

introducing (9) into (11) yields,

$$\dot{\boldsymbol{\nu}}_1 = -K_tK_b\boldsymbol{\varphi}_b \ . \quad (12)$$

A second positive definite function $V_2 : \mathbb{R}^3 \to \mathbb{R}^+$ is then introduced as $V_2 = \frac{1}{2}\boldsymbol{\nu}_1 \cdot \boldsymbol{\nu}_1$ such that $\dot{V}_2 = -\boldsymbol{\nu}_1 \cdot K_tK_b\boldsymbol{\varphi}_b$. Propose $\boldsymbol{\varphi}_b = \boldsymbol{\sigma}_b(\boldsymbol{\nu}_1)$, thus $\dot{V}_2 = -\boldsymbol{\nu}_1 \cdot K_tK_b\boldsymbol{\sigma}_b(\boldsymbol{\nu}_1) < 0$.

From the previous, it is implied that $\boldsymbol{\nu}_1 \to [0,0,0]^T$ and $\dot{\boldsymbol{\nu}}_1 \to [0,0,0]^T$, then, from (12), it follows that $\boldsymbol{\varphi}_b \to [0,0,0]^T$, from (7), it yields $\dot{\boldsymbol{p}}_e \to [0,0,0]^T$ and $\boldsymbol{p}_e \to [0,0,0]^T$, hence ensuring stability for the translational dynamics.

3.2 Attitude Reference

In this subsection the shortest rotation between $\boldsymbol{F}_u$ and the main vertical thrust is computed, ensuring that Assumption 1 holds true.

Define $\boldsymbol{q}_t = e^{\frac{1}{2}\vartheta_t\boldsymbol{u}_t}$ as the quaternion that yields the shortest rotation between vectors $\boldsymbol{F}_{th}$ and $\boldsymbol{F}_u$, where $\boldsymbol{u}_t$ and ϑ_t denote respectively the axis and the angle of such rotation. Let $\boldsymbol{n}_z$ and $\boldsymbol{n}_u$ be the normalized vectors of $\boldsymbol{F}_{th}$ and $\boldsymbol{F}_u$ respectively (note $\boldsymbol{n}_z = [0\ 0\ 1]^T$ is constant), then their cross and scalar products are

$$\boldsymbol{n}_u \times \boldsymbol{n}_z = \boldsymbol{u}_t \sin(\vartheta_t) \ , \ \ \boldsymbol{n}_u \cdot \boldsymbol{n}_z = \cos(\vartheta_t). \quad (13)$$

Some known trigonometric functions can be used for cutting the rotation in half as

$$\cos\left(\frac{\vartheta_t}{2}\right) = \pm\sqrt{\frac{1+\cos(\vartheta_t)}{2}} \, , \, \sin\left(\frac{\vartheta_t}{2}\right) = \pm\sqrt{\frac{1-\cos(\vartheta_t)}{2}}. \quad (14)$$

Then the quaternion that aligns the main thrust to the controller direction can be designed using trigonometric identities and the Euler-Rodrigues formula as

$$\boldsymbol{q}_t \coloneqq \pm\left(\sqrt{\frac{1+\boldsymbol{n}_u\cdot\boldsymbol{n}_z}{2}} + \frac{\boldsymbol{n}_u \times \boldsymbol{n}_z}{||\boldsymbol{n}_u \times \boldsymbol{n}_z||}\sqrt{\frac{1-\boldsymbol{n}_u\cdot\boldsymbol{n}_z}{2}}\right). \quad (15)$$

Defining $||\boldsymbol{F}_{th}|| \coloneqq ||\boldsymbol{F}_u||^T$, then $\boldsymbol{F}_{th}^{\mathcal{I}} \to \boldsymbol{F}_u$ when $\boldsymbol{q} \to \boldsymbol{q}_t$.

Since $\boldsymbol{n}_z$ is always aligned with the vertical axis of $\mathcal{B}$, and the direction of (15) is defined by a cross product, then $\boldsymbol{q}_t$ will only rotate the vehicle in the xy plane. For imposing a desired heading angle, ψ_d, it is necessary to define $\boldsymbol{q}_z = \cos(\psi_d/2) + [0\ 0\ 1]^T\sin(\psi_d/2)$. Thus finally, the desired quaternion to stabilize the whole orientation of the aerial vehicle can be described as $\boldsymbol{q}_d \coloneqq \boldsymbol{q}_t \otimes \boldsymbol{q}_z$.

3.3 Rotational Bounded Algorithm

Define the error quaternion as $\boldsymbol{q}_e \triangleq \boldsymbol{q}_d^* \otimes \boldsymbol{q}$, such that if $\boldsymbol{q}_e \to \boldsymbol{q}_0 = 1 + [0\ 0\ 0]^T$, it implies that $\boldsymbol{q} \to \boldsymbol{q}_d$.

***Lemma** 2. The attitude subsystem in* (1) *converges asymptotically to a quaternion reference* $\boldsymbol{q}_d$ *that aligns the quadrotor thrust vector* $\boldsymbol{F}_{th}^{\mathcal{I}}$ *in the inertial frame with control force* $\boldsymbol{F}_u$ *from* (5)*, by the effects of a bounded controller defined as*

$$\boldsymbol{\tau} = \boldsymbol{\Omega}\times J\,\boldsymbol{\Omega} - JK_r\boldsymbol{\sigma}_c\,(2K_c\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} + K_d\boldsymbol{\sigma}_d\,(2\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} + 2K_rK_c\ln(\boldsymbol{q}_z^* \otimes \boldsymbol{q}_t \otimes \boldsymbol{q}))), \quad (16)$$

where K_r, K_c, K_d symbolize positive diagonal matrices, and $\boldsymbol{\sigma}_c, \boldsymbol{\sigma}_d$ represent bounding functions.

***Proof** 2.* Introducing a positive-definite function $V_3 \coloneqq 2\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} \cdot \boldsymbol{q}^* \otimes \dot{\boldsymbol{q}}$, thus $\dot{V}_3 = 2\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} \cdot J^{-1}(\boldsymbol{\tau} - \boldsymbol{\Omega} \times J\,\boldsymbol{\Omega})$. Propose

$$\boldsymbol{\tau} = \boldsymbol{\Omega} \times J\,\boldsymbol{\Omega} - JK_r\boldsymbol{\sigma}_c\Big(2K_c\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} + K_d\boldsymbol{\varphi}_d\Big) \ , \quad (17)$$

where $\boldsymbol{\varphi}_d = [\varphi_{dx}, \varphi_{dy}, \varphi_{dz}]^T$ denotes a bounded vector limited as $|||[\varphi_{dx}, \varphi_{dy}]||| < d_{xy}$, $|\varphi_{dz}| < d_z$. $K_r, K_c, K_d \in \mathbb{R}^{3\times3}$ are positive diagonal matrices, therefore $\dot{V}_3$ becomes

$$\dot{V}_3 = -2K_r\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} \cdot \boldsymbol{\sigma}_c(2K_c\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} + K_d\boldsymbol{\varphi}_d) \ . \quad (18)$$

Define an axis-angle error as $\boldsymbol{\vartheta}_e = 2\ln(\boldsymbol{q}_z^* \otimes \boldsymbol{q}_t^* \otimes \boldsymbol{q}) \in \mathbb{R}^3$ and propose $\boldsymbol{\nu}_2 \coloneqq 2\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} + 2K_rK_c\ln(\boldsymbol{q}_z^* \otimes \boldsymbol{q}_t \otimes \boldsymbol{q})$, thus following the same methodology from section 3.1, propose $V_4 = \frac{1}{2}\boldsymbol{\nu}_2 \cdot \boldsymbol{\nu}_2$ and choosing $\boldsymbol{\varphi}_d = \boldsymbol{\sigma}_d(\boldsymbol{\nu}_2)$, it implies that $\dot{V}_4 < 0$. Similarly, $\boldsymbol{\nu}_2 \to [0,0,0]^T$, $\boldsymbol{\varphi}_d \to [0,0,0]^T$, implying $\dot{V}_3 < 0$ and $2\boldsymbol{q}^* \otimes \dot{\boldsymbol{q}} \to [0,0,0]^T$ and finally $\boldsymbol{\vartheta}_e \to [0,0,0]^T$. The bounds of functions $\boldsymbol{\sigma}_c$ and $\boldsymbol{\sigma}_d$ are chosen as $c_{xy} > 2k_{dx}d_{xy}$, $c_{xy} > 2k_{dy}d_{xy}$, and $c_z > 2k_{dz}d_z$.

Following the Euler-Rodrigues formula, if the axis-angle representation of the attitude error converges to the zero vector, then

$$\boldsymbol{q}_e = \boldsymbol{q}_z^* \otimes \boldsymbol{q}_t^* \otimes \boldsymbol{q} \to 1 + [0,0,0]^T, \quad (19)$$

therefore, the quadrotor attitude quaternion converges to the desired reference, hence $\boldsymbol{q} \to \boldsymbol{q}_d$ implying that $\boldsymbol{F}_{th}^{\mathcal{I}} \to \boldsymbol{F}_u$. Finally,

$$\boldsymbol{F}_{th}^{\mathcal{I}} = \boldsymbol{q} \otimes \boldsymbol{F}_{th} \otimes \boldsymbol{q}^* \to \boldsymbol{q}_t \otimes \boldsymbol{q}_z \otimes \boldsymbol{F}_{th} \otimes \boldsymbol{q}_z^* \otimes \boldsymbol{q}_t^* = \boldsymbol{F}_u. \quad (20)$$

4. COORDINATED CIRCULAR TARGET TRACKING

The objective of this section is to propose a technique to track a moving UGV by proposing a trajectory that takes the UAVs to a formation around the target. In order to implement the algorithm in on-board embedded systems, the UAVs must compute the trajectory in real time.

For this example, the formation will be defined by a desired distance d_{iT}^d with respect to the UGV, and a separation $d_{ij}^d = 2d_{iT}^d$ between the UAVs (see Figure 3).

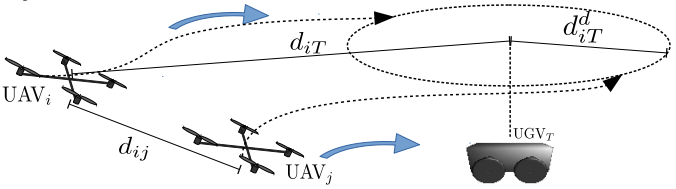

Figure 3. Two quadrotors take off from random positions and move to a symmetrical formation.

4.1 Distributed Path Planning Algorithm

A distributed path planning strategy is introduced to generate the trajectory that reaches the target, located at $p_T(t) \in \mathbb{R}^2$.

Let $\Lambda(t)$ define a positive cost function which value decreases when the horizontal position $p_i(t) \in \mathbb{R}^2$ of both quadrotors is such that $d_{iT}(t) \to d^d_{iT}$ and $d_{ij}(t) \to d^d_{ij}$. $\Lambda(t) := \sum_{i=1}^{n} \Lambda_i(t)$, is divided between the agents such that each agent i optimizes a part of the function using its own information and that of its neighbor j as:

$$\Lambda_i(h_i(t)) := \rho \Big| \|p_T(t) - [p_i(t) - \eta_i(t) + h_i(t)]\| - d^d_{iT} \Big| + a_{ij}(t) \Big| \|p_j(t) - [p_i(t) - \eta_i(t) + h_i(t)]\| - d^d_{ij} \Big|, \quad (21)$$

with $a_{ij}(t) = 1 + \exp((c - d_{ij}(t))/\gamma)$, being $i \neq j$ the index of each drone, $h_i(t) \in R^2$ represents the solution of the cost function for each quadrotor, $\rho, \gamma \in R^+$ denote tunning parameters, and c symbolizes their safety separation.

To achieve a path that surrounds the target while attaining the flight configuration, $\eta_i(t)$ will be designed, in this case of studio, to generate a circular movement around the UGV once it is reached by the UAVs. Considering $\eta_i(t) \perp (p_T - p_i(t))$ as a vector in $\mathbb{R}^2$ which is perpendicular to the radius between quadrotor i and the target as

$$\eta_i(t) := \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \left[[0\ 0\ 1]^T \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \left[\frac{(p_T(t) - p_i(t))}{\|(p_T(t) - p_i(t))\|} \right] \omega_i(t) \right] \quad (22)$$

where $\omega_i(t)$ defines a turning rate given by

$$\omega_i(t) := \omega_d \exp\Big(- \Big| \|p_j(t) - p_i(t)\| - d^d_{ij} \Big| - \Big| \|p_T - p_i(t)\| - d^d_{iT} \Big| \Big), \quad (23)$$

being $\omega_d \in \mathbb{R}$ a tunning parameter that regulates the speed and direction of the circular trajectory.

The optimal solution of (21) is found by following the PSO strategy as in Belkadi et al. [2019], which consists on finding a displacement $h^*_i(t) := \arg\min\big(\Lambda_i\big(h_{i,n}(t)\big)\big)$ for each drone $i = 1, 2$, while minimizing $\Lambda(t)$.

The construction of $\omega_i(t)$ is such that when the quadrotors are located far from the desired configuration, the turning rate diminishes exponentially, such that the vehicles can move to the required distances in contrast, the closer the UAVs get to a symmetrical formation, then $\omega_i(t) \to \omega_d$.

The minimum $\Lambda_i(h_i(t))$ is reached when both drones converge to positions such that

$$\|p_T(t) - [p_i(t) - \eta_i(t) + h_i(t)]\| \to d^d_{iT}$$
$$\|p_j(t) - [p_i(t) - \eta_i(t) + h_i(t)]\| \to d^d_{ij} \quad (24)$$

meaning that the optimal displacement $h^*_i(t)$ will converge to a perpendicular displacement with respect to the direction of the target $h^*_i(t) \to \eta_i(t)$, see Figure 4.

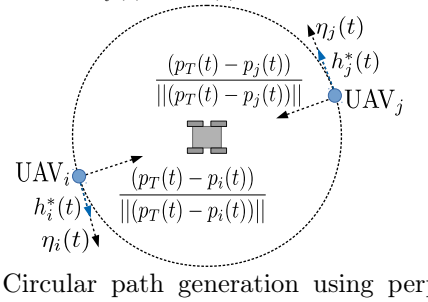

Figure 4. Circular path generation using perpendicular vectors on the optimization cost function.

Finally, the 3-dimensional trajectory is composed by adding $h^*_i(t)$ to the current quadrotor position, resulting in

$$\boldsymbol{p}_{di}(t) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} [p_i(t) + h^*_i(t)] + \begin{bmatrix} 0 \\ 0 \\ z_d \end{bmatrix}, \quad (25)$$

where z_d denotes the desired altitude over the target.

4.2 Target Locking

In the proposed scenario, the quadrotors are equipped with front-facing cameras. In order to maintain the target insight, the front axis of the quadrotors should be pointed towards the position of the UGV.

Define a normalized vector $\boldsymbol{\chi}_i \in \mathbb{R}^3$, which lies in the horizontal plane and goes from p_i to p_T.

$$\boldsymbol{\chi}_i = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \frac{p_T - p_i}{\|p_T - p_i\|}. \quad (26)$$

Considering that the front of the quadrotor points towards the x axis of the body frame, and following the idea of (15), a quaternion $\boldsymbol{q}_{z_i}$ which aligns each i vehicle's z axis with $\hat{m} = [1\ 0\ 0]^T$ is proposed as

$$\boldsymbol{q}_{z_i} := \pm \left(\sqrt{\frac{1 + \boldsymbol{\chi}_i \cdot \hat{m}}{2}} + \frac{\boldsymbol{\chi}_i \times \hat{m}}{\|\boldsymbol{\chi}_i \times \hat{m}\|} \sqrt{\frac{1 - \boldsymbol{\chi}_i \cdot \hat{m}}{2}} \right). \quad (27)$$

5. EMULATED RESULTS

The proposed algorithm was simulated in the "FL-Air" framework, developed at the Heudiasyc laboratory for two quadrotors. Figure 5 illustrates the simulated target movement, and the drones surrounding it. The arrows indicate the direction of their front axes.

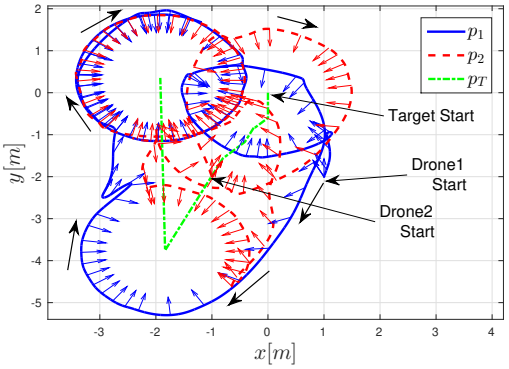

Figure 5. Emulation: Two quadrotors follow a target while pointing their fronts towards the objective.

Figure 6 illustrates the translational behavior of both drones on the x axis, fine-dotted lines illustrate the reference trajectories computed by the PSO algorithm.

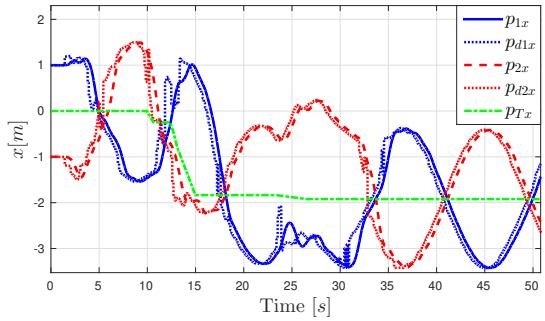

Figure 6. x axis translation, sinusoidal oscillations on both quadrotors indicate the circular target tracking.

6. EXPERIMENTAL VALIDATION

Real world tests were performed using two AR-Drone2 quadrotors and a motion capture system to track a UGV.

6.1 Stationary Target Scenario

On the first scenario, the target was left static while two quadrotors survey and surround it, the coordinated path resulted in circular trajectories, see Figure 7.

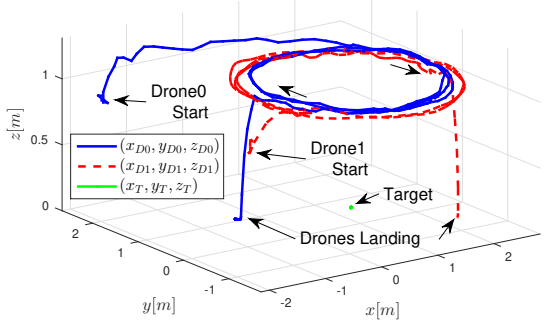

Figure 7. Two quadrotors surrounding a static UGV by describing circles.

A desired trajectory is computed by (25), controllers (5) and (16) then track it as shown in Figure 8. Note that the paths result in opposing positions for the UAVs.

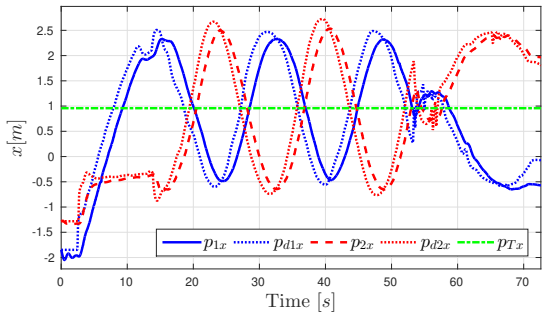

Figure 8. x axis position of both vehicles, fine-dotted lines represent the PSO-generated trajectory.

6.2 Moving Target Scenario

On the second experiment, the UGV was moved using a remote controller, the PSO algorithm generates trajectories for each quadrotor to reach the target and surround it with circular movements. If the target moves, the effects of (23) interrupt the quadrotors' circular paths until the formation is recovered, see Figure 9.

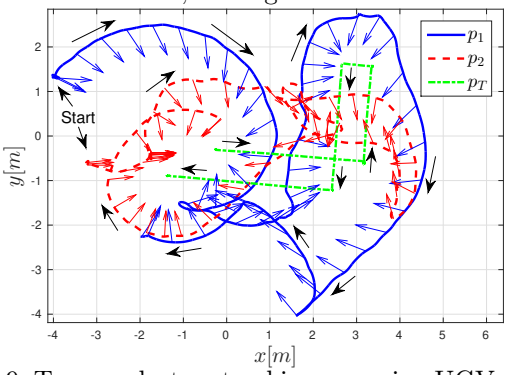

Figure 9. Two quadrotors tracking a moving UGV, colored arrows represent the front axes of each UAV.

Figure 10 illustrates the translation of the UGV and both quadrotors on the x and y axes, Notice when the target moves, the drones adjust their trajectories to reach opposing positions around it.

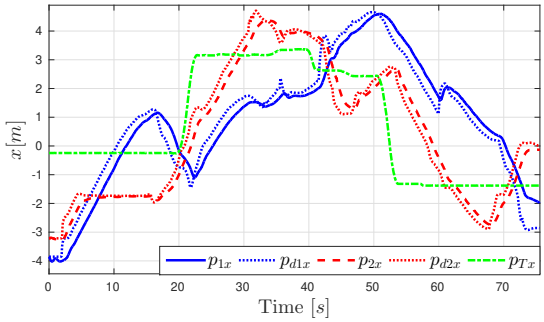

Figure 10. x axis response for a moving UGV test, The UAVs track the moving target while surrounding it.

6.3 Disturbed Agent Scenario

In the last test, the quadrotors were once again set to track a static target, but in this case, one of the drones was highly disturbed by pulling it by hand and placing it on different locations as depicted in Figure 11.

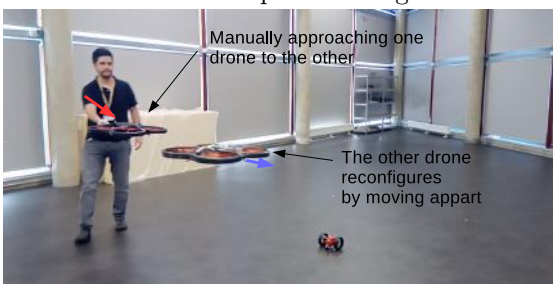

Figure 11. A drone is disturbed by displacing it, its neighbor adapts to recover the desired configuration.

On one hand, the trajectory generation algorithm provided robustness to the system by adapting the drones behavior and recovering the formation, on the other hand the proposed controllers ensure bounded force and control reactions, ensuring stability for the system. Figure 12 illustrates the upper view of the experiment, when one drone is disturbed, the other adapts its position according to the algorithm.

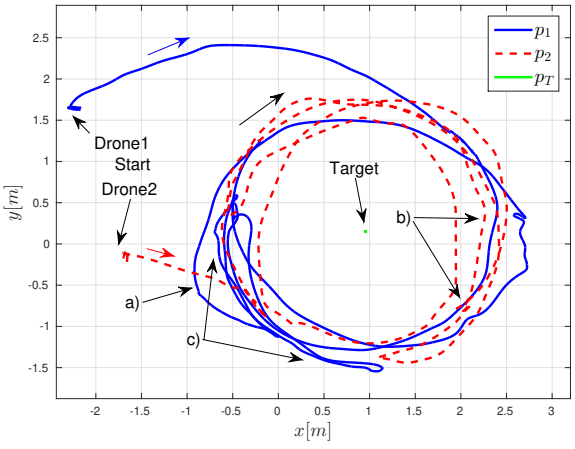

Figure 12. Disturbed experimet: a) Drone 1 is moved contrary from its path, b) Drone 2 is approached to Drone 1, c) Drone 1 is pushed towards Drone 2.

Finally, Figure 13 illustrates the separation distances between quadrotors (set at 3 meters) and towards the target. Notice that even if disturbances a) and b) reduce this distance the trajectory computed by the PSO algorithm adapts the trajectories to avoid colissions, and recovers to the desired value once the perturbation stops.

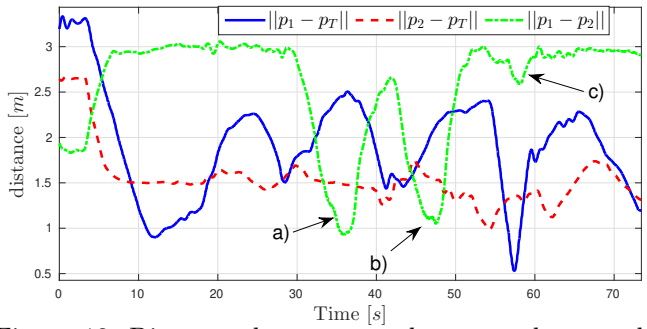

Figure 13. Distances between quadrotors and to to the UGV target are recovered disturbances.

All the previous experiments can be watched at the following link: `https://youtu.be/G31hZTr0m_s`

7. CONCLUSIONS

A cooperative navigation scheme was proposed in which multiple quadrotors autonomously track and surround ground target. The optimal solution of a distributed cost function that considers the desired UAVs behavior is found among the drones CPU, using the position signals of the drones and the target.

Lyapunov analysis was used to design a bounded controller, based on cylindrical functions. This approach ensures that the control action will be consistent on any direction of the x-y plane, and independent on the z axis.

Several real-time simulations and experiments, with multiple scenarios are used validate the proposals.

REFERENCES

Abaunza, H. and Castillo, P. (2019). Quadrotor aggressive deployment, using a quaternion-based spherical chattering-free sliding-mode controller. IEEE Transactions on Aerospace and Electronic Systems.

Bateman, A. and Lin, Z. (2003). An analysis and design method for linear systems under nested saturation. Systems & control letters, 48(1), 41–52.

Belkadi, A., Abaunza, H., Ciarletta, L., Castillo, P., and Theilliol, D. (2019). Design and implementation of distributed path planning algorithm for a fleet of uavs. IEEE Transactions on Aerospace and Electronic Systems.

Belkadi, A., Ciarletta, L., and Theilliol, D. (2016). Uavs fleet control design using distributed particle swarm optimization: A leaderless approach. In 2016 International Conference on Unmanned Aircraft Systems (ICUAS), 364–371. IEEE.

Brandao, A.S., Pizetta, I.H., Sarcinelli-Filho, M., and Carelli, R. (2012). High-level nonlinear underactuated controller for a leader-follower formation involving a miniature helicopter and a ground robot. In 2012 Brazilian Robotics Symposium and Latin American Robotics Symposium, 168–173. IEEE.

Cantelli, L., Mangiameli, M., Melita, C.D., and Muscato, G. (2013). Uav/ugv cooperation for surveying operations in humanitarian demining. In 2013 IEEE international symposium on safety, security, and rescue robotics (SSRR), 1–6. IEEE.

Castillo, P., Albertos, P., Garcia, P., and Lozano, R. (2006). Simple real-time attitude stabilization of a quadrotor aircraft with bounded signals. In Proceedings of the 45th IEEE Conference on Decision and Control, 1533–1538. IEEE.

Castillo, P., Lozano, R., and Dzul, A. (2005). Stabilization of a mini rotorcraft with four rotors. IEEE Control Systems Magazine, 25, 44–55.

Castillo, P., Lozano, R., and Dzul, A. (2004). Stabilization of a mini-rotorcraft having four rotors. In 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH3756z6), volume 3, 2693–2698. IEEE.

Cognetti, M., Oriolo, G., Peliti, P., Rosa, L., and Stegagno, P. (2014). Cooperative control of a heterogeneous multi-robot system based on relative localization. In 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems, 350–356. IEEE.

Diao, C., Xian, B., Zhao, B., Zhang, X., and Liu, S. (2013). An output feedback attitude tracking controller design for quadrotor unmanned aerial vehicles using quaternion. In 2013 IEEE/RSJ International Conference on Intelligent Robots and Systems, 3051–3056. IEEE.

Izaguirre-Espinosa, C., Muñoz-Vázquez, A., Sánchez-Orta, A., Parra-Vega, V., and Sanahuja, G. (2016). Fractional attitude-reactive control for robust quadrotor position stabilization without resolving underactuation. Control Engineering Practice, 53, 47–56.

Kempker, P.L., Ran, A.C., and van Schuppen, J.H. (2012). Controllability and observability of coordinated linear systems. Linear Algebra and Its Applications, 437(1), 121–167.

Kennedy, J. (2010). Particle swarm optimization. Encyclopedia of machine learning, 760–766.

Liu, H., Wang, X., and Zhong, Y. (2015). Quaternion-based robust attitude control for uncertain robotic quadrotors. IEEE Transactions on Industrial Informatics, 11(2), 406–415.

Meskin, N. and Khorasani, K. (2011). Fault detection and isolation: Multi-vehicle unmanned systems. Springer Science & Business Media.

Pang, S.K., Li, J., and Godsill, S.J. (2011). Detection and tracking of coordinated groups. IEEE Transactions on Aerospace and Electronic Systems, 47(1), 472–502.

Ran, M., Wang, Q., and Dong, C. (2016). Stabilization of a class of nonlinear systems with actuator saturation via active disturbance rejection control. Automatica, 63, 302–310.

Rendón, M.A. and Martins, F.F. (2017). Path following control tuning for an autonomous unmanned quadrotor using particle swarm optimization. IFAC-PapersOnLine, 50(1), 325–330.

Tanner, H.G. (2007). Switched uav-ugv cooperation scheme for target detection. In Proceedings 2007 IEEE International Conference on Robotics and Automation, 3457–3462. IEEE.

Tanner, H.G., Jadbabaie, A., and Pappas, G.J. (2005). Flocking in teams of nonholonomic agents. In Cooperative Control, 229–239. Springer.

Worthmann, K., Kellett, C.M., Braun, P., Grüne, L., and Weller, S.R. (2015). Distributed and decentralized control of residential energy systems incorporating battery storage. IEEE Transactions on Smart Grid, 6(4), 1914–1923.