Multi-UAV Distributed Control for Reconfigurable Formations

Kostas Skantzikas, Lara Briñón-Arranz, Pierre Susbielle, Nicolas Marchand

Multi-UAV Distributed Control for Reconfigurable Formations

Kostas Skantzikas, Lara Briñón-Arranz, Pierre Susbielle and Nicolas Marchand

***Abstract*— This paper deals with cooperative formation control design for multi-UAVs. A new control strategy is developed to steer a group of robots to three kind of evenly spaced formations: circular, linear and circular sector. Our distributed approach enables real-time formation reconfiguration when the parameters of the formations change or when a robot leaves the formation. Communication among robots is used to maintain the robots equally spaced in the desired formation and to avoid collisions during the reconfigurations. Real world experiments with four mini aerial vehicles demonstrate the efficacy of the proposed formation control strategy.**

I. Introduction

The fields of robotics and autonomous systems have seen a surge in interest and development in the area of multi-UAV (Unmanned Aerial Vehicle) applications. Offering distinct advantages over single UAV setups, multi-UAV systems have recently found application in various real-life scenarios, notably in hazardous environments where they play a crucial role in safeguarding human lives from potential harm or injury. Surveillance and mapping, search and rescue, forest monitoring and fire prevention, disaster management and precision agriculture are promising tasks that can be executed by a UAV team [1]. Furthermore, in industrial environments, UAVs equipped with manipulation mechanisms are capable of helping man to carry out complex tasks [2].

While UAVs have been extensively studied by the robotics community over recent decades, and numerous flight controllers [3] and path planning techniques [4] for individual UAVs have been investigated, the complexities of multi-UAV systems present unique challenges that merit further exploration and study. Cooperative multi-UAV systems belong to the field of multi-robot and multi-agent systems. In this context, there is a considerable volume of theoretical literature addressing multi-agent problems and multi-UAV systems in particular [1], yet there is a scarcity of works delving into real-world applications.

An essential aspect of motion coordination in multi-robot systems is formation control. Formation control has increasingly garnered attention as a significant problem to be addressed. In [5], a survey of the various classifications of formation control in multi-agent systems is presented. Consensus and formation control share a common objective of achieving coordinated behavior among multiple agents. Therefore, several works address the formation control problem of a multi-robot system based on consensus control strategies [6], [7]. In [8], a consensus strategy is implemented to control the geometrical configuration of a group of three UAVs in a leader-follower approach.

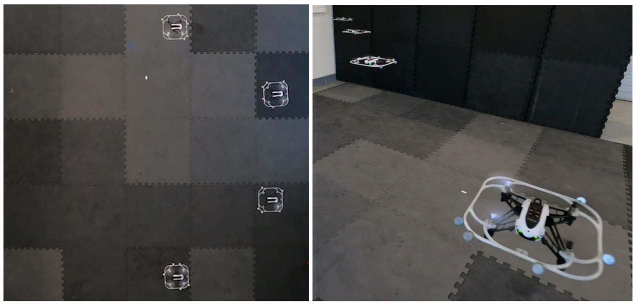

Fig. 1. Four mini UAVs forming two evenly spaced configurations: circular sector formation (left) and linear formation (right). Video of the real world experiments available at: *https://youtu.be/wTe1MbYiZxs*

The applications involving multi-UAVs require robustness and adaptability to face dynamic environments and complex tasks. In challenging missions, the multi-robot system may need to alter its configuration in order to adapt to the environment. The ability to switch among various configurations while ensuring robustness of the team to the failure of a robot is thus an essential condition for the multi-robot system. Therefore, an extremely challenging problem is to control multi-robot formations that can reconfigure. [9] provides an overview on recent developments in fault-tolerant cooperative control of multi-UAV to achieve the reconfiguration of the formation after unexpected failures.

Circular formation control has attracted significant research attention due to its diverse applications, such as target tracking and source-seeking missions. Circular collective motion of a network of unicycle-like agents was explored in [10] considering various communication constraints. Building upon these findings, [11] introduced an new control approach aimed at stabilizing time-varying circular formations. Following these ideas, an extension of these works to deal with reconfigurable formations for a team of mini-UAVs is presented in [12]. In this context, the challenging problem tackled in this work is the design of a distributed control strategy to coordinate a multi-robot system while ensuring safe transitions during the formation reconfigurations.

We propose a new formation control strategy for multi-UAV systems, designed to obtain three different evenly spaced configurations: linear, circular and circular sector. The contribution of this work is to provide a distributed strategy to enable real-time switching among the different formations while ensuring avoidance of collisions during transitions and

This work has been partially supported by ROBOTEX 2.0 (Grants ROBOTEX ANR-10-EQPX-44-01 and TIRREX ANR-21-ESRE-0015) funded by the French program Investissements d'Avenir.

All the authors are with Univ. Grenoble Alpes, CNRS, Grenoble INP, GIPSA-lab, 38000 Grenoble, France; Email: firstname.lastname@gipsa-lab.grenoble-inp.fr

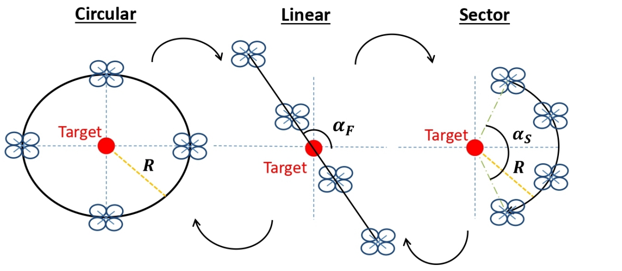

Fig. 2. The three uniformly distributed formations considered in this work: circular, linear and circular sector.

exhibiting robustness with respect to robots' failures. The proposed formation control is validated through real world experiments with a team of mini-UAVs.

II. Problem Formulation

A. Multi-UAV system

Consider a group of N aerial robots. The dynamic model of the UAVs used in this work is detailed in Section III-A. The 3D position of robot $i = 1,\ldots,N$ in the inertial global frame is denoted by $\xi_i = [x_i\ y_i\ z_i]^T \in \mathbb{R}^3$ and we denote by $p_i \in \mathbb{R}^2$ its position in the X-Y plane, such that $\xi_i = [p_i^T\ z_i]^T$. As presented in Section III-B, each UAV can be controlled to stabilize its state, position and velocity, to a desired reference.

In the adopted setup, each UAV can communicate with a set of neighboring robots. We assume that the communication network of the multi-robot system is represented by an undirected graph $\mathscr{G} = (V,E)$ where $V = \{1,\ldots,N\}$ denotes the set of vertices of the graph and E represents the set of edges such that $(i,j) \in E$ if robots i and j communicate with each other. Let $\mathscr{N}_i = \{j \in V \,|\, (i,j) \in E\}$ be the set of neighbors of robot i and $|\mathscr{N}_i|$ the number of its neighbors. In this paper, we consider a distance-dependent communication graph where the maximum communication distance is ρ_{comm}, i.e., $(i,j) \in E \iff \|\xi_i(t) - \xi_j(t)\| < \rho_{comm}$.

B. Uniformly distributed formations

The three formations considered in this paper are 2D configurations in the X-Y plane, shown in Fig. 2. For each configuration, the center of the formation is a given target position denoted by $p_T \in \mathbb{R}^2$. The circular formation is defined by a radius R. Regarding the linear configuration, the angle α_F represents the desired orientation of the formation with respect to the x-axis of the global inertial frame, such that $\alpha_F = 0$ when is aligned with the x-axis. The circular sector configuration is defined by radius R and angle α_S which determines the central angle of the circular sector.

The desired position in the formation for each UAV, denoted by p_i^{ref} is defined via a relative position vector between the UAV and the target position. To obtain a uniformly distributed configuration, the UAVs must be equally spaced along the formation. In other words, the distance between every pair of adjacency robots in the formation must be the same. These desired formation positions p_i^{ref} will be the position references to be tracked by the UAV's controller.

C. Control Objectives

The target position and the desired altitude of the UAVs are known for all the robots. Additionally, we assume that the desired formation (circle, line or circular sector) and the formation parameters R, α_F and α_S can be broadcasted in real-time. With these assumptions, the main control objectives are:

- **Objective 1**: Position tracking and state stabilization for all the UAVs in the team, i.e.,

$$\xi_i \to \xi_i^{ref} \quad \forall i = 1,\ldots,N \tag{1}$$

where ξ_i^{ref} is defined by the desired 2D position p_i^{ref} in the formation and a given hovering altitude.

- **Objective 2**: Uniformly distributed formation (circular, linear or circular sector) of the UAVs around a central target regardless the total number of UAVs in the team. The desired positions p_i^{ref} in the formation satisfy

$$\|p_i^{ref} - p_j^{ref}\| \to \lambda \tag{2}$$

for every pair (i,j) of adjacency robots in the formation, where $\lambda > 0$ denotes a constant value.

- **Objective 3**: Collision avoidance during transitions for all UAVs in the team, such that

$$\|\xi_i(t) - \xi_j(t)\| > \rho_c \quad \forall i,j \quad \text{and} \quad \forall t \tag{3}$$

where ρ_c is a given safe value.

D. Control Strategy

Our proposed control approach involves a formation generator responsible for producing the appropriate reference positions for each UAV combined with a position and velocity control to track these reference positions. To achieve a distributed control approach, each robot should compute and track its desired position in the formation by using only information from its neighbors. The formation generator of each UAV should be real-time reconfigurable in order to be robust to the failure of one or more robots. In other words, if a UAV fails and consequently leaves the team, the remaining UAVs must be able to reconfigure their positions to maintain an evenly spaced configuration. Moreover, the UAVs must be able to switch between formations while avoiding collisions.

The proposed control scheme to achieve these three objectives is shown in Fig. 3. The control strategy for real-time reconfigurable formations, is composed of three modules:

- The UAV state feedback controller, presented in Section III-B, makes each robot converge to its desired position computed by the formation generator.
- The formation generator, detailed in Section IV, determines the desired position for each aerial robot.
- The collision avoidance module, explained in Section V, controls the UAV's state to ensure that there is no collision with other robots.

III. UAVs Control Design

This section presents the model of the UAVs considered in this paper and the position and velocity control design ensuring each robot converges to its desired reference position.

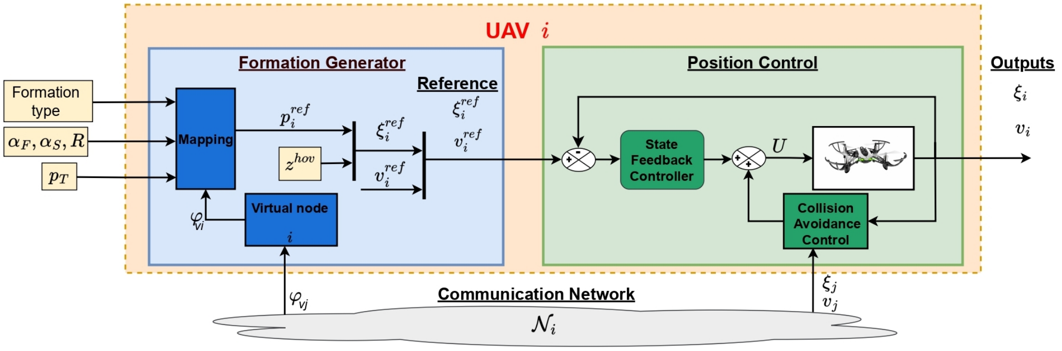

Fig. 3. Block diagram of the distributed control architecture.

A. UAV Model

The UAVs considered in this work are commercial quadrotors, aerial vehicles built on a rigid cross-shaped body structure with four counter-rotating rotors in conjunction with four propellers [13]. Each propeller $r \in \{1,2,3,4\}$ generates a thrust T_r and a torque. The total thrust T applied to the UAV is the summation of the four thrust forces generated by the rotors and Γ represents the total torque (see Fig. 4).

Let us define two basic coordinates reference frames: let $\{I\}$ be the Inertial reference frame, assumed as a flat earth frame, described by the unit vectors $\{\vec{i}_1, \vec{i}_2, \vec{i}_3\}$ and $\{B\}$ the body fixed frame, attached onto the UAV Center-of-Mass (CoM) moving along with its motion, with $\{\vec{b}_1, \vec{b}_2, \vec{b}_3\}$ unit vectors. According to the Z-Y-X Euler angles formalism the rotational matrix $\boldsymbol{R}^{B\to I} \in SO(3)$ expresses the orientation of the body fixed frame $\{B\}$ with respect to $\{I\}$ in the reference frame $\{I\}$, where s, c denote the sin() and cos() functions respectively and ϕ, θ and ψ denote the Euler angles [14]:

$$\boldsymbol{R}^{B\to I} = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix} \quad (4)$$

The Newton-Euler formulation is used to model the non linear rigid body dynamics of the quadrotor [15]:

$$\dot{\xi} = \upsilon \quad (5a)$$

$$m\dot{\upsilon} = -mg\vec{i}_3 + \boldsymbol{R}^{B\to I} T \vec{b}_3 \quad (5b)$$

$$\dot{\boldsymbol{R}}^{B\to I} = \boldsymbol{R}^{B\to I} \times \Omega \quad (5c)$$

$$\boldsymbol{J}\dot{\Omega} = -\Omega \times \boldsymbol{J}\Omega + \Gamma \quad (5d)$$

where $\upsilon := [\upsilon_x\ \upsilon_y\ \upsilon_z]^T$ denotes the speed vector of $\{B\}$ with respect to $\{I\}$ expressed in $\{I\}$, $\Omega := [p\ q\ r]^T$ denotes the angular rotation rate vector of $\{B\}$ with respect to $\{I\}$ expressed in $\{B\}$, $\Gamma := [\Gamma_r\ \Gamma_p\ \Gamma_y]^T$ denotes the torques applied to the UAV body, and the symbol $\times$ is the cross product between two vectors. T is the total thrust force produced by the UAV which is equal to the sum of the four thrust forces produces by each motor, $\boldsymbol{J}$ is the constant rigid-body moment of inertia matrix, m the total mass of the UAV and g the acceleration of gravity.

The nonlinear model (5) expresses the simplified dynamics of the UAV without taking into account phenomena like the gyroscopic effects, ground effects, blade flapping and air friction. Moreover, this model neglects the dynamics of the motors assuming them sufficiently fast, with respect to the dynamics of the UAV body [14].

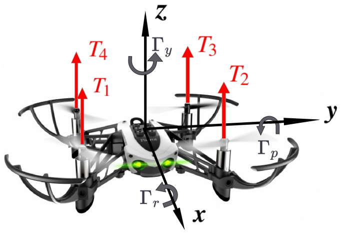

Fig. 4. Thrust forces and torques in a quadrotor Parrot Mambo.

Let us define the position ξ and the linear velocity υ as the UAV's state vector denoted by $X = [\xi^T\ \upsilon^T]^T$. The Euler angles and the thrust force expressed in $\{B\}$ are the control inputs, such that, $U = [\phi\ \theta\ \psi\ T]^T$. Linearizing the sub-system (5a)-(5b), which expresses the translational dynamics of the UAV, around a hovering position with zero yaw angle, $\psi = 0$ (see [14], [15], [16]), such that, $X^{eq} = [x^{eq}\ y^{eq}\ z^{eq}\ 0\ 0\ 0]^T$ and $U^{eq} = [0\ 0\ 0\ mg]^T$, the quadorotor translational linear model can be expressed as:

$$\dot{X} = \mathbf{A}X + \mathbf{B}U \quad (6)$$

where:

$$\mathbf{A} = \begin{bmatrix} 0_{3x3} & I_{3x3} \\ 0_{3x3} & 0_{3x3} \end{bmatrix}, \ \mathbf{B} = \begin{bmatrix} 0_{3x4} \\ \boldsymbol{B}_{lin} \end{bmatrix}, \text{ and } \mathbf{B}_{lin} = \begin{bmatrix} 0 & g & 0 & 0 \\ -g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/m \end{bmatrix}.$$

B. UAV Control Synthesis

A commonly used strategy in quadrotor control to track a reference trajectory is the hierarchical control approach [15]. We focus here on the position and velocity control of the UAV, considering that the inner control loops (rotors' speeds and attitude control) are correctly designed. Assuming that each UAV follows the linear model (6), a linear state-feedback control U can be designed to track a reference position. The linear acceleration of the UAVs CoM, given by the linear model (6), can be expressed as in [16]:

$$\ddot{x} = \theta g$$
$$\ddot{y} = -\phi g \tag{7}$$
$$\ddot{z} = \frac{T}{m}$$

Therefore, the variations in the UAV position along the x and y axes of $\{I\}$ are coupled with the changes in pitch and roll angles, while changes along the z-axis with the thrust. Consequently, the desired values for θ, ϕ and T must be derived based on the reference X-Y-Z respectively.

In this study, an efficient external control loop has been developed to regulate the UAV's position and velocity. The control goal of this control setup is to make each UAV follow a 3D reference trajectory in space by adjusting its Euler angles and thrust, which are considered as actuators with specific dynamic behavior. The reference position vector of each UAV CoM in space ξ^{ref} is constituted by the desired position p^{ref} in the X-Y plane computed by the formation generator and the desired hover altitude. The reference state vector is thus computed as:

$$X^{ref} = \begin{bmatrix} \xi^{ref} \\ \upsilon^{ref} \end{bmatrix} = \begin{bmatrix} p^{ref} \\ z^{hov} \\ 0_{3x1} \end{bmatrix} \in \mathbb{R}^6$$

To stabilize the UAV to the state reference, the Linear Quadratic Regulation synthesis (LQR) is applied, where the control input is:

$$U = K\left(X^{ref} - X\right) \tag{8}$$

with K a control gain chosen to minimize a quadratic cost based on the Riccati equation [16]. This linear control technique has proved its effectiveness to achieve high performance trajectory tracking [13], [16]. To ensure a precise tracking, an integral action has been added into (8) resulting in the following control input

$$U = K_P\left(\xi^{ref} - \xi\right) + K_D\left(\upsilon^{ref} - \upsilon\right) + K_I \int_0^t \left(\xi^{ref} - \xi\right) dt \tag{9}$$

In the next section we present how to compute the reference position ξ_i^{ref} for each robot i in the team, in order to achieve an evenly spaced configuration in a distributed way.

IV. Reconfigurable Formation Generator

The role of the formation generator described in this section is to compute the appropriate reference positions p_i^{ref} in the X-Y plane, in order to achieve the desired evenly spaced formation in a distributed way.

As previously presented, the center of the formation is located at a given target position p_T. Let us define $d_i \in \mathbb{R}^2$ as the desired relative position of robot i with respect to the target, such that, the desired position is defined as:

$$p_i^{ref} = p_T + d_i, \quad i = 1, \ldots, N. \tag{10}$$

The relative position d_i is expressed in polar coordinates:

$$d_i = R_i \left[\cos(\varphi_i) \quad \sin(\varphi_i)\right]^T \tag{11}$$

where R_i is a positive scalar and φ_i denotes the direction of the desired relative position in the X-Y plane.

Based on distributed synchronization algorithms, we develop a new framework to stabilize the group of robots to several reconfigurable formations. The main idea is to define the desired relative position for each robot i with respect to the target, i.e., d_i, as a transformation of the position vectors of an evenly spaced circular formation of virtual nodes. This approach allows defining circular, linear and sector formations in which the robots are evenly spaced. An appropriate transformation is designed to map the positions of the virtual nodes in the circular formation into the desired ones for the UAVs. As shown in Fig. 3, the formation generator combines the output of the virtual system with the formation parameters via an appropriate mapping to generate the reference positions for each UAV.

A. Distributed control of the virtual nodes

Our strategy relies on the use of a virtual system that can be stabilized to a uniformly distributed circular configuration around the origin with radius R. A virtual node vi is assigned to each robot i, whose position is given by

$$p_{vi} = R\left[\cos(\varphi_{vi}) \quad \sin(\varphi_{vi})\right]^T \tag{12}$$

where φ_{vi} denotes the orientation of the virtual position vector. The angle φ_{vi} becomes a dynamic variable and each robot is required to continuously transmit it to its neighbors, based on the distance-dependent communication graph $\mathscr{G}$. Each robot i receives the virtual variable φ_{vj} from the close robots that satisfy $\|\xi_i - \xi_j\| < \rho_{comm}$. The robots compute an algorithm based on the Kuramoto model for synchronization of coupled oscillators to make these virtual nodes achieve an evenly spaced distribution along a circle, using the information transmitted by their neighbors.

In particular, the following control law

$$\dot{\varphi}_{vi} = \frac{1}{|\mathscr{N}_i|} \sum_{j \in \mathscr{N}_i} \sum_{m=1}^{\lfloor \mathscr{N}_i/2 \rfloor} \frac{\sin\left(m\varphi_{vi} - m\varphi_{vj}\right)}{m} \tag{13}$$

where $\lfloor \mathscr{N}_i/2 \rfloor$ is the largest integer less than or equal to $\mathscr{N}_i/2$, makes the angles φ_{vi} to "desynchronize". In other words, the algorithm enforces the relative angles $\varphi_{vi} - \varphi_{vj}$ of two connected robots to enlarge.

For circular graphs, the evenly spaced configuration is locally asymptotically stable, as proven in [10]. Considering distance-dependent communication for the group of robots, we ensure that the evenly spaced configuration is the only possible stable equilibrium as proven in [11]. Using this distributed control law, the virtual system converges to an evenly spaced circular configuration, satisfying $\varphi_{vi} = \varphi_0 + \frac{2\pi}{N} i$, $\forall i = 1, \ldots, N$ where φ_0 is a constant.

B. Renconfigurable formations

In this section we define the transformations needed to stabilize the group of robots to three particular configurations based on the uniformly distributed circular formation of the virtual nodes.

1) Circular formation: In the simplest case, we stabilize the robots to an evenly spaced distribution along a circular formation with center the target position and radius R. The desired distances d_i are defined directly by the angular positions φ_{vi} of the virtual nodes and the radius of the circular configuration. In this situation, the mapping to obtain the circular formation is:

$$\begin{aligned} \varphi_i &= \varphi_{vi}, \quad \forall i = 1,\dots,N \\ R_i &= R \end{aligned} \tag{14}$$

Using directly the angular values of the virtual system, the final positions of the robots in the circular formation depend on the initial conditions of the virtual system and therefore they can be predefined.

2) Linear formation: The robots can be stabilized to a linear formation, i.e., the robots are disposed in a segment line whose midpoint corresponds to the position of the target and whose orientation with respect to the x-axis is defined by a given angle α_F. In order to transform the evenly spaced configuration of the virtual nodes into an evenly spaced linear formation, the introduction of an additional virtual node vl is required. We assume that all the robots have the information of this virtual node with constant value $\varphi_{vl} = \pi$.

This node is connected to all the virtual nodes vi and consequently, the set of neighbors of robot i becomes $\mathcal{N}_i^* = \{\mathcal{N}_i \cup vl\}$ and thus, the number of its neighbors becomes $|\mathcal{N}_i^*| = |\mathcal{N}_i| + 1$. The distributed synchronization algorithm (13) to obtain a uniform distribution of the virtual nodes along a circle becomes

$$\dot{\varphi}_{vi} = \frac{1}{|\mathcal{N}_i^*|} \sum_{j \in \mathcal{N}_i^*} \sum_{m=1}^{[\mathcal{N}_i^*/2]} \frac{\sin(m\varphi_{vi} - m\varphi_{vj})}{m} \tag{15}$$

This equation enforces all the virtual nodes vi to form an evenly spaced configuration composed of $N+1$ nodes, the virtual node vl is fixed at $\varphi_{vl} = \pi$ and the rest of the virtual nodes satisfy $\varphi_{vi} = \pi + \frac{2\pi}{N+1} i$.

In order to transform the evenly spaced circular configuration of the virtual nodes to a linear configuration, first of all the virtual angular values φ_{vi} are wrapped to the interval $[-\pi, \pi]$. Then, the following mapping:

$$\begin{aligned} \varphi_i &= \alpha_F \quad \forall i = 1,\dots,N \\ R_i &= R\varphi_{vi} \end{aligned} \tag{16}$$

allows defining each desired distance d_i to stabilize the group of robots to a line whose center is located at the target position. The distance vectors d_i depend on the orientation of the desired linear formation α_F.

3) Circular sector formation: Circular sector formations can be considered, i.e., all the robots are disposed in a circular arc whose center corresponds to the position of the target. The central angle of the sector is defined by a given constant α_S. In order to transform the evenly spaced virtual nodes into a circular sector configuration the introduction of an additional virtual node is required, as in the case of the linear formation. This node $\varphi_{vl} = \pi$ is connected to all the virtual nodes φ_{vi} and the distributed synchronization-based algorithm for the virtual system is (15).

The orientation of the desired distance vectors d_i depends on the virtual nodes and on the given central angle of the desired sector formation α_S. First of all, the virtual angular values θ_{vi} are wrapped to the interval $[-\pi, \pi]$ and then, the following mapping is applied:

$$\begin{aligned} \varphi_i &= \frac{\varphi_{vi}\alpha_S}{2\pi} \quad \forall i = 1,\dots,N \\ R_i &= R \end{aligned} \tag{17}$$

In all the aforementioned cases, the desired relative positions d_i of the UAVs are defined by (11) through the use of the neighboring virtual agents φ_{vi} and the given formation parameters, R, α_F, α_S.

V. Collision Avoidance Control

As presented in previous section, the formation generator computes for each UAV the reference position and the position controller ensures stability and tracking of this reference. However, because no global planner is considered for the trajectories of the UAVs, during the transient before the UAV team converges to its stable configuration, there could be situations leading to potential collisions among them:

- During the transient from initial conditions of the UAVs to their reference positions in the desired formation.
- During the transient when switching between two different possible formation configurations.
- Under the impact of external disturbances (e.g., wind).

To address these issues, the control strategy is improved by including a collision avoidance method to make the multi-robot system robust to potential collisions.

Force-field methods are commonly deployed and tested on UAVs to ensure that no collisions with any obstacle occurs [17], particularly in dynamic environments. In artificial potential field approaches, each UAV is treated as a particle moving within a repulsive potential field generated by obstacles in the environment [18]. By its nature, the repulsive potential increases as the UAV approaches an obstacle and decreases as moves away from it.

Let $\rho_{ij} = \|\xi_i - \xi_j\|$ denote the Euclidean distance between the CoM positions of UAVs i and j. Therefore, the potential field computed by robot i considering robot j as a dynamic obstacle can be expressed as in [19]:

$$V_{rep}(\rho_{ij}) = \begin{cases} \frac{1}{2} K_{rep} \left(\frac{1}{\rho_{ij}} - \frac{1}{\rho_0} \right)^2 & \text{if } \rho_{ij} \leq \rho_0 \\ 0 & \text{if } \rho_{ij} > \rho_0, \end{cases} \tag{18}$$

where K_{rep} is a gain to be tuned and ρ_0 is the influence distance of each UAV. The influence distance is introduced to enforce the potential function to be inactive if UAV i is far away from UAV j and thus it does not affect its motion. The minimum of this potential field is reached when $\rho_{ij} \geq \rho_0$. To reach this equilibrium point, the gradient of the potential field $F_{rep} = -\nabla V_{rep}$ is used for control the UAV. Therefore, each robot i computes the following repulsive force to control its

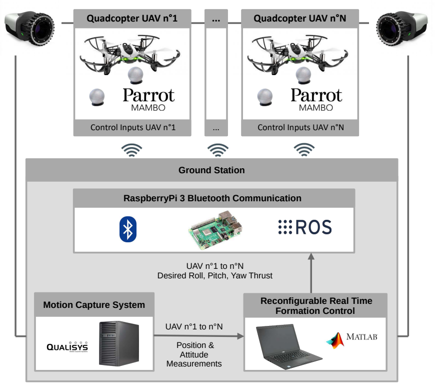

Fig. 5. Multi-UAV experimental setup

motion with the aim to avoid robot j, as in [19]:

$$F_{rep}(\rho_{ij}) = \begin{cases} K_{rep}\left(\frac{1}{\rho_{ij}} - \frac{1}{\rho_0}\right)\frac{\xi_i - \xi_j}{\rho_{ij}{}^3} & \text{if } \rho_{ij} \leq \rho_0 \\ 0 & \text{if } \rho_{ij} > \rho_0 \end{cases} \qquad (19)$$

An improved repulsive potential field has been proposed in [20] to take into account the relative velocity between the UAV and the obstacle. The improved repulsive force

$$\tilde{F}_{rep}(\rho_{ij}, \upsilon_i, \upsilon_j) = \begin{cases} F_{rep}(\rho_{ij}) - K^{\upsilon}_{rep}(\upsilon_i - \upsilon_j) & \text{if } \rho_{ij} \leq \rho_0 \\ 0 & \text{if } \rho_{ij} > \rho_0 \end{cases} \qquad (20)$$

introduces a second term which depends on the relative velocity between the two UAVs. The total repulsive potential field computed by each UAV will be the summation of the repulsive potentials generated by its neighboring UAVs in the group and thus, the total repulsive force for UAV i is:

$$F_{rep_i} = \sum_{j \in \mathcal{N}_i} \tilde{F}_{rep}(\rho_{ij}, \upsilon_i, \upsilon_j) \qquad (21)$$

The UAV simplified linear dynamics (7) express the Newton's second law of motion, therefore we can directly apply the resulting repulsive force form (21) $F_{rep_i} = [F^x_{rep_i}\ F^y_{rep_i}\ F^z_{rep_i}]^T$ to control the motion of the robots. As shown in Fig. 3, the repulsive control is added to the state feedback control U defined in (9). Assuming all UAVs hover at the same constant altitude, the combined controller to achieve objectives 1 and 3 is

$$\tilde{U} = U + \begin{bmatrix} -F^y_{rep_i}/g & F^x_{rep_i}/g & 0 & 0 \end{bmatrix}^T. \qquad (22)$$

VI. Experimental results

In this section we present the real world experiments which validate the presented distributed control strategy. The video showing these experimental results is available at *https://youtu.be/wTe1MbYiZxs*.

A. Experimental Setup

The experimental setup illustrated in Fig. 5 has been used to conduct the experiments. In this multi-UAV platform, developed at GIPSA-lab, four Parrot Mambo UAVs are connected via Bluetooth bridge to a ground station, which runs Matlab Simulink at 100Hz, implementing the proposed control architecture. Matlab Simulink also performs data acquisition from the capture motion system and generates the appropriate control signals for all the UAVs.

In this experimental setup, the formation control cannot be embedded into the Mambo UAVs. However, with a view to validate the proposed distributed strategy, the ground station computes N different controllers, one for each UAV in the team. The control inputs for each UAV are calculated by using only information from the UAV's neighbors. Therefore, this architecture allows demonstrate the performances of the distributed strategy. Because all the UAVs are hovering at the same altitude, we consider that two UAVs are neighbors if the 2D distance between them is less than $\rho_{comm} = 2m$. When a robot fails, communication is lost with other robots.

The control inputs for each robot are computed by using (22) and sent to the Mambos in real-time. Through an iterative trial-and-error process, the position controller (9) has been tuned by adjusting the $\boldsymbol{Q}$ and $\boldsymbol{R}$ matrices appropriately. For all the desired formations, the target p_T is located at the origin and the radius is $R = 0.8m$.

B. Scenario 1: Reconfigurable circular formation

In this scenario we evaluate the proposed algorithm with respect to a robot's failure. Positions of the four UAVs at three successive time intervals are shown in Fig. 6. The difference on opacity represents time evolution, so that the trajectories of the UAVs during the respective time interval are clearly displayed.

Firstly, the UAVs takeoff and hover at the desired hovering altitude $z^{hov} = 0.9m$. The formation generator algorithm starts running at $t = 15s$ and thanks to the position control the UAVs converge to their desired positions forming a uniformly distributed circular formation. UAV 1 leaves the formation at $t = 30s$, and the three remaining UAVs are able to reconfigure to achieve again an evenly spaced configuration. In particular, UAVs 2 and 3 move along the circle to reduce their distance and cover the space left by UAV 1. Once UAV 1 returns back to the group at $t = 50s$, the entire formation reconfigure to converge again to a uniformly distributed circular formation.

The time evolution of the distances between each pair of adjacency robots in the formation, as plotted in Fig. 7, aligns with Objective 2 as defined in Eq. (2) as all distances converge to a common value λ. For the circular formation this value depends on the number of robots in the formation, and can be analytically computed as $\lambda = 2Rsin(\frac{\pi}{N})$. In this case, $\lambda = 1.13$ when $N = 4$ and $\lambda = 1.39$ when $N = 3$, which are the values the UAV team reaches, as shown in Fig. 7.

C. Scenario 2: Circular, linear and sector formations

In this scenario, the effectiveness of the proposed algorithm to generate the three formation configurations (circle, line and circular sector) is experimentally verified. In the flight performed, the time evolution of the 2D trajectories of the four UAVs, illustrated in Fig. 8, shows the successful switching from one configuration to another.

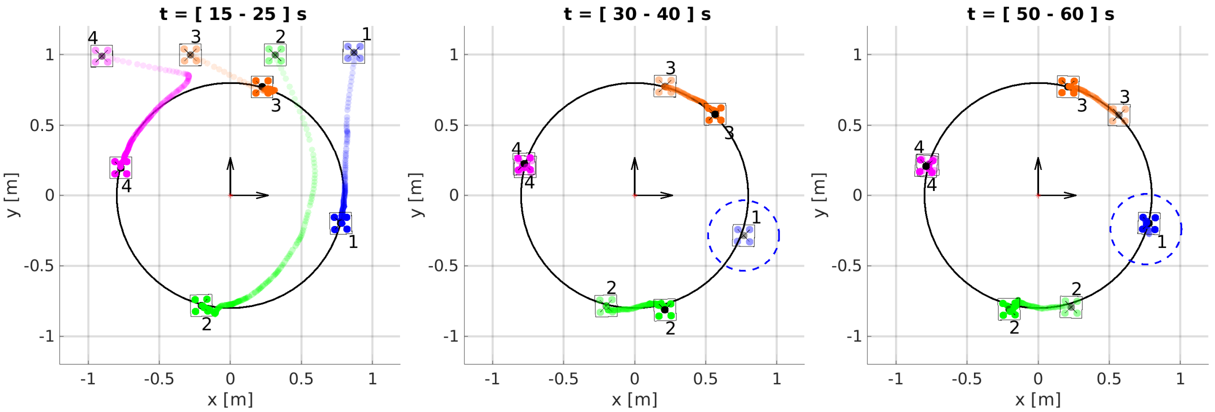

Fig. 6. Reconfigurable circular formation with four UAVs. UAV 1 leaves the formation at $t = 30$ and returns at $t = 50$.

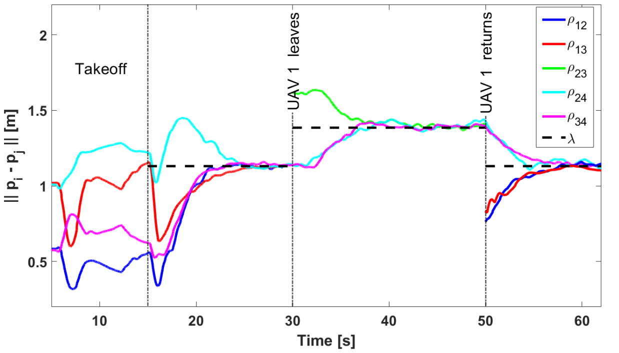

Fig. 7. Evolution of the distances among adjacency robots for Scenario 1.

As in Section VI-B, the four UAVs takeoff and hover at a constant altitude; for this scenario $z^{hov} = 0.8m$. At time $t = 15s$ the formation generator computes the references to make the robots converge to a circular formation while avoiding collisions. At $t = 30s$ the desired formation changes to a linear one with $\alpha_F = \frac{\pi}{4}$ and consequently, the UAVs move to their new reference positions generated by the formation generator algorithm. Similarly, at $t = 50s$ the formation is reconfigured to a circular sector with $\alpha_S = \frac{3\pi}{2}$ and as a result, the UAVs are distributed uniformly along the sector.

This experimental scenario emphasizes the effectiveness of the implemented repulsive forces algorithm which can handle the collision avoidance objective of this work. In this experiment, the additional collision avoidance controller based on repulsive potential fields, ensures safe transitions, from a particularly disadvantageous initial takeoff positions to the desired circular formation, as well as from one configuration to another. Fig. 8 illustrates the trajectories of the UAVs during these transitions showing how the UAVs modified their motion to avoid potential collisions.

Initially, at $t = 30s$ all UAVs start moving toward their respective reference positions. Notably, UAV 4 approaches UAV 3 leading to a potential collision between them. The collision avoidance controller remains inactive until the distance between them is less than $\rho_0 = 1m$. Under the impact of the repulsive term, UAVs 4 and 3 successfully maneuver to avoid collision. Then, they move away from each other towards their final positions in the linear formation. The time evolution of the distances between each pair of UAVs, plotted in Fig. 9 shows that the critical limit $\rho_c = 0.16m$ is never reached. Thus Eq. (3) is satisfied, validating Objective 3.

To show the effectiveness of the position control presented in Section III-B, Fig. 10 displays the time evolution of the position of UAV 4. The experimental data is compared with the desired reference to be tracked and with the simulation data. This experimental result confirms that Objective 1 is achieved as Eq. (1) is satisfied and shows that the proposed linear controller implemented in the more complex real UAV dynamical system ensures position tracking.

VII. Conclusion

In this paper, we proposed an holistic control scheme designed to enable a group of UAVs to be uniformly distributed around a designated target, along three different planar formations (circle, line and sector) while hovering at the same altitude. Our distributed approach relies on real-time information exchange among the UAVs within the network. The control strategy combines a synchronization-based formation generator algorithm with a linear state feedback position control to ensure convergence of the multi-UAV team to the desired evenly spaced formation. This strategy allows for a real-time formation reconfiguration. In addition, a collision avoidance mechanism ensures safe transition during the reconfiguration by making the robots keep a safety distance from each other.

Experimental validation of the proposed control architecture has been carried out with four mini-quadrotors. The experimental results confirmed the successful stabilization of the multi-UAV team to the reconfigurable desired formations.

References

[1] G. Skorobogatov, C. Barrado, and E. Salamí, "Multiple UAV Systems: A Survey," *Unmanned Systems*, vol. 08, no. 02, pp. 149–169, 2020.

[2] G. Li and G. Loianno, "Nonlinear Model Predictive Control for Cooperative Transportation and Manipulation of Cable Suspended Payloads with Multiple Quadrotors," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2023, pp. 5034–5041.

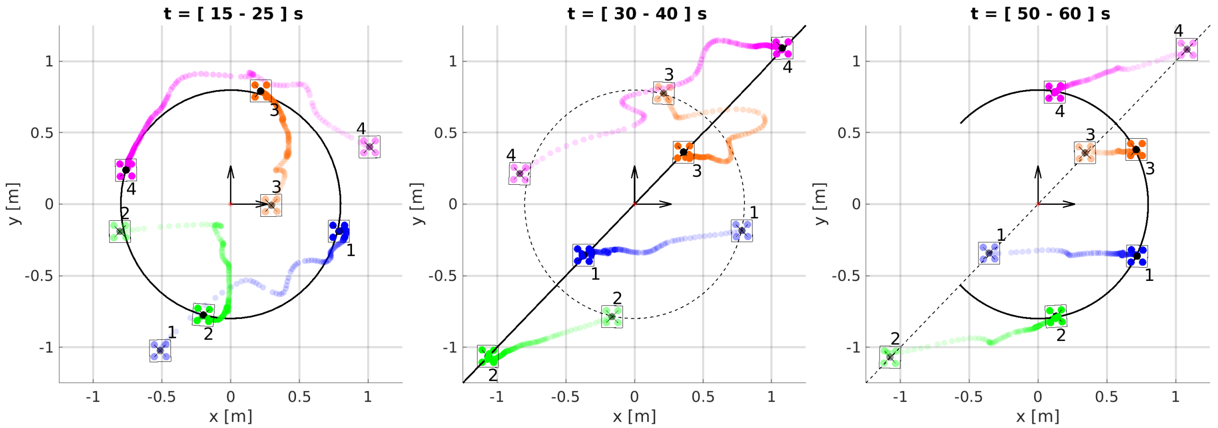

Fig. 8. Reconfigurable circular, linear and circular sector formations with four UAVs. Black solid lines represent the desired formation while black dashed lines represent the previous configuration before formation reconfiguration.

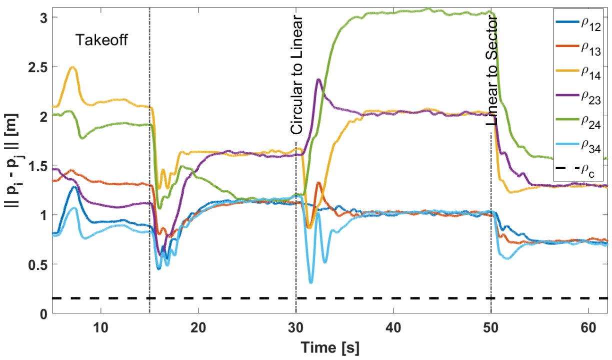

Fig. 9. Evolution of the distances ρ_{ij} between each pair of UAVs for Scenario 2 and the safe value ρ_c.

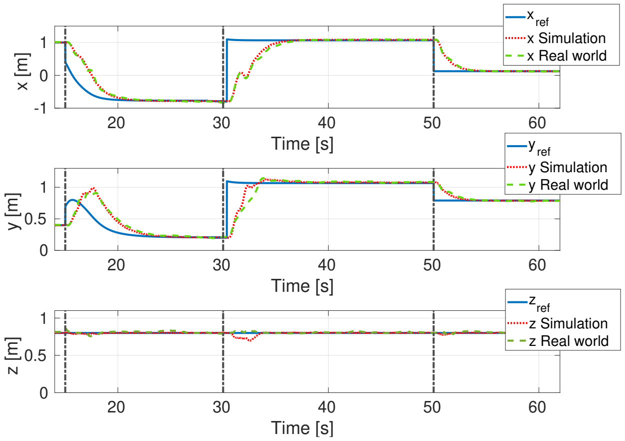

Fig. 10. Evolution of the three components of UAV 4 position in Scenario 2. Solid blue lines represent the references, dotted red lines the simulation data and dashed green lines the experimental data.

[3] Z. Zuo, C. Liu, Q.-L. Han, and J. Song, “Unmanned aerial vehicles: Control methods and future challenges,” *IEEE/CAA Journal of Automatica Sinica*, vol. 9, no. 4, pp. 601–614, 2022.

[4] S. Aggarwal and N. Kumar, “Path planning techniques for unmanned aerial vehicles: A review, solutions, and challenges,” *Computer Communications*, vol. 149, pp. 270–299, 2020.

[5] K.-K. Oh, M.-C. Park, and H.-S. Ahn, “A survey of multi-agent formation control,” *Automatica*, vol. 53, pp. 424–440, 2015.

[6] X. Dong, J. Xi, G. Lu, and Y. Zhong, “Formation control for high-order linear time-invariant multiagent systems with time delays,” *IEEE Trans. on Control of Network Systems*, vol. 1, no. 3, pp. 232–240, 2014.

[7] W. Ren and N. Sorensen, “Distributed coordination architecture for multi-robot formation control,” *Robotics and Autonomous Systems*, vol. 56, no. 4, pp. 324–333, 2008.

[8] A. A. Najm, I. K. Ibraheem, A. T. Azar, and A. J. Humaidi, “Genetic optimization-based consensus control of multi-agent 6-DoF UAV system,” *Sensors*, vol. 20, no. 12, 2020.

[9] Z. Yu, Y. Zhang, B. Jlang, J. Fu, and Y. Jin, “A review on fault-tolerant cooperative control of multiple unmanned aerial vehicles,” *Chinese Journal of Aeronautics*, vol. 35, no. 1, pp. 1–18, 2022.

[10] R. Sepulchre, D. A. Paley, and N. E. Leonard, “Stabilization of planar collective motion with limited communication,” *IEEE Trans. on Automatic Control*, vol. 53, no. 3, pp. 706–719, 2008.

[11] L. Briñón-Arranz, A. Seuret, and C. Canudas-de-Wit, “Cooperative control design for time-varying formations of multi-agent systems,” *IEEE Trans. on Automatic Control*, vol. 59, no. 8, pp. 2283–2288, 2014.

[12] L. Briñón-Arranz, M. Muschinowski, J. Dumon, and N. Marchand, “Distributed reconfigurable formation generator for mini aerial vehicles,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2018.

[13] S. Bouabdallah, A. Noth, and R. Siegwart, “PID vs LQ control techniques applied to an indoor micro quadrotor,” in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, vol. 3, 2004, pp. 2451–2456.

[14] E. Carvalho, P. Susbielle, A. Hably, J. S. Dibangoye, and N. Marchand, “Neural enhanced control for quadrotor linear behavior fitting,” in *Int. Conf. on Unmanned Aircraft Systems*, 2022, pp. 378–385.

[15] R. Mahony, V. Kumar, and P. Corke, “Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor,” *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[16] L. Martins, C. Cardeira, and P. Oliveira, “Linear quadratic regulator for trajectory tracking of a quadrotor,” *IFAC-PapersOnLine*, vol. 52, no. 12, pp. 176–181, 2019, 21st IFAC Symposium on Automatic Control in Aerospace.

[17] J. N. Yasin, S. A. S. Mohamed, M.-H. Haghbayan, J. Heikkonen, H. Tenhunen, and J. Plosila, “Unmanned aerial vehicles (UAVs): Collision avoidance systems and approaches,” *IEEE Access*, vol. 8, pp. 105 139–105 155, 2020.

[18] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” in *IEEE Int. Conf. on Robotics and Automation*, vol. 2, 1985, pp. 500–505.

[19] A. Azzabi and K. Nouri, “Path planning for autonomous mobile robot using the potential field method,” in *Int. Conf. on Advanced Systems and Electric Technologies*, 2017, pp. 389–394.

[20] S. Yan, F. Pan, J. Xu, and L. Song, “Research on UAV path planning based on improved artificial potential field method,” in *2nd Int. Conf. on Computer, Control and Robotics*, 2022, pp. 1–5.