AIAA SciTech Forum
6-10 January 2025, Orlando, FL
AIAA SCITECH 2025 Forum

10.2514/6.2025-2070

Experiments of formation flight for a medium-sized swarm of micro UAVs in a confined space with obstacle

Benjamin Piet, Mehmet Ablak, Guillaume Strub, Sébastien Changey
Institut Saint Louis (ISL), Saint-Louis, France

Nicolas Petit
Mines Paris PSL University, Paris, France

We present experimental results on formation flight within a confined space for a swarm of micro UAVs. The user selects a target for the swarm to reach, while maintaining a specified formation shape. This high-level requirement is addressed by a classical centralized control algorithm combining potential attractive, repulsive and gyroscopic forces to respectively maintain the formation and avoid collisions between UAVs and obstacles. To enable swift and fluid maneuvers in the confined environment, we propose an Integer Linear Programming methodology for real-time optimization of the formation's rearrangement. The control algorithms have been implemented and tested on a group of Crazyflies, which fly in an arrow-like formation occupying about 20% of the available area.

I. Nomenclature

N	=	Number of drones in the swarm
α	=	Desired orientation angle of the formation
d	=	Minimal distance between two point on the formation
$\mathbf{q}_i$	=	Position of the i^{th} drone
$\mathbf{q}_j^f$	=	j^{th} position in the formation
$\mathbf{q}^{goal}$	=	Position of the objective
$\mathbf{q}_j^{obs}$	=	Position of the j^{th} obstacle
$\mathbf{u}_d$	=	Desired group velocity for the swarm
$\mathbf{u}_i$	=	Commanded velocity for the i^{th} drone

II. Introduction

The subject of coordinated flight for a swarm of unmanned aerial vehicles (UAVs or drones) has garnered significant interest in the last decades, both for civil applications such as search and rescue, mobile sensor grids, and large-scale transportation, as well as for military purposes including surveillance, reconnaissance, and combat missions, see e.g. [1–4] for recent surveys. A primary challenge in coordinating a swarm of micro UAVs is ensuring that, while moving toward a user-defined target, the fleet maintains a specified formation shape and avoids collisions with environmental obstacles and between each other. This problem becomes all the more challenging when a confined working space is considered, leaving little room for movement and thus requiring accurate formation control.

The literature on the associated control problems is rich. For formation keeping tasks, various control schemes have been proposed, see e. g. [5, 6] and references therein. In *Virtual structure based methods*[7, 8], one defines a rigid structure formation that includes the drones and which is moved around, thus defining at the same time the trajectory that each drone needs to follow. In *Behavior based methods* [9], the control strategies changes according to the situation at hand and the tasks to be performed, e. g. tracking a target, avoiding an obstacle or reaching a given zone. In *Artificial potential field approaches* [8, 10–12], a potential field is defined that drives the UAVs towards attraction zones while creating repulsion between them to avoid collisions. In *Consensus-based strategies* [13–16] the states of all agents converge towards a common goal. In terms of formation control, the consensus is usually limited to the velocity of the agents. The swarm model is usually inspired from the pioneering work of Reynolds for representing flocks [17]. For convergence to target and obstacle avoidance, many techniques have also been developed. *Collision Cone based*

approaches [18–21] consider a collision cone defined for each obstacle, which represents a zone in which a collision with an obstacle is bound to happen if the velocity vector of the agent is in this zone. Once all collision cones are defined for the agent, the method consist in finding a velocity vector that is closest to its "preferred" velocity vector (for instance the one that brings him the closest to its objective) and remain outside of all the collision cone. *Model Predictive Control* (MPC), see e.g. [22–24] and references therein, considers direct transcription of the mission goals under the form of an optimal control problem where obstacles and possible collisions are introduced as constraints. Among many possibilities, polyhedral separations allow to employ convex optimization methods. MPC is a versatile technique that can also be used for formation keeping, the other drones acting as moving obstacles. Leaving the question of optimality of trajectories that are carefully treated by MPC, *Artificial potential fields* and *gyroscopic forces* is a more basic method that can be used to guide the drones away from the obstacles, see [11, 25–27] and reference therein.

In this paper, we consider a particular scenario of a medium-sized swarm of UAVs flying in a confined environment, tasked with reaching a user-defined target while keeping a specified formation shape (here an arrow) and avoiding obstacles, and collisions between each other along the way. Our desire is to develop an easy to implement method, scalable when the number of drones is increased (in future applications, up to 50). So to minimize the computational footprint, we do not rely on a MPC method and instead chose a twofold strategy. Firstly, inspired by the works of [8] and [11], we employ artificial potential fields and gyroscopic forces for formation keeping, convergence to target, collision and obstacle avoidance. Then, this solution is complemented by the main contribution of this paper, which is a proposed real-time optimization strategy for formation reconfiguration. Using Integer Linear Programming (ILP), we re-allocate at each control loop iteration the positions within the formation for all the UAVs. This proves particularly valuable when sudden changes of the target are required by the user. This re-allocation is illustrated in Fig. 1 for a case where the formation has to make a U-turn. Instead of having the whole formation turn around, a reordering within the formation allows to reorient the arrow in a swift and fluid manner.

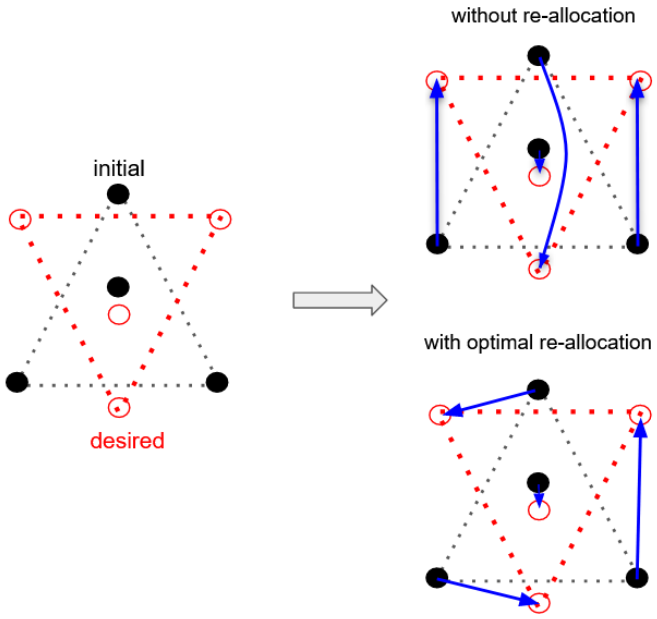

Fig. 1 Keeping or re allocating positions on the formation in the case of a 180° turn has an impact on the overall manoeuvre. To reorient the arrow-shaped formation, it is advantageous to consider the re allocation of the formation

The paper is organized as follows. In Section III, we present a mathematical formulation of the formation reconfiguration problem and solution. We also present the velocity control approach implementing artificial potential fields and gyroscopic forces. In Section III.E, we present the resulting control algorithm. Then in Section IV, we reports test results obtained on the Institut Saint-Louis (ISL) swarm test-bed. Details on the UAVs, sensing systems, implementation and tuning are provided. Section V sketches conclusions and perspectives.

III. Methodology

We will describe here the methods used for controlling each drone of the swarm in a way to complete the given objective, that is reaching a goal while avoiding obstacles and staying in a specified formation shape.

A. Problem definition

We consider a homogeneous swarm of UAVs, and we note $\mathbf{q}_i = [x_i, y_i, z_i]^T \in \mathbb{R}^3$ the position vector of each drone i with respect to a global reference frame. We also suppose that the number of drones in the swarm can change overtime, and we note N(t) the current number of active drone in the swarm at time t. To define the formation shape, we use a formation function that, given a parameter vector η, provides a list of positions that the drones should occupy to create the desired formation shape. We chose to take a 2D arrow-like shape on the [x,y] plane for our formation, which allows us to define an orientation angle α as the angle of the direction the arrow is pointing toward, and we also note d the minimal distance between two positions on the formation (see Fig. 2) The angle α could be chosen freely, but

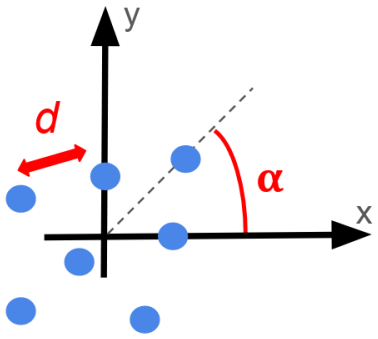

Fig. 2 Formation shape for N =7, with $\alpha = \frac{\pi}{4}$

will usually be chosen as the angle of the direction between the barycenter of the swarm's position and the target's position projected on the horizontal [x,y] plane. That way the arrow-like shape will always be pointing towards the current objective.

$$\alpha = angle([\mathbf{q}_{goal} - \frac{1}{N}\sum_{i=1}^{N}\mathbf{q}_i]_{[x,y]}, \mathbf{e}_x) \tag{1}$$

with $\mathbf{q}_{goal}$ the position of the objective, $[(v)]_{[x,y]}$ the projection of the vector $\mathbf{v}$ on the [x,y] plane, $\mathbf{e}_x$ the unit vector along the x axis, and $angle(\mathbf{v}, \mathbf{w})$ the function that gives the oriented angle between two 2D vectors $\mathbf{v}$ and $\mathbf{w}$.
Thus, considering the parameter vector $\eta(t) = (N(t), \alpha(t), d)$, we define the formation function as

$$\begin{aligned} F: \quad & \mathbb{R}^3 \to \mathbb{R}^{3\times N} \\ & \eta(t) \mapsto F(\eta(t)) = (\mathbf{q}_1^{f\,T}, ..., \mathbf{q}_N^{f\,T})^T \end{aligned} \tag{2}$$

with $\mathbf{q}_i^f = [x_i^f, y_i^f, z_i^f]^T \in \mathbb{R}^3$ the i^{th} position on the formation. The value of F are centered with respect to the origin of the reference frame.
To better visualize, Fig. 3 shows how the formation looks like depending on the number of drones. With this, our goal in formation control is to reach the following equality

$$R_{el}(\pi(X(t))) = R_{el}(F(N(t), \alpha(t), d))$$

with $\mathbf{X}(t) = (\mathbf{q}_1, ..., \mathbf{q}_N)^T$ the state of the drones at time t, $\pi(\mathbf{X})$ a permutation function, and $R_{el}(\mathbf{X})$ the relative position function, defined by

$$\begin{aligned} R_{el}: \quad & (\mathbb{R}^3)^N \to (\mathbb{R}^3)^{N\times N} \\ & (\mathbf{q}_1, ..., \mathbf{q}_N) \mapsto (\mathbf{q}_i - \mathbf{q}_j)_{1\le i,j\le N} \end{aligned}$$

The permutation function $\pi(X)$ allows for way to allocate each drone to a position on the formation. This is especially important in the case of a sudden change in the formation, or in the case of a loss of a drone. The method to find this permutation function is described in the next section.

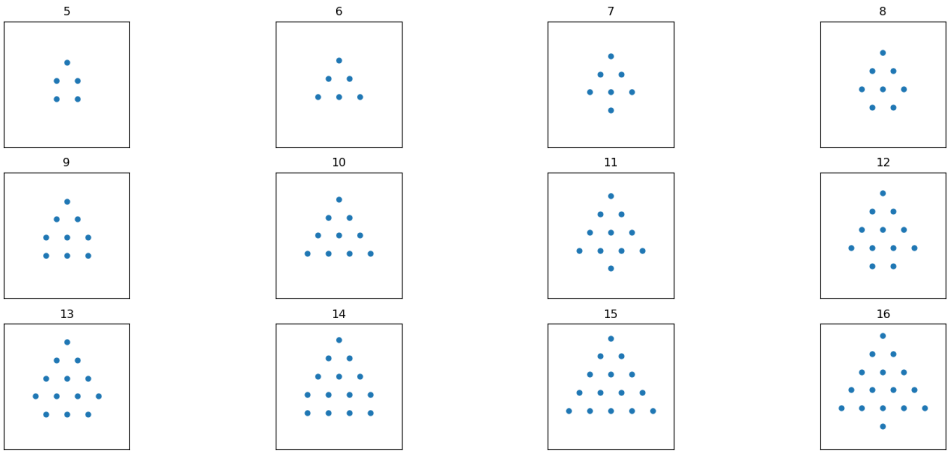

Fig. 3 Formation shapes for N ranging from 5 to 16, with $\alpha = \frac{\pi}{2}$

B. Reconfiguration

Since we consider a homogeneous swarm, it is possible to allocate a drone one position or another without issue. Because of this, we are free to find a position allocation that can minimize the total movement in the swarm, i.e. minimize the total distance between the current drone position and their respective allocated position on the formation. This could be useful when for instance the arrow-like shape needs to rotate by 180° . Not using position re-allocation would lead to the drone at the front having to cross the whole formation to reach its position, thus generating more possible collision. But with optimized position allocation, it could take a position at the back of the arrow and not move too much. To this end we have cast this position allocation problem as an optimisation problem.

An optimisation algorithm that finds the previously mentioned permutation function $\pi(X)$ that re allocate positions on the formation in an optimal manner in order to minimize the total movement in the swarm in case of change in the formation was thus implemented.

To do this, we will use Integer Linear Programming (ILP) optimisation that finds position allocation to minimize total distance between current position of the drones and its assigned position on the formation.

Let $\mathbf{q}_i$ be the position of the drone number i, and $\mathbf{q}_j^f$ the position number j on the formation centered around the origin of the global frame. We are looking for the permutation function $\pi(.)$ over the set $\{1, ..., N\}$ that minimizes the cost defined by

$$J = \sum_{i=1}^{N} \|\mathbf{q}_i - (\mathbf{q}^f_{\pi(i)} + \mathrm{B}(\mathbf{q}))\|^2$$

with $\mathrm{B}(\mathbf{q}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{q}_i$ the barycenter of the current position of the drones.

We introduce the cost matrix $\mathbf{M} \in \mathbb{R}^{N\times N}$ such that $\mathbf{M}_{ij} = \|\mathbf{q}_i - \mathbf{q}_j^f\|^2$, and the choice variable v_{ij} which is equal to 1 if the drone i is allocated to position j on the formation, 0 otherwise. The optimisation problem then becomes:

$$\begin{aligned} \min_{(v_{ij})_{i,j}} \quad & \sum_{i=1}^{N}\sum_{j=1}^{N} \mathbf{M}_{ij} v_{ij} \\ \text{s.t.} \quad & \sum_{j=1}^{N} v_{ij} = 1, \forall i \in \{1, ..., N\} \\ & \sum_{i=1}^{N} v_{ij} = 1, \forall j \in \{1, ..., N\} \\ & v_{ij} = 0 \text{ or } 1, \forall (i,j) \in \{1, ..., N\}^2 \end{aligned} \tag{3}$$

Due to the nature of the problem, the optimisation has at least one solution, but there is no guarantee that it can be unique. For instance Fig. 4 shows a situation where the two presented allocations have the same minimal cost. To

prevent any issue of oscillation between two or more possible solutions, the optimisation search is warm-started at the previously found solution. Once we have the optimal $\mathbf{V} = (v_{ij})_{ij}$ from the resolution of Eq. (3), the corresponding

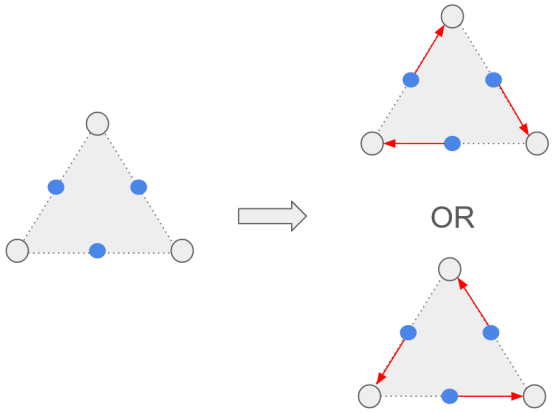

Fig. 4 Two possible position allocation with the same total cost

permutation function is simply $\pi(\mathbf{X}) = V.\mathbf{X}$.

C. Formation control

The method is taken from [8], and uses a potential function to keep the swarm in the wanted formation while avoiding collisions between the drones. The chosen model is

$$\dot{\mathbf{q}}_i = \mathbf{u}_i \quad \forall i \in (1, ..., N) \tag{4}$$

with $\mathbf{u}_i$ being the commanded velocity vector. The relative vector between the positions i and j on the formation is noted $\mathbf{l}_{ij} = \mathbf{q}_i^f - \mathbf{q}_j^f$

Each drone has to align its velocity vector with the global swarm vector $\mathbf{u}_d$ which is defined using an attractive force between the position of the goal $\mathbf{q}_{goal}$ and the barycenter of the swarm, with k_a a positive weight.

$$\mathbf{u}_d = k_a(\mathbf{q}^{goal} - \frac{1}{N}\sum_{i=1}^{N}\mathbf{q}_i)$$

Then, following the method described in [8], we consider the following potential function

$$\phi_i = \gamma_i + \delta\beta_i$$

with δ a constant positive tuning function, and γ_i and β_i defined as follows, with N_i the drones in the neighborhood of the drone i and k a positive constant:

$$\gamma_i = \sum_{j\in N_i}\gamma_{ij}, \quad \gamma_{ij} = \frac{1}{2}\|\mathbf{q}_i - \mathbf{q}_j - \mathbf{l}_{ij}\|^2$$

$$\beta_i = \sum_{j\in N_i}(\frac{\beta_{ij}^k}{\beta_{ijl}^{2k}} + \frac{1}{\beta_{ijl}^k}), \quad \beta_{ij} = \frac{1}{2}\|\mathbf{q}_i - \mathbf{q}_j\|^2, \quad \beta_{ijl} = \frac{1}{2}\|\mathbf{l}_{ij}\|^2$$

These two part each correspond to a part of the objective : γ_i is for making the drone reach the wanted formation shape, while β_i is for avoiding collision between the drones. The potential function is differentiated along the solutions of Eq. (4):

$$\dot{\phi}_i = \sum_{j\in N_i}\mathbf{\Omega}_{ij}^T(\mathbf{u}_i - \mathbf{u}_d) - \sum_{j\in N_i}\mathbf{\Omega}_{ij}^T(\mathbf{u}_j - \mathbf{u}_d)$$

with

$$\Omega_{ij} = \mathbf{q}_i - \mathbf{q}_j - \mathbf{l}_{ij} + \delta k(\frac{1}{\beta_{ijl}^{2k}} - \beta_{ij}^{2k})\beta_{ij}^{k-1}(\mathbf{q}_i - \mathbf{q}_j)$$

The command $\mathbf{u}_i$ is then chosen as

$$\mathbf{u}_i = \mathbf{u}_d - \mathbf{C}\sum_{j\in N_i}\Omega_{ij} \tag{5}$$

where $\mathbf{C} \in \mathbb{R}^{N\times N}$ is a symmetric positive definite matrix.
This control command allows us to keep the desired formation while avoiding collisions between drones. The proof for this result can be found in [8].

D. Obstacle avoidance

The above sections have discussed the control of a swarm of N UAVs towards a goal while keeping a desired formation and re-allocating positions on this formation to minimize the total movement required. In this section, we will add a way to avoid obstacles. To do this, repulsive and rotational forces are applied to each individual drone. For simplicity sake, the position and dimensions of all obstacles are assumed to be known.

1. Repulsive Force

First for the repulsive force, we use the classic repulsive potential field form [11]

$$U_{rep} = \begin{cases} \frac{1}{2}(\frac{1}{d(\mathbf{q},\mathbf{q}^{obs})} - \frac{1}{d_0})^2 & \text{if } d(\mathbf{q},\mathbf{q}^{obs}) \le d_0 \\ 0 & \text{if } d(\mathbf{q},\mathbf{q}^{obs}) > d_0 \end{cases}$$

with $d(\mathbf{q},\mathbf{q}_{obs})$ is the shortest distance between the current position of the drone $\mathbf{q}$ and the obstacle, and d_0 is the radius of influence of the obstacle (another tuning parameter). The potential is the sum of the potential for each obstacle, and it gives the following form for the repulsive force on drone i.

$$\mathbf{F}_i^{rep} = k_r \sum_{j\in N_i^{obs}} (\frac{1}{d(\mathbf{q}_i,\mathbf{q}_j^{obs})} - \frac{1}{d_0})\frac{1}{d^2(\mathbf{q}_i,\mathbf{q}_j^{obs})}\mathbf{nOD}_i^j$$

with N_i^{obs} the indices of the obstacles that are at a distance lower than d_0 to the drone i, $\mathbf{nOD}_i^j$ the unit vector going from the closest point of the obstacle j to the drone i, and k_r a constant positive weight.

2. Rotational force

Inspired by [11, 25], we add rotational forces to guide the drone around the obstacle, their form is (the forces are not derived from a potential field)

$$\mathbf{F}_i^{rot} = k_g \sum_{j\in N_i^{obs}} \begin{bmatrix} 0 & -\omega & 0 \\ \omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \mathbf{nOD}_i^j$$

with k_g a constant positive weight, and ω calculated as

$$\omega = \begin{cases} \frac{\pi V_{max}}{d(\mathbf{q},\mathbf{q}^{obs})} & \text{if } det[d(\mathbf{q},\mathbf{q}^{obs}),\mathbf{q}^{goal}-\mathbf{q}] \ge 0 \text{ and } d(\mathbf{q},\mathbf{q}^{obs}) < d_0 \\ \frac{-\pi V_{max}}{d(\mathbf{q},\mathbf{q}^{obs})} & \text{if } det[d(\mathbf{q},\mathbf{q}^{obs}),\mathbf{q}^{goal}-\mathbf{q}] < 0 \text{ and } d(\mathbf{q},\mathbf{q}^{obs}) < d_0 \\ 0 & \text{if } d(\mathbf{q},\mathbf{q}^{obs}) \ge d_0 \end{cases} \tag{6}$$

In Eq. (6), V_{max} is approx. the maximum allowable velocity for the drones, the condition on the sign of $det[d(\mathbf{q},\mathbf{q}^{obs}),\mathbf{q}^{goal}-\mathbf{q}]$ is to make the rotational force go towards the goal. The slight disymetry between the two conditions ≥ 0 and < 0 in Eq. (6) is here to avoid the method from generating a singular stop point at the surface of the obstacle.

Combining the repulsive and rotational forces with the expression of the command vector from Eq.(5) yields a command vector to allow us to keep the desired formation and avoid obstacles at the same time.

$$\mathbf{u}_i = \mathbf{u}_d - C\sum_{j\in N_i}\Omega_{ij} + \mathbf{F}_i^{rep} + \mathbf{F}_i^{rot} \tag{7}$$

E. Summary

In summary, our method requires a choice of 9 tuning parameters in total : k_a,k,δ,C,k_r,k_g,d_0,V_{max},d, where d is the parameter of the formation that represent the minimal distance between two positions. The value of these parameters were chosen after the tests presented in Section IV.C. Finally, the method could be summarized as the following loop, assuming the positions and properties of the obstacles are known at all times:

Algorithm 1 (Main control loop)

- Read user-defined goal position
- Get position of the drones from the sensors
- Compute the formation angle α with Eq. (1), and compute the corresponding formation positions with F from Eq. (2)
- Solve the optimisation problem of Eq. (3), and re-allocate drones to corresponding positions on the formation
- Compute and apply u_i, the set point for the velocity of each drone in the formation, given by Eq. (7)

For safety purposes, the velocity set point u_i is saturated to the maximum allowed velocity V_{max}

IV. Results

The control method described in the previous section was implemented on a small swarm of micro UAVs called CrazyFlie, and different scenarios were run in an indoor confined environment to test the performances of the algorithms.

A. The CrazyFlie UAV

The CrazyFlie is a "small and versatile quad-copter for education and research"* developed by Bitcraze that allows quick and easy testing of guidance, navigation and control algorithm on a real life system. It is a modular platform with a wide ecosystem of add-on boards adding further functionality to the core drone, such as external positioning, onboard image processing, or RGB LEDs for drone shows. For this study, we used an indoor positioning system based on a *Lighthouse Deck* installed on the drones, and a network of Steam VR base stations located around the flight space. These base stations scan the flight area with distinct time-encoded infrared signals, allowing the drone to compute onboard a precise (error<1 cm) location solution without external communications.

Fig. 5 One CrazyFlie used during the experiments

*See at https://www.bitcraze.io/products/crazyflie-2-1/

A simulation environment was developed in order to allow fast iterations on the algorithm development. For this, a representative dynamic model of the closed-loop behavior of the CrazyFlie platform is needed. To get this, the dynamic response to a velocity step command sent to a single CrazyFlie was recorded, and a 2nd order model with delay (see Eq. (8)) was fitted to this response. The data and fitted model can be seen on Fig. 6

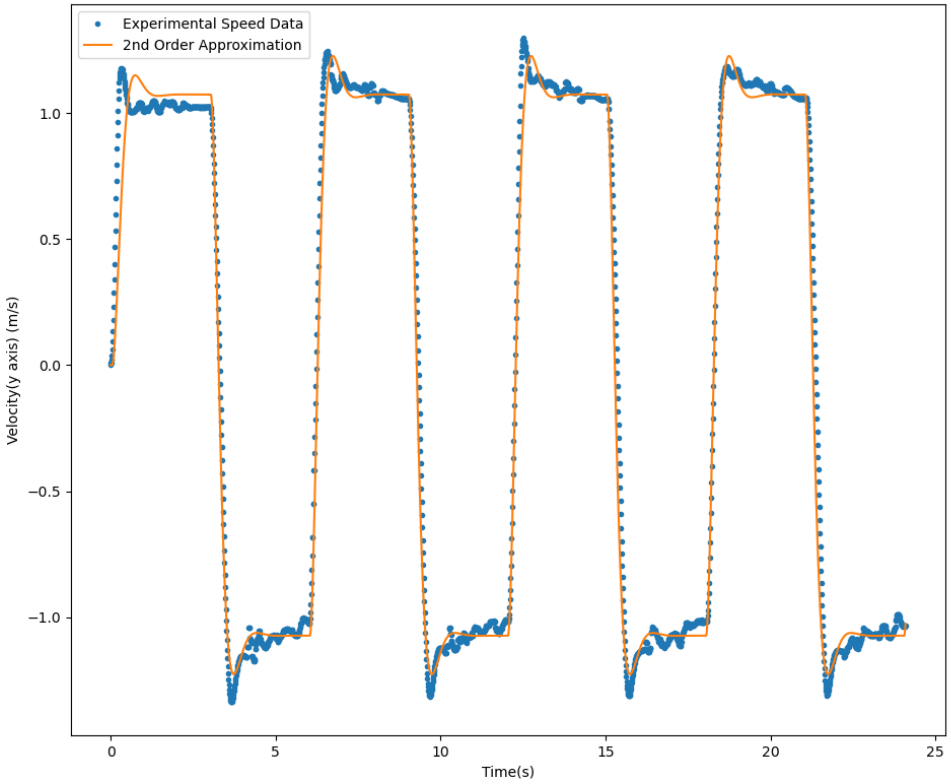

Fig. 6 Experimental data and fitted second order model, for a series of step command in velocity along the lateral body frame (y-axis)

The chosen second-order dynamic model of a single CrazyFlie is:

$$\tau_s^2 \frac{d^2 V_i}{dt^2} + 2\xi\tau_s \frac{dV_i}{dt} + V_i = K_p u(t - \theta_p) \tag{8}$$

with V_i the velocity along one of the axis (x,y or z), u the velocity command, and $(K_p, \tau_s, \xi, \theta_p)$ the fitted parameters. The best fitting parameters for the experimental data we collected were found to be $K_p = 1.07$, $\tau_s = 0.17$ s , $\xi = 0.64$ and θ_p =0.04 s. The CrazyFlie API allows sending position and/or velocity set points to the drones. We send the velocity set point Eq. (7).

B. Parameter values

The swarming algorithms presented in Section III are parameterized through several tuning gains and parameters. The selection of these parameters was done iteratively using the simulation model, then refined using the real-world swarming setup. The parameters used for the reported experiments are summarized in Table 1.

k_a (-)	k (-)	δ (-)	c (-)	k_r (-)	k_g (-)	d_0 (m)	V_{max} (m/s)	d (m)
0.2	1	0.002	0.2	0.02	0.1	1	0.5	0.6

Table 1 Values of tuning parameters

The parameter c corresponds to the matrix $\mathbf{C}$ in Eq. (7) with the expression $\mathbf{C} = c\mathbf{I}$, with $\mathbf{I}$ the identity matrix. Moreover,

the values of these parameters, especially for d_0, V_{max} and d depend on the size of the considered environment. These values were taken for a $6m \times 5m \times 2m$ working space, but a larger working space could benefit from taking different values.

C. Test results

In the presented scenario, we consider a swarm of 7 CrazyFlies in an indoor confined environment initially at rest in one corner of the room, and 3 static targets positioned in each of the remaining corners of the room. In this experiment, one goal is "activated" (meaning the swarm is tasked with reaching this goal), and once the swarm has reached it, another goal (randomly chosen) is activated, and this procedure continues until the swarm receives a signal to land. A large sphere lies in the middle of the room and serves as an obstacle that the drones have to avoid.

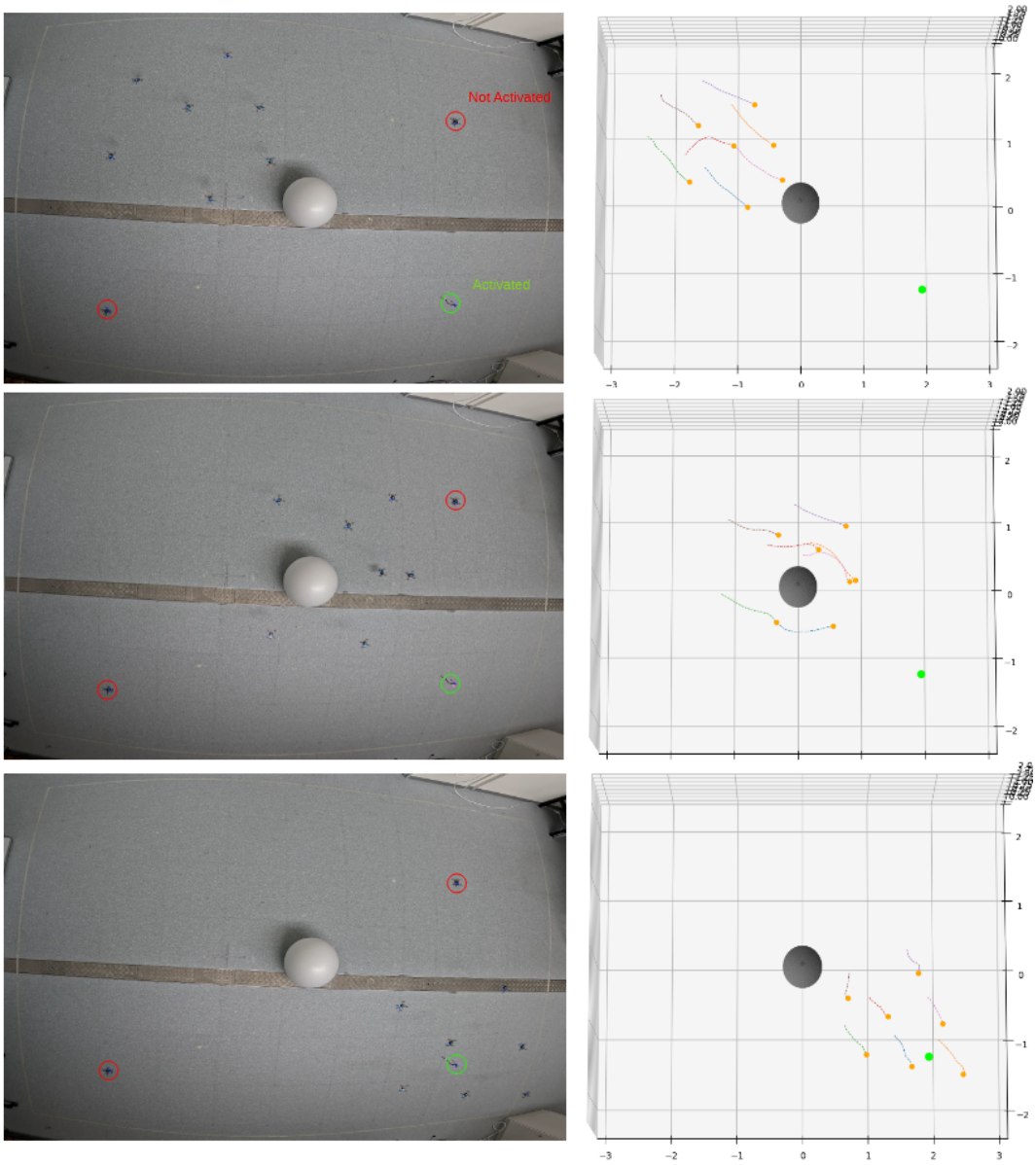

Fig. 7 Beginning of the experiment: the arrow-shaped formation passes an obstacle and reaches the user-defined target

Fig. 7 shows the beginning of the experiment. After a takeoff, the drones go to the first activated goal while traversing

the obstacle in the middle (5 of the drones goes on the left, and 2 on the right of the obstacle). The left side of the figure are the images taken from a camera, and the right side are plots obtained from logs. We can see that the drones manage to avoid the obstacle by turning around it, and reach the target in the desired arrow-shaped formation.

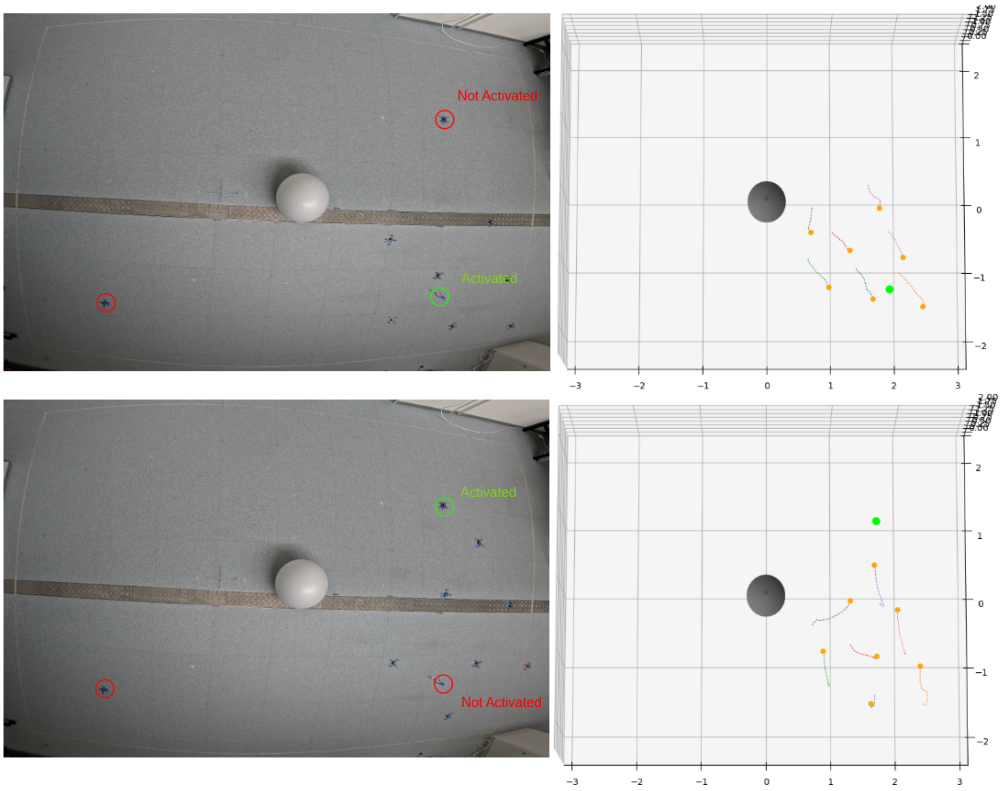

Fig. 8 Reconfiguration of the formation in response to a sudden change of target

Fig. 8 reports the next phase of the experiment, with a sudden target change. In response to this change, Algorithm 1 decides to operate a change in the formation. One can see the reallocation being done, e.g. with one of the side drones becoming the tip of the arrow.

Finally, Figs. 9 to 11 provide some metrics to evaluate the performances of the method. One of the first thing to note is that both the distance between the drones and the distance with the obstacle remain at all times higher than 0.2 m, which considering the size of a Crazyflie (a bit less of 10cm from one end to the other) and the accuracy of our positioning system (<1cm), indicates that no collision ever happened during the experiment. Another thing worth noting is that during obstacle avoidance, the distance between drones is also affected, and drones go closer to each other. This can be explained by the fact that the formation for the drones is not completely rigid, and allows a more fluid-like behavior of the swarm, a natural effect of the trade-off between repulsive and attractive potential in Eq. (7). The relative rigidity is mainly affected by the two parameters δ and c that are used as weights on the local potential function that aims at keeping the formation. This fluid-like behavior can also be witnessed in Fig. 9 where some bumps in the average distance to the formations positions are seen during the phases when the swarm is avoiding the obstacle. Moreover, we can clearly see the two moments in the experiments when the target was changed, creating a sudden change in the formation, leading to spikes in formation position error. Interestingly, after the first target change, the obstacle was not in the way to the new objective, so the time to reach back the new formation shape is really short, whereas after the second change of target an obstacle was indeed in the way, thus causing an increase of the time needed to reach the equilibrium.

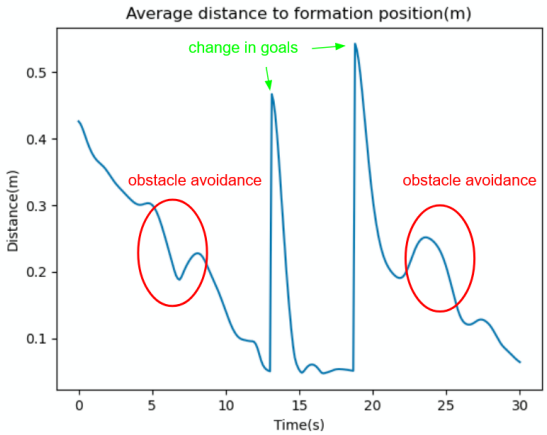

Fig. 9 Average distance to formation position at each time step

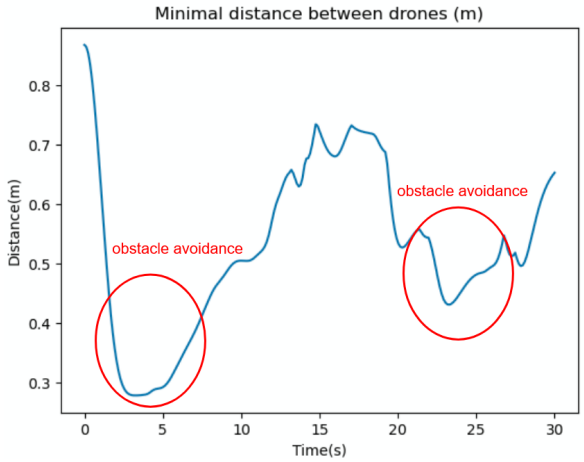

Fig. 10 Minimal distance between drones at each time step

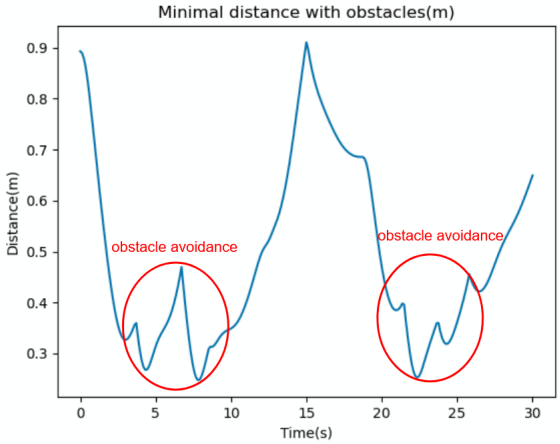

Fig. 11 Minimal distance with the obstacle at each time step

V. Conclusion

In this paper, experimental results were presented for a proposed centralized formation control and obstacle avoidance method able to deal with a medium-sized swarm of micro UAVs flying in a confined environment. The tests were run indoor, using a centralized computer performing the algorithmic calculations, communications and synchronisation between the UAVs. In view of applications, a few leads on what could be done next are as follows.

- *Handle addition or removal of one or several UAVs in the swarm.* Preliminary simulations and experiments show that this can be easily done. Indeed, Algorithm 1 can be readily extended to handle a varying number N of UAVs, as the mathematical formulation of the ILP shows it in Section III.B.
- *Increase the size of the swarm.* The test results we presented were done with a swarm of 7 drones, but a swarm with a higher count of UAVs could be considered. However, we did some test with more drones, and one of the problems was the increase in communication delay, as well as increased oscillations in the swarm because of the confinement (wall-effect). In Section V, an image with a swarm of 20 CrazyFlie can be found, and it is clear that with this amount of drones covering approx. 60% of the room area, maneuverability is quite a challenge.
- *Try a more decentralized/distributed approach.* The centralized nature of our approach can be seen as a weakness, especially in view of applications in scenarios of communication partial or full outage. The main part of our algorithm that needs a centralized approach is the part on position allocation, as the rest of the method on formation control and obstacle avoidance can be easily changed in the case of a distributed approach. We decided to solve a global ILP problem, but this could maybe done differently with a more decentralized or distributed approach.

Appendix : An extended swarm

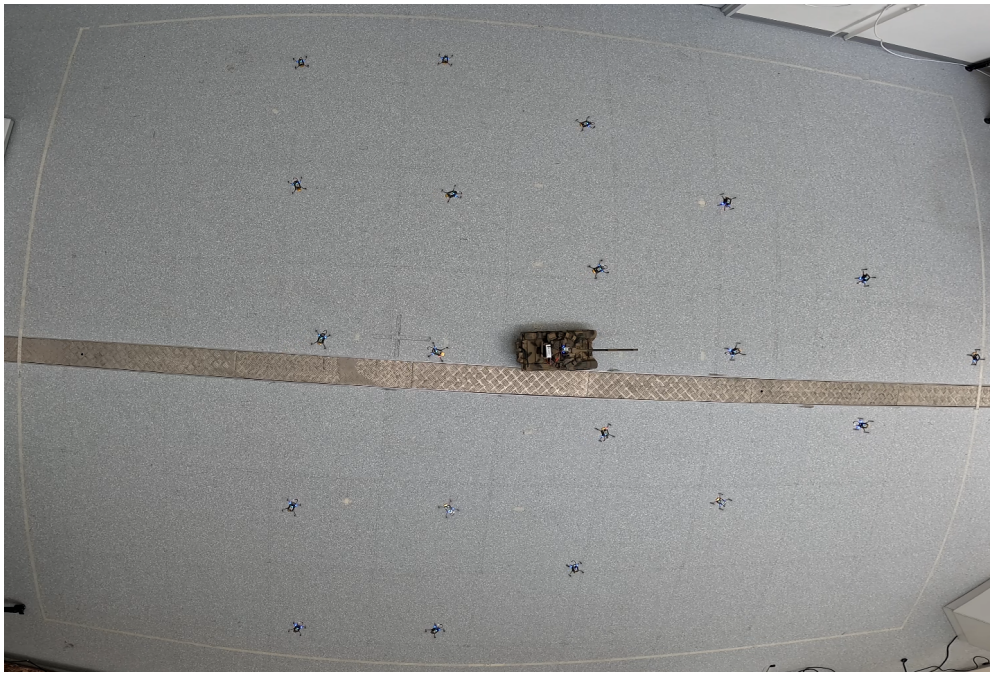

Fig. 12 20 CrazyFlies occupying the majority of the available area. A mobile target is considered.

Appendix : End of the experiment

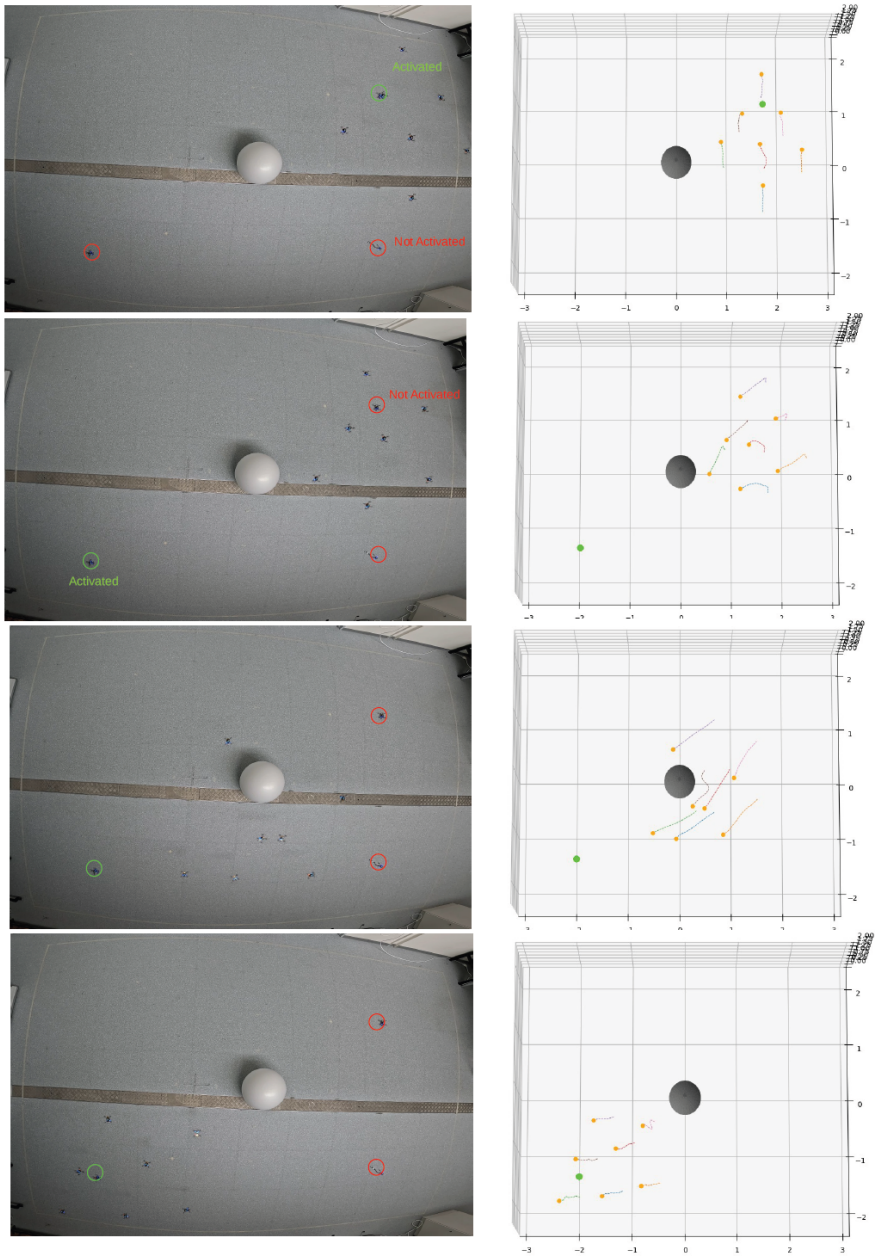

Fig. 13 The final sequence of the experiment

References

[1] Tahir, A., Böling, J., Haghbayan, M.-H., Toivonen, H. T., and Plosila, J., "Swarms of unmanned aerial vehicles—a survey," *Journal of Industrial Information Integration*, Vol. 16, 2019, p. 100106.

[2] Abdelkader, M., Güler, S., Jaleel, H., and Shamma, J. S., "Aerial swarms: Recent applications and challenges," *Current robotics reports*, Vol. 2, 2021, pp. 309–320.

[3] Chung, S.-J., Paranjape, A. A., Dames, P., Shen, S., and Kumar, V., "A survey on aerial swarm robotics," *IEEE Transactions on Robotics*, Vol. 34, No. 4, 2018, pp. 837–855.

[4] Javed, S., Hassan, A., Ahmad, R., Ahmed, W., Ahmed, R., Saadat, A., and Guizani, M., "State-of-the-Art and Future Research Challenges in UAV Swarms," *IEEE Internet of Things Journal*, 2024.

[5] Do, H., Hua, H., Nguyen, M., Nguyen, V.-C., Nguyen, H., and Nga, N., "Formation Control Algorithms for Multiple-UAVs: A Comprehensive Survey," *EAI Endorsed Transactions on Industrial Networks and Intelligent Systems*, Vol. 8, 2021, p. 170230. https://doi.org/10.4108/eai.10-6-2021.170230.

[6] Kamel, M. A., Yu, X., and Zhang, Y., "Formation control and coordination of multiple unmanned ground vehicles in normal and faulty situations: A review," *Annual Reviews in Control*, Vol. 49, 2020, pp. 128–144. https://doi.org/https://doi.org/10.1016/j.arcontrol.2020.02.001, URL https://www.sciencedirect.com/science/article/pii/S1367578820300031.

[7] Kahn, A., Marzat, J., and Piet-Lahanier, H., "Formation flying control via elliptical virtual structure," *2013 10th IEEE INTERNATIONAL CONFERENCE ON NETWORKING, SENSING AND CONTROL (ICNSC)*, 2013, pp. 158–163. https://doi.org/10.1109/ICNSC.2013.6548729.

[8] Do, K., "Formation control of mobile agents using local potential functions," *2006 American Control Conference*, 2006, pp. 6 pp.–. https://doi.org/10.1109/ACC.2006.1656537.

[9] Xu, D., Zhang, X., Zhu, Z., Chen, C., and Yang, P., "Behavior-Based Formation Control of Swarm Robots," *Mathematical Problems in Engineering*, Vol. 2014, 2014, p. 205759. https://doi.org/10.1155/2014/205759, URL https://doi.org/10.1155/2014/205759, publisher: Hindawi Publishing Corporation.

[10] Bunic, M., and Bogdan, S., "Potential Function Based Multi-Agent Formation Control in 3D Space," *IFAC Proceedings Volumes*, Vol. 45, No. 22, 2012, pp. 682–689. https://doi.org/https://doi.org/10.3182/20120905-3-HR-2030.00142, URL https://www.sciencedirect.com/science/article/pii/S1474667016336898.

[11] Anh, D., La, H., Nguyen, T., and Horn, J., "Formation control for autonomous robots with collision and obstacle avoidance using a rotational and repulsive force–based approach," *International Journal of Advanced Robotic Systems*, Vol. 16, 2019, p. 172988141984789. https://doi.org/10.1177/1729881419847897.

[12] Pan, Z., Zhang, C., Xia, Y., Xiong, H., and Shao, X., "An Improved Artificial Potential Field Method for Path Planning and Formation Control of the Multi-UAV Systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, Vol. 69, No. 3, 2022, pp. 1129–1133. https://doi.org/10.1109/TCSII.2021.3112787.

[13] Jadbabaie, A., Lin, J., and Morse, A., "Coordination of Groups of Mobile Autonomous Agents Using Nearest Neighbor Rules," *Automatic Control, IEEE Transactions on*, Vol. 48, 2003, pp. 988 – 1001. https://doi.org/10.1109/TAC.2003.812781.

[14] Rochefort, Y., Piet-Lahanier, H., Bertrand, S., Beauvois, D., and Dumur, D., "Guidance of flocks of vehicles using virtual signposts," *IFAC Proceedings Volumes*, Vol. 44, No. 1, 2011, pp. 5999–6004.

[15] Fu, X., Pan, J., Wang, H., and Gao, X., "A Formation Maintenance and Reconstruction Method of UAV Swarm based on Distributed Control with Obstacle Avoidance," *2019 Australian & New Zealand Control Conference (ANZCC)*, 2019, pp. 205–209. https://doi.org/10.1109/ANZCC47194.2019.8945601.

[16] Eren, U., and Hansen, E., "Collision-Free Decentralised Density Feedback Control for Swarms *∗Authors have contributed equally to this work." *IFAC-PapersOnLine*, Vol. 52, No. 12, 2019, pp. 364–369. https://doi.org/https://doi.org/10.1016/j.ifacol.2019.11.270, URL https://www.sciencedirect.com/science/article/pii/S2405896319312224.

[17] Reynolds, C. W., "Flocks, herds, and schools: a distributed behavioral model," *Seminal graphics: pioneering efforts that shaped the field*, 1987. URL https://api.semanticscholar.org/CorpusID:546350.

[18] Fiorini, P., and Shiller, Z., "Motion planning in dynamic environments using the relative velocity paradigm," *[1993] Proceedings IEEE International Conference on Robotics and Automation*, 1993, pp. 560–565 vol.1. https://doi.org/10.1109/ROBOT.1993.292038.

[19] Van Den Berg, J., Guy, S. J., Lin, M., and Manocha, D., "Reciprocal n-body collision avoidance," *Robotics Research: The 14th International Symposium ISRR*, Springer, 2011, pp. 3–19.

[20] Chakravarthy, A., and Ghose, D., "Generalization of the collision cone approach for motion safety in 3-D environments," *Autonomous Robots*, Vol. 32, No. 3, 2012, pp. 243–266. https://doi.org/10.1007/s10514-011-9270-z, URL https://doi.org/10.1007/s10514-011-9270-z.

[21] Chakravarthy, A., and Ghose, D., "Collision cone-based net capture of a swarm of unmanned aerial vehicles," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 9, 2020, pp. 1688–1710.

[22] Adinandra, S., Schreurs, E., and Nijmeijer, H., "A Practical Model Predictive Control for A Group of Unicycle Mobile Robots," *IFAC Proceedings Volumes*, Vol. 45, No. 17, 2012, pp. 472–477. https://doi.org/https://doi.org/10.3182/20120823-5-NL-3013.00060, URL https://www.sciencedirect.com/science/article/pii/S1474667016314872.

[23] Shim, D. H., and Sastry, S., "A dynamic path generation method for a UAV swarm in the urban environment," *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2008, p. 6836.

[24] Toumieh, C., and Floreano, D., "High-Speed Motion Planning for Aerial Swarms in Unknown and Cluttered Environments," , 2024. _eprint: 2402.19033.

[25] Chang, D. E., and Marsden, J. E., "Gyroscopic Forces and Collision Avoidance with Convex Obstacles," *New Trends in Nonlinear Dynamics and Control and their Applications*, edited by W. Kang, C. Borges, and M. Xiao, Springer Berlin Heidelberg, Berlin, Heidelberg, 2003, pp. 145–159.

[26] Ge, S., and Cui, Y., "New potential functions for mobile robot path planning," *IEEE Transactions on Robotics and Automation*, Vol. 16, No. 5, 2000, pp. 615–620. https://doi.org/10.1109/70.880813.

[27] Chang, D., Shadden, S., Marsden, J., and Olfati-Saber, R., "Collision avoidance for multiple agent systems," *42nd IEEE International Conference on Decision and Control*, Vol. 1, IEEE, 2003, pp. 539–543.