2024 European Control Conference (ECC)
June 25-28, 2024. Stockholm, Sweden

A leader-follower strategy with distributed consensus for the coordinated navigation of a team of quadrotors in an environment with obstacles

Daniele Ravasio, Luca Bascetta, Gian Paolo Incremona, Maria Prandini

***Abstract*— This paper proposes a novel strategy for the coordination of a team of quadrotor unmanned aerial vehicles (UAVs) that has to reach a target area, while moving in an environment with obstacles. The strategy consists of a coordination phase, where the quadrotors assume a given formation, followed by a mission phase, where the UAV formation navigates to the target. In the latter phase, according to a leader-follower configuration, the leading agent receives from the ground station an obstacle-free trajectory to track, whereas the coordinated followers reconstruct and track their collision-free reference trajectories via a distributed consensus scheme. Finally, some simulation results are presented.**

***Index Terms*— Unmanned aerial vehicles, formation control, consensus, obstacle avoidance.**

I. Introduction

The coordinated navigation of multiple unmanned aerial vehicles (UAVs) has gained significant attention in control and robotics research thanks to its diverse applications in civilian and military settings [1]–[3]. In fact, UAVs can operate in hazardous conditions and execute missions requiring complex or onerous piloting. However, the complexity of certain missions (e.g., the transportation of significant payloads, or the search for objects over wide areas) is often beyond the capabilities of a single UAV, thus requiring the collaboration of multiple agents.

Coordinating a team of UAVs in an environment with obstacles results in a series of challenges, prompting the need for suitable strategies. First of all, joint decision-making is required to make UAVs reach the target while avoiding collisions with each other and obstacles. Furthermore, the adopted communication architecture plays a fundamental role in the design of the coordination strategy. In particular, a decentralized scheme, where UAVs communicate with their neighbours and only one of them communicates with a ground station, offers significant benefits in terms of reliability and range over a centralized one, where all communicate with the ground station. However, network connectivity has to be maintained throughout the mission [4].

Motivated by these issues, we address the coordination of a team of quadrotors that has to reach some target area while moving in an environment with obstacles known to the ground station. The ultimate goal is to enable tasks like surveillance in areas that are typically inaccessible or extremely difficult to reach for humans or ground robots. Coordinated navigation has been addressed with various approaches, such as virtual structure [5]. behaviour-based [6], and leader-follower [7], [8] strategies. The leader-follower approach – where a designated leader quadrotor tracks a given trajectory while the other quadrotors follow the leader maintaining a certain spatial configuration – is particularly popular due to its simplicity and flexibility, [9]. In [10], leader-follower navigation in a cluttered environment is addressed by combining model predictive control (MPC) and feedback linearization. However, MPC relies on a linearized model of the quadrotor dynamics, which can limit its effectiveness in handling aggressive manoeuvres. To ensure a fast convergence of the quadrotor controllers and enhance formation robustness, finite-time leader-follower formation control is addressed in [11], [12]. In particular, [12] introduces a distributed finite-time observer to enable the quadrotor team to reconstruct the position of the leader. Based on this observer, a sliding-mode finite-time controller with adaptive disturbance compensation is designed. However, the proposed implementation does not account for scenarios in which quadrotors may start far apart without initial means of communication, and require measurements of high-order time derivatives of position, which can be challenging to obtain during high-speed flight, [13]. To guarantee accurate tracking of aggressive manoeuvres and fast convergence, without requiring high-order derivative measurements, the authors of [14] propose a different approach to trajectory tracking for a single quadrotor, where high-order position derivatives are tracked using feedforward inputs based on the differential flatness of UAV dynamics.

In this paper, we propose a coordinated control strategy which integrates trajectory planning, distributed consensus, and trajectory tracking techniques. The novelty of the work lies in properly accounting for the interaction of these approaches, while considering communication constraints related to network connectivity. The merit of the proposed strategy is the capability of effectively exploiting available measurements, and addressing complex scenarios where the agents are initially far from each other and aggressive manoeuvres are required. From a computational point of view, it results in an easy-to-implement solution, thus being promising for real field implementation.

Notation: Given a vector $\mathbf{v} \in \mathbb{R}^n$, $\mathbf{v}^\top$ is its transpose and v_i is its ith entry. Given two vectors $\mathbf{v} \in \mathbb{R}^n$ and $\mathbf{w} \in \mathbb{R}^n$,

This work has been supported by the Italian Ministry of Enterprises and Made in the framework of the project 4DDS (4D Drone Swarms) under grant no. F/310097/01-04/X56.

D. Ravasio, L. Bascetta, G. P. Incremona and M. Prandini are with the Dipartimento di Elettronica, Informazione e Bioingegneria, Politecnico di Milano, 20133 Milan, Italy (e-mail: {*daniele.ravasio, luca.bascetta, gianpaolo.incremona, maria.prandini*}*@polimi.it*). D. Ravasio is also with the Istituto di Sistemi e Tecnologie Industriali Intelligenti per il Manifatturiero Avanzato, Consiglio Nazionale delle Ricerche, 20133 Milan, Italy (e-mail: *daniele.ravasio@stiima.cnr.it*).

$\mathbf{v} \circ \mathbf{w}$ denotes the Hamilton product. $SO(3)$ denotes the three-dimensional rotation group. Finally, given $\alpha \geq 0$, we define $\text{sig}^{\alpha}(s) = |s|^{\alpha} \text{sign}(s)$, $s \in \mathbb{R}$, where $\text{sign}(s) = 1$ if $s > 0$, $\text{sign}(s) \in [-1, 1]$ if $s = 0$, and $\text{sign}(s) = -1$ if $s < 0$.

II. Proposed two-phase solution

We propose to structure the coordinated navigation of a team of n quadrotors into two phases: (i) a formation phase leading the team in formation, and (ii) a mission phase, where the team reaches the target while keeping in formation.

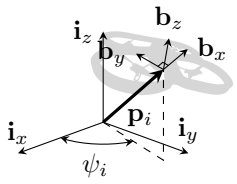


Fig. 1. Quadrotor frame description.

Consider Figure 1, where a fixed inertial frame $\{\mathbf{i}_x, \mathbf{i}_y, \mathbf{i}_z\}$ and a body-fixed frame $\{\mathbf{b}_x, \mathbf{b}_y, \mathbf{b}_z\}$, attached to quadrotor i centre of mass, are shown.

Under the assumption that when in formation all quadrotors keep their z-axes $\mathbf{b}_z$ parallel, a formation of n quadrotors with quadrotor 1 as leader and the others as followers can be represented by a set of $n-1$ vectors $\delta\boldsymbol{\sigma}_i = \begin{bmatrix}\delta\mathbf{p}_i^{\top} & \delta\psi_i\end{bmatrix}^{\top}$, $i = 2, \ldots, n$, that include the relative displacement $\delta\mathbf{p}_i = \begin{bmatrix}\delta x_i & \delta y_i & \delta z_i\end{bmatrix}^{\top}$ and the relative attitude $\delta\psi_i$, i.e., the linear and angular distances between the ith follower and the leader. The linear distance is computed considering the centre of mass of the two quadrotors, and the angular displacement as the rotation around $\mathbf{b}_z$ that aligns the two body-fixed reference frames.

During both the formation and the mission phase, the ground station is in charge of trajectory generation. In the formation phase, collision-free trajectories are generated for all quadrotors jointly and communicated to them according to a centralized communication architecture. This kind of communication is required because quadrotors may either be initially located far apart, and unable to communicate with each other, or unable to coordinate speeds and attitudes. In the mission phase, only the reference trajectory of the leader $\boldsymbol{\sigma}_1^{\star}$ is generated, while accounting for the volume occupied by the formation for obstacle avoidance. This trajectory is communicated only to the leader. Each follower derives its collision-free trajectory $\boldsymbol{\sigma}_i^{\star}$ by shifting the leader trajectory according to the relative distance specified in the formation structure, i.e., $\boldsymbol{\sigma}_i^{\star}(t) = \boldsymbol{\sigma}_1^{\star}(t) + \delta\boldsymbol{\sigma}_i$, $i = 2, \ldots, n$.

Since during the mission phase the team maintains the formation, we adopt a decentralized communication architecture inspired by [10]–[12], where the leader acts as the backbone, relaying the information from the ground station, and a distributed consensus protocol is considered to enable the team to reconstruct the leader's reference trajectory, in finite time, exploiting multi-hop communications. Since in both phases each quadrotor independently executes the assigned (or reconstructed) trajectory, accurate trajectory tracking is key to the effectiveness of the adopted strategy. As shown in Figure 2, this is achieved by combining the INDI approach [14] and differential flatness to track very aggressive trajectories with high accuracy.

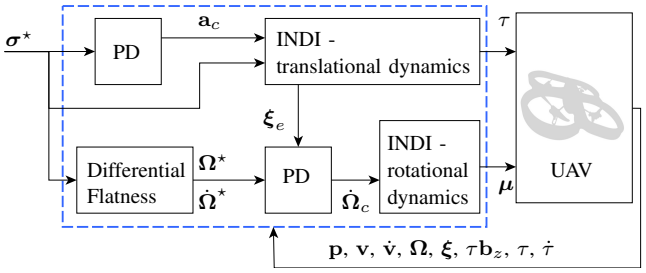


Fig. 2. Trajectory tracking controller block diagram.

III. Quadrotor dynamic model

We shall start introducing the quadrotor model [14]. Specifically, the considered longitudinal quadrotor dynamics is captured by the following equations

$$\begin{aligned}\dot{\mathbf{p}} &= \mathbf{v} \\ \dot{\mathbf{v}} &= g\mathbf{i}_z + \tau\mathbf{b}_z + m^{-1}\mathbf{f}_{\text{ext}},\end{aligned} \tag{1}$$

where $\mathbf{p}$ and $\mathbf{v}$ represent the position and velocity in the inertial reference frame, respectively, g is the gravitational acceleration, τ denotes the specific thrust (the ratio of total thrust T to the total mass m of the vehicle), and $\mathbf{f}_{\text{ext}}$ accounts for additional forces affecting the vehicle, such as aerodynamic drag.

The vehicle rotational dynamics is governed by

$$\begin{aligned}\dot{\boldsymbol{\xi}} &= \frac{1}{2}\boldsymbol{\xi} \circ \begin{bmatrix}0 \\ \boldsymbol{\Omega}\end{bmatrix} \\ \dot{\boldsymbol{\Omega}} &= \mathbf{J}^{-1}(\boldsymbol{\mu} + \boldsymbol{\mu}_{\text{ext}} - \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega}),\end{aligned} \tag{2}$$

where $\boldsymbol{\Omega}$ is the angular velocity in the body-fixed reference frame, $\boldsymbol{\xi} = \begin{bmatrix}\xi^w & \xi^x & \xi^y & \xi^z\end{bmatrix}^{\top}$ is the normalized attitude quaternion vector, so that $\mathbf{R}_B^I\mathbf{x} = \boldsymbol{\xi} \circ \begin{bmatrix}0 & \mathbf{x}\end{bmatrix}^{\top} \circ \boldsymbol{\xi}^{-1}$ where $\mathbf{R}_B^I = [\mathbf{b}_x \ \mathbf{b}_y \ \mathbf{b}_z] \in SO(3)$ is the rotation matrix providing the transformation from the body-fixed reference frame to the inertial reference frame, matrix $\mathbf{J}$ is the vehicle inertia tensor, $\boldsymbol{\mu}$ denotes the control moment vector, and $\boldsymbol{\mu}_{\text{ext}}$ indicates the external disturbance moment vector.

Note that, rather than explicitly accounting for aerodynamic effects, which significantly impact vehicle dynamics at high speeds, the considered model combines rotor drag and other disturbances into external forces and moments. This modelling choice enables the adoption of a tracking control law that does not rely on potentially inaccurate estimations of aerodynamic parameters.

A peculiarity of model (1)-(2) is that it is flatness linearizable for a particular selection of the flat outputs. More in detail, selecting flat outputs $\boldsymbol{\sigma} = \begin{bmatrix}\mathbf{p}^{\top} & \psi\end{bmatrix}^{\top}$, where ψ is the yaw angle, one can write states $\mathbf{p}$, $\mathbf{v}$, $\boldsymbol{\xi}$, $\boldsymbol{\Omega}$ and control

inputs τ and $\boldsymbol{\mu}$ as a function of $\boldsymbol{\sigma}$ and a finite number of its derivatives. Assuming no external forces and moments acting on the vehicle dynamics, i.e., $\mathbf{f}_{\text{ext}} = 0$ and $\boldsymbol{\mu}_{\text{ext}} = 0$, the specific thrust can be obtained by left-multiplying (1) by $\mathbf{b}_z^\top$, thus getting $\tau = \mathbf{b}_z^\top(\ddot{\mathbf{p}} - g\mathbf{i}_z)$, where $\mathbf{b}_z = (\ddot{\mathbf{p}} - g\mathbf{i}_z)/\|\ddot{\mathbf{p}} - g\mathbf{i}_z\|$. Defining vector $\mathbf{r}_\psi = [\cos(\psi)\ \sin(\psi)\ 0]^\top$, and imposing collinearity with vector $\mathbf{b}_y$, it is possible to write $\mathbf{b}_y = (\mathbf{b}_z \times \mathbf{r}_\psi)/\|\mathbf{b}_z \times \mathbf{r}_\psi\|$ and $\mathbf{b}_x = \mathbf{b}_y \times \mathbf{b}_z$. Then, the total moments vector $\boldsymbol{\mu}$ can be computed by inverting (2), thus getting $\boldsymbol{\mu} = \mathbf{J}(\dot{\boldsymbol{\Omega}} + \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega})$, where $\boldsymbol{\Omega}$ and $\dot{\boldsymbol{\Omega}}$ can be rewritten as functions of $\boldsymbol{\sigma}$ and a finite number of its derivatives, according to the following equations, [14]:

$$\begin{bmatrix}\boldsymbol{\Omega}\\ \dot{\tau}\end{bmatrix} = \begin{bmatrix}\tau\mathbf{R}_B^I[\mathbf{i}_z]_x^\top & \mathbf{b}_z\\ \mathbf{S} & 0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{p}^{(3)}\\ \dot{\psi}\end{bmatrix}, \tag{3}$$

$$\begin{bmatrix}\dot{\boldsymbol{\Omega}}\\ \ddot{\tau}\end{bmatrix} = \begin{bmatrix}\tau\mathbf{R}_B^I[\mathbf{i}_z]_x^\top & \mathbf{b}_z\\ \mathbf{S} & 0\end{bmatrix}^{-1}\begin{bmatrix}\mathbf{p}^{(4)} - \mathbf{R}_B^I(2\dot{\tau} + \tau[\boldsymbol{\Omega}]_\times)[\mathbf{i}_z]_\times^\top\boldsymbol{\Omega}\\ \ddot{\psi} - \dot{\mathbf{S}}\boldsymbol{\Omega}\end{bmatrix}, \tag{4}$$

where matrix $\mathbf{S}$ is such that $\dot{\psi} = \|\mathbf{r}_\psi \times \dot{\mathbf{r}}_\psi\|/\mathbf{r}_\psi^\top\mathbf{r}_\psi = \mathbf{S}\boldsymbol{\Omega}$. It is interesting to note that equations (3)-(4) relate the vehicle angular rates to the jerk $\mathbf{p}^{(3)}$ and the vehicle angular accelerations to the snap $\mathbf{p}^{(4)}$ and $\ddot{\psi}$, respectively.

IV. Trajectory planner

To generate the trajectories for the quadrotors, the algorithm proposed in [15] is adopted. The method, based on mixed-integer quadratic programming (MIQP), enables the generation of dynamically feasible trajectories in the flat output space, characterized by smooth transitions between n_w specified waypoints $\{\mathbf{p}^i, \psi^i\}_{i=0}^{n_w}$ at specified time instants $\{t_i\}_{i=0}^{n_w}$. These waypoints can be derived, for instance, from planning algorithms like RRT*, considering the quadrotor kinematic constraints. The method also allows modelling of obstacles, ensuring collision avoidance.

Exploiting a finite parameterization approach, the trajectory is described through piecewise polynomial functions of order n_p across n_w time intervals. This choice simplifies the application of waypoint constraints, while differentiability constraints are guaranteed by selecting the order of continuity between the subsequent polynomial pieces of the trajectory. Using this technique, optimal paths are generated up to the fourth order time derivative for the position, and up to the second order for the attitude.

The following cost function is then introduced for the ith quadrotor:

$$J_i = \int_{t_0}^{t_{n_w}} \left(w_{p,i} \left\| \frac{d^4\mathbf{p}_i}{dt^4} \right\|^2 + w_{\psi,i}\ddot{\psi}_i^2 \right) \mathrm{d}t,$$

where $w_{p,i}$ and $w_{\psi,i}$ are weighting parameters, and the index has been written with reference to the overall time interval $[t_0, t_{n_w}]$. Minimizing J_i allows to minimize the combined integral of snap (fourth-order position derivative) and the squared second-order yaw derivative. This choice stems from their influence on vehicle control torque and angular acceleration, as evident from differential flatness relations, allowing feasible trajectory generation.

To prevent collisions, the time interval between each pair of waypoints is discretized into n_k time steps, denoted as t_k, and vehicles and obstacles are represented as three-dimensional convex polygonal regions. In particular, each quadrotor is enveloped in a rectangular box large enough so that the quadrotor can roll, pitch, and yaw of any angle and stay within the box. This box is further inflated with an additional safety envelope, compensating for the trajectories discrete-time nature (see [15] for further details). As for obstacle avoidance, the condition that the ith quadrotor does not collide with the obstacle at time t_k translates into preventing collisions with at least one of the planes defined by the obstacle faces. This can be enforced by introducing as many binary variables $b_f \in \{0, 1\}$ as the obstacle faces $f = 1, \ldots, n_f$, that is

$$\begin{aligned}&\mathbf{n}_f \cdot \mathbf{p}_i(t_k) \le s_f + Mb_f, \quad f = 1, \ldots, n_f\\ &\sum_{f=1}^{n_f} b_f \le n_f - 1,\end{aligned} \tag{5}$$

where $\mathbf{n}_f$ is the normal vector to the face f of the obstacle, s_f is a scalar that determines the location of the plane defined by the face f, and M is a large positive number. In the formation phase, condition (5) is implemented for all quadrotors, while in the mission phase it is implemented only for the leader, while accounting for the space occupied by the entire team flying in formation. The size of the box enveloping the formation structure is defined by considering vehicle sizes, formation distances, and potential variations in quadrotor positions from the expected trajectory due to tracking errors.

As for inter-quadrotor collision avoidance in the formation phase, safety distances are enforced between each pair of quadrotors. Considering the ith and jth quadrotor trajectories, this condition can be enforced at time t_k by ensuring that safety distances are kept in at least one direction parallel to the axes of the inertial reference frame, i.e.,

$$\begin{aligned}&|x_i(t_k) - x_j(t_k)| \ge d_x - Mb_x\\ &|y_i(t_k) - y_j(t_k)| \ge d_y - Mb_y\\ &|z_i(t_k) - z_j(t_k)| \ge d_z - Mb_z\\ &b_x + b_y + b_z \le 2\end{aligned}$$

where d_x, d_y, d_z terms represent safety distances, b_x, b_y, b_z terms are binary decision variables, M is a number larger than any of the safety distances, and each absolute value constraint can be trivially rewritten as two linear constraints. To prevent the generation of trajectories that could lead to collisions when tracked, it is important to define safety distances that consider not only the vehicle size, but also the tracking accuracy of the controllers. An estimate of the bound on the trajectory tracking error can be obtained by means of simulations involving complex, high-speed trajectories.

In the formation phase, joint trajectories are generated for all quadrotors by minimizing the cost function given by $J = \sum_{i=1}^{n} w_i J_i$, where w_i is the weighting factor assigned to the

ith quadrotor trajectory. A higher weight corresponds to a more direct route from the start to the destination, whereas a lower weight prompts the quadrotor to adjust its trajectory to avoid interactions with other vehicles. Conversely, in the mission phase, the cost function $J = J_1$ is minimized, since, in this phase, only the trajectory of the leader is generated.

V. Distributed consensus strategy

In the mission phase, only the leader knows the reference trajectory to be tracked, while the followers should reconstruct it. To this purpose, the sliding mode distributed observer in [12] offers the advantage of decentralizing the exchange of information so as to enhance system robustness against link or agent failures, while enabling the followers to achieve the consensus on the leader trajectory within a finite time, under suitable assumptions on the communication architecture. The only drawback is that the proposed implementation requires that the leader communicates its full state to its neighbours, which might be unavailable due to unmeasurable components like jerk. To overcome this limitation, we assume that the leader perfectly tracks its assigned trajectory, and communicates this trajectory to its neighbours instead of its actual position. The observer is then used to reconstruct the leader reference trajectory, i.e., the position and its derivatives up to the fourth order, along with the attitude and its derivatives up to the second order.

Let $\mathbf{p}_1 = \begin{bmatrix} x_1 & y_1 & z_1 \end{bmatrix}^\top$ and ψ_1 denote the position and yaw angle references assigned to the leader quadrotor, respectively, and define the variables $\mathbf{p}_{1,0} = \mathbf{p}_1$, $\mathbf{p}_{1,1} = \dot{\mathbf{p}}_1$, $\mathbf{p}_{1,2} = \ddot{\mathbf{p}}_1$, $\mathbf{p}_{1,3} = \mathbf{p}_1^{(3)}$, $\mathbf{p}_{1,4} = \mathbf{p}_1^{(4)}$, $\psi_{1,0} = \psi_1$, $\psi_{1,1} = \dot{\psi}_1$, and $\psi_{1,2} = \ddot{\psi}_1$. A model for the leader position and attitude can be introduced as follows:

$$\begin{aligned} &\dot{\mathbf{p}}_{1,0} = \mathbf{p}_{1,1} \quad \dot{\mathbf{p}}_{1,1} = \mathbf{p}_{1,2} \quad \dot{\psi}_{1,0} = \psi_{1,1} \\ &\dot{\mathbf{p}}_{1,2} = \mathbf{p}_{1,3} \quad \dot{\mathbf{p}}_{1,3} = \mathbf{p}_{1,4} \quad , \quad \dot{\psi}_{1,1} = \psi_{1,2}, \\ &\dot{\mathbf{p}}_{1,4} = \boldsymbol{\Delta}_p \qquad\qquad\quad \dot{\psi}_{1,2} = \boldsymbol{\Delta}_\psi \end{aligned}$$

where the inputs $\boldsymbol{\Delta}_p = \begin{bmatrix} \Delta_x & \Delta_y & \Delta_z \end{bmatrix}^\top$ and $\boldsymbol{\Delta}_\psi$ are considered unknown bounded disturbances. As in [12], it is assumed that the leader transmits information to at least one follower without receiving data from any follower, and that the communication graph among the followers is undirected. For the sake of clarity, we focus only on the reconstruction of the x component of the position. The distributed observer takes the following form:

$$\begin{aligned} \dot{\hat{x}}^i_{1,0} &= \hat{x}^i_{1,1} - k_1 \operatorname{sig}^{m_o/n_o}(e^{x,i}_{1,0}) - k_2 \operatorname{sig}^{n_o/m_o}(e^{x,i}_{1,0}) \\ \dot{\hat{x}}^i_{1,1} &= \hat{x}^i_{1,2} - k_1 \operatorname{sig}^{m_o/n_o}(e^{x,i}_{1,1}) - k_2 \operatorname{sig}^{n_o/m_o}(e^{x,i}_{1,1}) \\ \dot{\hat{x}}^i_{1,2} &= \hat{x}^i_{1,3} - k_1 \operatorname{sig}^{m_o/n_o}(\hat{e}^{x,i}_{1,2}) - k_2 \operatorname{sig}^{n_o/m_o}(e^{x,i}_{1,2}) \\ \dot{\hat{x}}^i_{1,3} &= \hat{x}^i_{1,4} - k_1 \operatorname{sig}^{m_o/n_o}(\hat{e}^{x,i}_{1,3}) - k_2 \operatorname{sig}^{n_o/m_o}(e^{x,i}_{1,3}) \\ \dot{\hat{x}}^i_{1,4} &= -k_1 \operatorname{sig}^{m_o/n_o}(e^{x,i}_{1,4}) - k_2 \operatorname{sig}^{n_o/m_o}(e^{x,i}_{1,4}) \\ &\quad - \rho^o \hat{\Delta}_{x_i} \operatorname{sign}(e^{x,i}_{1,4}), \end{aligned}$$

where $\hat{x}^i_{1,k}$ is the estimate of the x component of the leader position ($k = 1$) and of its derivatives up to the fourth order ($k = 2, 3, 4$), computed by follower i, k_1 and k_2 are positive real design parameters, m_o and n_o denote positive odd integers satisfying $m_o < n_o$, and $e^{x,i}_{1,k}$ denotes the local estimated error, defined as $e^{x,i}_{1,k} = \sum_{j=2}^{n} a_{ij}(\hat{x}^i_{1,k} - \hat{x}^j_{1,k}) + a_{i1}(\hat{x}^i_{1,k} - x_{1,k})$, with $a_{ij} = 1$ if there exists a direct communication between quadrotors i and j, and $a_{ij} = 0$ otherwise. Finally, the gains $\rho^o \hat{\Delta}_{x_i}$, with $\rho^o > 1$, are used to compensate for the unknown disturbance Δ_x. In particular, the robust gain $\hat{\Delta}_{x_i}$ is updated without exploiting information on the upper bound of the external disturbance, which is possibly unknown, according to the following adaptation law

$$\dot{\hat{\Delta}}_{x_i} = \begin{cases} \rho^o |\hat{e}^{x,i}_{1,4}| + \arccos |\hat{e}^{x,i}_{1,4}|, & 0 < |\hat{e}^{x,i}_{1,4}| < \nu \\ \rho^o |\hat{e}^{x,i}_{1,4}|, & \text{otherwise} \end{cases}$$

with $0 < \nu < 1$ and $\hat{\Delta}_{x_i}(0) > 0$.
By selecting k_1 and k_2 as defined in [12], the sliding mode distributed observer reaches the sliding manifold $e^{x,i}_{1,k} = 0$, achieving consensus on the leader position and its derivatives up to the fourth order, in finite time. Equivalent expressions are adopted to reconstruct $y_1(t)$, $z_1(t)$, and $\psi_1(t)$ components, and their derivatives.

VI. Trajectory tracking

During the formation and mission phases, each quadrotor has to track a reference trajectory $\boldsymbol{\sigma}^\star = \begin{bmatrix} \mathbf{p}^{\star\top} & \psi^\star \end{bmatrix}^\top$ determined by the trajectory generation approach in Section IV. In particular, such trajectories could be aggressive, i.e., characterized by high speeds and fast-changing directions, especially if quadrotors navigate in a cluttered environment subject to tight time constraints. Accurate tracking of complex trajectories and rapid convergence of the tracking error to zero are particularly important, since we assumed the leader effectively tracks its reference trajectory in the design of the distributed consensus protocol. Moreover, adopting a decentralized communication architecture in the mission phase requires quadrotors to keep similar mobility patterns for connectivity. Onboard controllers are thus equally tuned to preserve the overall formation stability.

In this work, we adopt the control law for trajectory tracking proposed in [14], which exploits the INDI technique. This approach transforms the nonlinear vehicle dynamics into a linear input-output map, while simultaneously applying incremental control inputs based on inertial measurements, to counteract disturbances including modelling errors and aerodynamic effects. To alleviate the effect of noise, all inertial measurements are low-pass filtered (subscript f is used to indicate filtered signals). Motor speed measurements are required to compute thrust and control moments [14].

With reference to Figure 2, INDI is used to feedback linearize the quadrotor translational and rotational dynamics. As for the translational dynamics, the external force is estimated from equation (1) as $\mathbf{f}_{\text{ext}} = m\,(\mathbf{a}_f - (\tau \mathbf{b}_z)_f - g\mathbf{i}_z)$, where $\mathbf{a}_f$ represents the gravity-corrected linear acceleration in the inertial frame. Plugging the expression for $\mathbf{f}_{\text{ext}}$ into (1), and defining the acceleration command as $\mathbf{a}_c = \mathbf{a}_f + \tau \mathbf{b}_z + (\tau \mathbf{b}_z)_f$, the vehicle translational dynamics are feedback linearized, resulting in $\ddot{\mathbf{p}} = \mathbf{a}_c$. As for the vehicle rotational

dynamics, an estimate for the external moment vector is obtained from equation (2) as $\boldsymbol{\mu}_{\text{ext}} = \mathbf{J}\dot{\boldsymbol{\Omega}}_f - \boldsymbol{\mu}_f + \boldsymbol{\Omega}_f \times \mathbf{J}\boldsymbol{\Omega}_f$. Plugging $\boldsymbol{\mu}_{\text{ext}}$ expression into (2) and defining the angular acceleration command as $\dot{\boldsymbol{\Omega}}_c = \dot{\boldsymbol{\Omega}}_f + J^{-1}(\boldsymbol{\mu} - \boldsymbol{\mu}_f)$, the vehicle rotational dynamics is linearized as $\dot{\boldsymbol{\Omega}} = \dot{\boldsymbol{\Omega}}_c$. This approach enables the implementation of a linear control scheme involving two cascaded PD controllers. The outer loop PD controller exploits position, velocity, and acceleration to compute the desired control acceleration $\mathbf{a}_c$. This value is then used within the inner loop to determine the thrust command and attitude error vector $\boldsymbol{\xi}_e$. At this level, the method also exploits differential flatness to enable jerk and snap tracking. In particular, since measurements of high-order derivatives of position and attitude tend to be unreliable, differential flatness relations are used to derive angular velocity and angular acceleration reference signals, denoted respectively as $\boldsymbol{\Omega}^\star$ and $\dot{\boldsymbol{\Omega}}^\star$, from jerk and snap references. Specifically, $\boldsymbol{\Omega}^\star$ is derived by rewriting equation (3) with respect to the jerk reference and $\dot{\psi}^\star$, while $\dot{\boldsymbol{\Omega}}^\star$ is computed by rewriting equation (4) with respect to the snap reference and $\ddot{\psi}^\star$. These signals are then incorporated into the PD inner control loop as feedforward terms, to generate $\dot{\boldsymbol{\Omega}}_c$. Finally, this command is exploited to generate the control moment vector.

VII. Case Study

In this section, the proposed scheme is assessed in simulation on a realistic scenario.

Consider a team of $n = 5$ quadrotors, with communication topology in Figure 3, whose parameters are m =0.74 kg, $\ell = 0.25\,\text{m}$, $J_x = 0.04\,\text{kg m}$, $J_y = 0.04\,\text{kg m}$, and $J_z = 0.08\,\text{kg m}$.

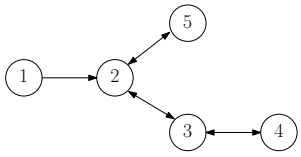
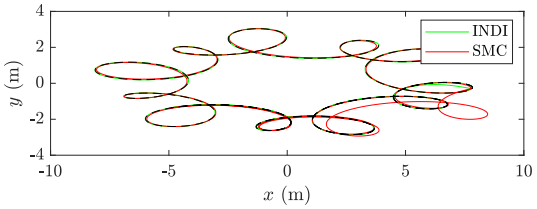
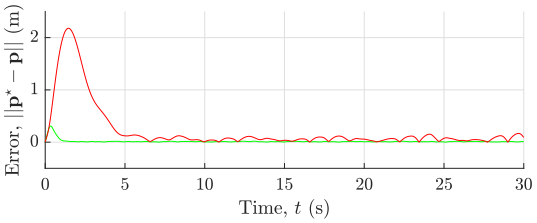


Fig. 3. UAVs' communication topology.

The gains of the trajectory tracking controllers are set as in [14]. The observer parameters are $m_o = 5$, $n_o = 7$, $\nu = 0.4$, and $\rho^o = 5$. The presence of aerodynamic disturbances of sinusoidal type is taken into account through the components $\mathbf{f}_{\text{ext}}$ and $\boldsymbol{\mu}_{\text{ext}}$. As for the mission phase, it consists in passing through specific waypoints at designated times, and reaching a target area in hovering conditions. The formation is specified by the vectors: $\delta\boldsymbol{\sigma}_2 = [3\ 3\ 0\ 0]^\top$, $\delta\boldsymbol{\sigma}_3 = [-3\ \ -3\ 0\ 0]^\top$, $\delta\boldsymbol{\sigma}_4 = [3\ \ -3\ 0\ 0]^\top$, and $\delta\boldsymbol{\sigma}_5 = [-3\ 3\ 0\ 0]^\top$. It is assumed that quadrotors operate at an altitude of 4 meters, and the formation is kept as a rigid structure with size 8×8 m, oriented along the $\mathbf{i}_x$ and $\mathbf{i}_y$ axes of the inertial reference frame, and centred at the leader centre of mass. Collision-free paths guiding the team of quadrotors to reach the formation at a prescribed time of 5 s are generated by using 10th order polynomials. Moreover, collision constraints are enforced at 11 intermediate time



(a) roulette curve



(b) position error

Fig. 4. Roulette curve trajectory and time evolution of the position error norm in the case of the INDI (green line) and SMC (red line) strategies.

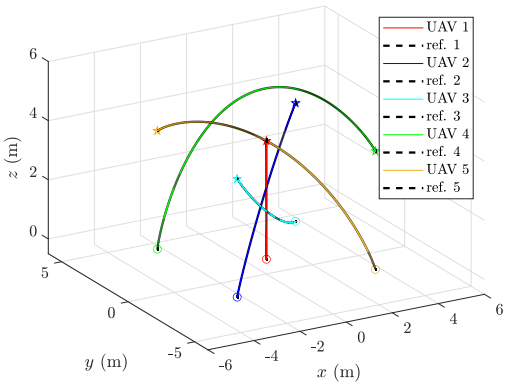
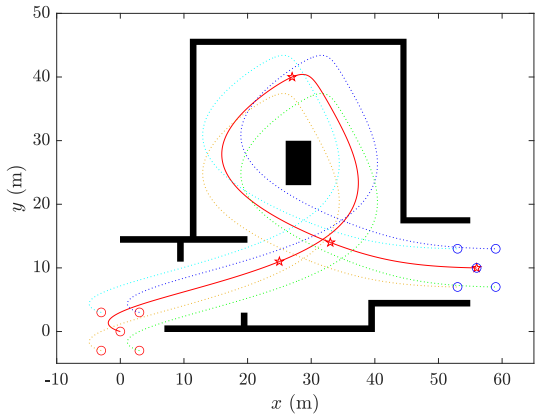


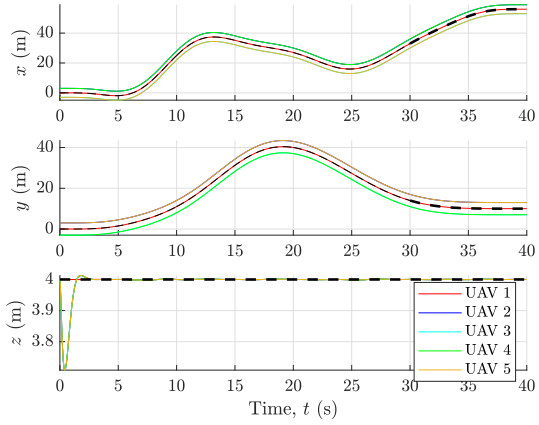
Fig. 5. Trajectories (solid lines) and references (dashed lines) of the UAVs team achieving the desired formation (circles are the initial positions and stars are the final ones).

steps. In the overall cost function, the contribution of the leader is weighted three times higher than those of the other quadrotors, so as to obtain a more direct path to reach the formation for the leader. The adopted solver is CPLEX, in MATLAB environment, and it takes 0.4 and 36 seconds for trajectory generation for the formation and mission phases, respectively.

To estimate a bound on the trajectory tracking error, necessary to enforce safety distances for trajectory generation, we make a quadrotor track a complex high-speed trajectory. In doing this, in order to further assess the proposed control scheme, we compare it with the tracking control technique in [12]. The latter exploits a sliding mode control (SMC) law, and, in contrast with our approach, needs jerk measurements, which are not obvious to retrieve or estimate during high-speed flight. Performance of the two controllers is assessed via the maximum errors of the UAV position and yaw angle with respect to their references, i.e., $e_{\mathbf{p}}^{\max} := \max(\|\mathbf{p}^\star - \mathbf{p}\|)$ and $e_{\psi}^{\max} := \max(|\psi^\star - \psi|)$, respectively, the corresponding root mean square errors (RMS) $e_{\mathbf{p}}^{\text{RMS}}$ and e_{ψ}^{RMS}, and the



(a) mission scenario



(b) position trajectories

Fig. 6. Top view of trajectories followed by the quadrotors during the mission phase (a), time evolution of the position trajectories (b).

integral squared errors (ISE) $e_{\mathbf{p}}^{\text{ISE}}$ and e_{ψ}^{ISE}.

As for the trajectory tracking performance, the quadrotor has to track the roulette curve trajectory, which is notably challenging due to its fast successive turns that lead to high jerk and snap. As it can be observed in Figure 4, the adopted INDI controller outperforms the SMC, which presents a significant initial overshoot and a higher error in steady-state. These results are also confirmed by the computed performance indexes, reported in Table I.

TABLE I

PERFORMANCE INDEXES.

	$e_{\mathbf{p}}^{\text{RMS}}$ (m)	$e_{\mathbf{p}}^{\max}$ (m)	$e_{\mathbf{p}}^{\text{ISE}}$ (m)	e_{ψ}^{RMS} (rad)	$e_{\psi}^{\max}$ (rad)	e_{ψ}^{ISE} (rad)
INDI	0.0372	0.308	0.042	0.015	0.199	0.007
SMC	0.5078	2.178	7.737	0	0	0

Figure 5 shows a 3D rendering of the collision-free trajectories of the UAVs, while they are attaining the designated formation, with an error of the order 1×10^{-3}. Figure 6(a) shows that each UAV reaches the designated target area while avoiding collisions. In Figure 6(b), it can be observed that the followers accurately reconstruct the leader trajectory, while keeping the formation.

VIII. CONCLUSIONS

In this paper, we addressed the challenge of coordinating a team of quadrotors that has to reach a target area while passing through an environment with obstacles. We developed an integrated framework structured in a coordination phase and a mission phase with a distributed consensus protocol relying on a leader-follower communication architecture. The adopted algorithm for the generation of collision-free trajectories at the ground station level and the INDI trajectory tracking controller complete the navigation scheme.

Future works could explore the navigation in partially known time-varying environments, e.g., by updating the obstacle map during flight, a switched-directed communication topology to enhance robustness against communication link failures, and the adoption of a less conservative approximation of the obstacles shape for trajectory planning.

REFERENCES

[1] S. Morris, E. W. Frew, and H. Jones, "Cooperative tracking of moving targets by teams of autonomous unmanned air vehicles," *Final report, University of Colorado at Boulder, MLB Company*, vol. 2551, 2005.

[2] B. J. O'Brien, D. G. Baran, and B. B. Luu, "Ad hoc networking for unmanned ground vehicles: Design and evaluation at command, control, communications, intelligence, surveillance and reconnaissance on-the-move," *ARL, Adelphi, MD, USA, Army Res. Lab. Tech. Rep. ARL-TR-3991*, 2006.

[3] D. W. Casbeer, D. B. Kingston, R. W. Beard, and T. W. McLain, "Cooperative forest fire surveillance using a team of small unmanned air vehicles," *International Journal of Systems Science*, vol. 37, no. 6, pp. 351–360, 2006.

[4] J. Li, Y. Zhou, and L. Lamont, "Communication architectures and protocols for networking unmanned aerial vehicles," in *2013 IEEE Globecom Workshops (GC Wkshps)*. IEEE, 2013, pp. 1415–1420.

[5] M. A. Lewis and K.-H. Tan, "High precision formation control of mobile robots using virtual structures," *Autonomous robots*, vol. 4, pp. 387–403, 1997.

[6] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE transactions on robotics and automation*, vol. 14, no. 6, pp. 926–939, 1998.

[7] K. A. Ghamry and Y. Zhang, "Formation control of multiple quadrotors based on leader-follower method," in *2015 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2015, pp. 1037–1042.

[8] D. Mercado, R. Castro, and R. Lozano, "Quadrotors flight formation control using a leader-follower approach," in *2013 European Control Conference (ECC)*. IEEE, 2013, pp. 3858–3863.

[9] J. Wang and M. Xin, "Integrated optimal formation control of multiple unmanned aerial vehicles," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1731–1744, 2012.

[10] W. Zhao and T. H. Go, "Quadcopter formation flight control combining MPC and robust feedback linearization," *Journal of the Franklin Institute*, vol. 351, no. 3, pp. 1335–1355, 2014.

[11] H. Du, W. Zhu, G. Wen, and D. Wu, "Finite-time formation control for a group of quadrotor aircraft," *Aerospace Science and Technology*, vol. 69, pp. 609–616, 2017.

[12] X. Ai and J. Yu, "Flatness-based finite-time leader–follower formation control of multiple quadrotors with external disturbances," *Aerospace Science and Technology*, vol. 92, pp. 20–33, 2019.

[13] J. Medrano, F. Yumbla, S. Jeong, I. Choi, Y. Park, E. Auh, and H. Moon, "Jerk estimation for quadrotor based on differential flatness," in *2020 17th International Conference on Ubiquitous Robots (UR)*. IEEE, 2020, pp. 99–104.

[14] E. Tal and S. Karaman, "Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 1203–1218, 2020.

[15] D. Mellinger, A. Kushleyev, and V. Kumar, "Mixed-integer quadratic program trajectory generation for heterogeneous quadrotor teams," in *2012 IEEE international conference on robotics and automation*. IEEE, 2012, pp. 477–483.