Mixed guidance law for capturing a reactive target by coordinated Multi-UAV

Felipe Kataoka Ishikawa, Sarah Aouiche, Bojan Mavkov, Guillaume Allibert

Mixed guidance law for capturing a reactive target by coordinated Multi-UAV

Felipe Kataoka Ishikawa[1], Sarah Aouiche[2], Bojan Mavkov[1] and Guillaume Allibert[1]

***Abstract*— We present a pursuit law for capturing a moving target with multiple aerial drones in a bounded environment. The objective is to develop strategies to allow a set of drones to perform a mission cooperatively to complete the task. The presented method combines two established pursuit laws: Group Deviated Pure Pursuit (GDP) and Proportional Navigation Guidance (PNG), and the resulting control architecture is in a cascade form. The results were validated in simulation and experimentation on quadrotor aerial drones. The obtained results confirmed the efficacy of the group mixed pursuit strategy, especially in the case of an agile and faster target. A video of the performed experiments can be seen at: https://youtu.be/cRKRUOV-lV4.**

I. Introduction

Over the years, the popularity and accessibility of Unmanned Aerial Vehicles (UAVs), e.g. aerial drones, has greatly increased. These vehicles have become powerful and necessary tools in broad fields of application: industrial, logistical, military, and defense. Civil authorities have faced new challenges as a result, such as the fight against drone flights overprotected areas like airports and prisons.

Preventive solutions have been put in place to overcome this problem, and among these solutions, an interesting one has recently appeared: the use of "anti-drone" drones. Concepts for designing and planning "anti-drone" techniques are presented in [1], [2]. Using a single drone for pursuit presents significant limitations, mainly when the target exhibits chaotic movement or exceeds the drone's speed capabilities.

To address this problem, approaches involving multiple drones for pursuit have been recently deployed. However, deploying cooperative aerial drones requires complex algorithms and prior mission planning. For this purpose, algorithms utilizing multiple robots to pursue an agent have been developed as in [3], [4], [5].

In this context, this work focuses on implementing tracking algorithms applied to quadrotor aerial drones. The objective is to develop strategies to allow a set of drones to perform a mission cooperatively to pursue a reactive target in a controlled environment. The methods studied in this work result from related works focused on pursuit-evasion, formation flight, and anti-drone control, whose intruder is autonomous, more agile, and performs similarly or better than pursuer drones, hence the need for coordination of navigation of the pursuers to make an ambush. This phenomenon can be compared to the behavior observed in a group of predators collaboratively hunting their prey in nature [6].

Our approach aligns with the one presented in [7], which proposes a strategy for cooperative hunting using drones, where the deviated pure pursuit technique governs the pursuit behavior. In [8], a controller system for pursuit using a swarm of drones is developed utilizing Parallel Navigation (PN) in conjunction with a Proportional-Integral-Derivative (PID) controller. In [9], cooperative multi-quadrotor pursuit is achieved by using a path planning algorithm combined with a robust model predictive control to avoid no-fly zones. Recently, machine learning tools like reinforcement learning have also been used for decentralized multi-agent pursuit and applied on drones [10].

In this work, the pursuit strategy between the pursuers and the evader is designed based on the relative kinematic model and geometric, commonly used in missile guidance laws. In other words, pursuer behavior is determined by guidance laws, with parameters adjusted based on the pursuit engagement. We extend this strategy to the quadrotor dynamic model, by applying these group pursuit strategies in robots in real-time. We formulate a behavior-based multi-agent strategy that combines the Group Deviated Pure Pursuit (GDP) from [7] with PN, similar to the approach proposed in [11], [12]. This strategy combines PN for efficient pursuit trajectories with GDP to control the group strategy and to ensure the pursuers maintain the target in sight during the pursuit. This strategy was applied in both simulations and real-time experiments to validate its effectiveness.

II. Problem formulation

The challenge is to devise a strategy for tracking and capturing a reactive target using multiple pursuers in a bounded environment. We consider the pursuit-evasion problem in a horizontal plane (xy-plane). To capture the target in the shortest time possible, the pursuer drones must collaborate to limit the target's movement and reduce their distance from it while avoiding collisions. We consider an agile target with a higher maximal velocity than the pursuers. In addition, the target is reactive and follows an evasion strategy, with its movement in the x-y plane being unknown to the pursuers.

The objective is that at least one of the pursuers intercepts the target. We consider a successful capture when the relative distance between pursuers and target is less than a given threshold (r_{cap}) within a fixed time value (t_{end}).

An illustration of the pursuit-evasion scenario is depicted in Fig. 1. The pursuer i must track the target T while

[1] F. Kataoka Ishikawa, B. Mavkov and G. Allibert are with Université Côte d'Azur, CNRS, I3S, Nice, France, (email:{kataoka;mavkov;allibert}@i3s.unice.fr)

[2] S. Aouiche is with Université Côte d'Azur, IUT, Nice, France (email:sarah.aouiche@etu.univ-cotedazur.fr)

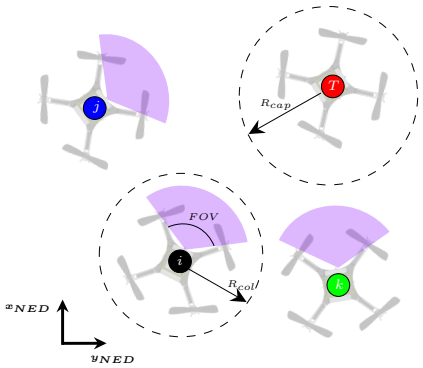

Fig. 1: Pursuit-evasion scenario in the xy-plane.

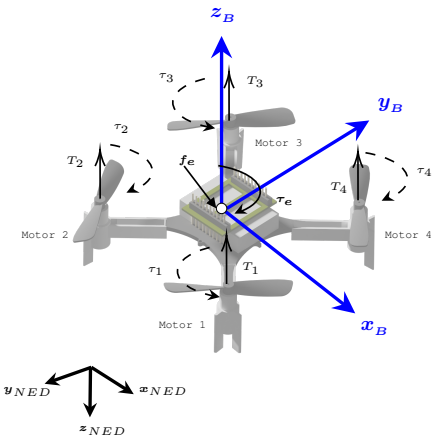

Fig. 2: Free-body diagram of *Crazyflie* quadrotor used in the experiments.

avoiding collision with the other pursuers (j and k). In the experiments and simulations that were conducted, the position of each drone was known. Moreover, the pursuers' position is known by the target. However, the target's position may not always be known in real-world scenarios. Thus, we took these constraints into consideration by assuming that each pursuer drone is equipped with a camera whose optical axis coincides with the x-axis of each drone. Consequently, motion constraints must be considered to ensure that the target remains within the field of view of the pursuers' cameras.

A. Dynamic model

To all quadrotors used, the equations of motion about the center of gravity can be written as follows [13]:

$$\begin{cases} m\ddot{\boldsymbol{r}} = mg\,\boldsymbol{e_3} + \boldsymbol{R}\,(\boldsymbol{f} + \boldsymbol{f_e}) \\ \boldsymbol{J}\dot{\boldsymbol{\omega}} = \mathcal{S}(\boldsymbol{J\omega})\boldsymbol{\omega} + \boldsymbol{R}\,(\boldsymbol{\tau} + \boldsymbol{\tau_e}) \end{cases} \tag{1}$$

where m is the mass of the drone, g is the acceleration due to gravity, $\boldsymbol{r} = [x, y, z]^T$ is the position in the inertial frame, $\boldsymbol{R}$ is the rotation matrix from the body frame to the inertial frame, $\boldsymbol{\omega}$ is the angular velocity of the body, $\boldsymbol{J}$ is the matrix of moments of inertia. The control inputs are given by the thrust force $\boldsymbol{f}$ and the moments $\boldsymbol{\tau}$, while $\boldsymbol{f_e}$ and $\boldsymbol{\tau_e}$ are the external forces and moments, all expressed in the body frame. Additionally, $\mathcal{S}(\cdot)$ represents the antisymmetric matrix operation, and $\boldsymbol{e_3}$ is the unit vector along the vertical axis of the inertial frame. The free-body diagram of a quadrotor is depicted in Fig. 2.

B. Planar engagement

The synthesis of the pursuit laws is based on the relative kinematics between the pursuer and the target. This kinematics equation is deduced from the geometrical engagement of the pursuer-target, as illustrated in Fig. 3.

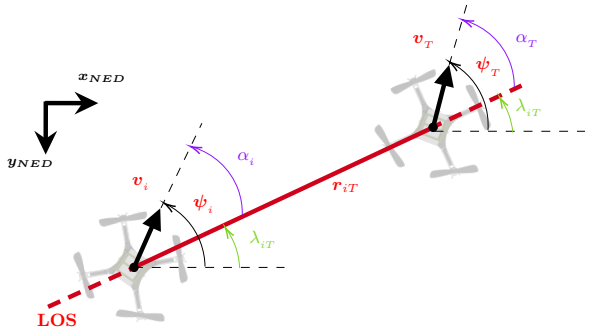

Fig. 3: Geometrical engagement between pursuer and target

In Fig. 3, the straight line ($\boldsymbol{r}_{iT} = \boldsymbol{r}_T - \boldsymbol{r}_i$) is referred as Line-of-Sight (LOS). The bearing angle formed by the LOS and the inertial x-axis is denoted here by λ_{iT}. ψ_i and ψ_T are the heading (yaw angle) of the pursuer i and the target, respectively. α_i and α_T are the deviation angle of the i-th pursuer and the target, respectively. $\boldsymbol{v}_i$ and $\boldsymbol{v}_T$ are the vector velocities of the i-th pursuer and the target, respectively. The relative vector velocity between the target and the i-th pursuer is defined as $\boldsymbol{v}_r = \boldsymbol{v}_T - \boldsymbol{v}_i$.

The relative vector velocity can be projected to the LOS, by decomposing it along parallel ($v_\parallel$) and perpendicular ($v_\perp$) components:

$$\begin{aligned} v_\parallel &= \dot{r}_{iT} = v_T \cos(\alpha_T) - v_i \cos(\alpha_i), \\ v_\perp &= r_{iT}\dot{\lambda}_{iT} = v_T \sin(\alpha_T) - v_i \sin(\alpha_i) \end{aligned} \tag{2}$$

where r_{iT} is the distance between the i-th pursuer and the target, $\alpha_T = \psi_T - \lambda_{iT}$ and $\alpha_i = \psi_i - \lambda_{iT}$.

C. Motion constraints

The main advantage of using multi-agent pursuit is the possibility of cooperative ambushing to intercept a more agile target, as we consider in our case study. The term "agile" refers to possessing superior velocity or having a greater number of degrees of freedom [7].

Quadrotor drones exhibit dynamic constraints due to the coupling of translational and angular velocities. To achieve

translational motion in the x_B direction, the drone must rotate around the y_B axis, and to move in the y_B direction, it must rotate around the x_B axis. In addition to this, we impose specific constraints on the pursuers and the target:

1) Pursuer: We impose additional non-holonomic motion constraints for the pursuer drones, which results as a consequence of the assumption that the target's position is estimated using a camera. Thus, the pursuer displacement is limited in the sensing area, that is, the zone perceived by the onboard camera, as shown in Fig. 1.

We propose a drone control algorithm that will limit the drone's motion by forcing it to act only in turn around the z-axis (ψ) and in the forward velocity (along x_B direction). By consequence, we mainly focus on computing the desired yaw rate ($f_{\dot{\psi}_i}$). We also impose on the pursuers, the desired lateral and altitude velocities to zero.

2) Target: We assume that the target's velocity can be up to twice that of the pursuers ($v_{T,max} = 2\, v_{i,max}$), where $v_{T,max}$ and $v_{i,max}$ are the maximal velocities imposed to the target and the pursuers, respectively. The target is presented using a simple particle model, which gives locomotion advantages.We define a bounded environment called an "arena" to implement the pursuit-evasion strategies. For experimental purposes, we define the arena as a virtual circle with a radius r_{ar}.

Inspired by the target escape strategy proposed by [4], we define the target's reactive velocity as a repulsion model, in which every pursuer implies a repulsive component. Additionally, we consider that the arena bounds also imply a repulsive component to keep the target away from the boundaries. It is defined in terms of the relative distance and decreases proportionally to the distance squared.

The target's reactive velocity is computed as follows [4]:

$$\boldsymbol{v}_T = \sum_{i=1}^{N} \frac{\boldsymbol{r}_{iT}}{\|\boldsymbol{r}_{iT}\|^2} + \frac{\boldsymbol{r}_{T,ar}}{\|\boldsymbol{r}_{T,ar}\|^2} \tag{3}$$

where N is the number of pursuers and $\boldsymbol{r}_{T,ar}$ is the distance vector of the target to the closest point on the arena.

III. Control Architecture

Implementing pursuit-evasion strategies on a non-linear dynamic model of the drone led to using a cascade controller to adapt the pursuit laws to this model. It has a similar composition as the motion control hierarchy used in [12]. Fig. 4 illustrates the implemented cascade control strategy divided into four layers implemented in three levels.

In the high-level control, the *Navigation Guidance* layer refers to the pursuit laws, responsible for generating the desired longitudinal and lateral velocities, yaw rate, and altitude. It is followed by the intermediate-level layer referred to as *Safety constraints*. Here constraints are added to the desired velocities, such as the arena's repulsive effect, maximum velocities, and collision avoidance repulsive effect between agents. The low-level control, represented by *Motion constraints* layer, transforms the safety velocities into desired angles that are the drone's references for the attitude control. The *Attitude and altitude controller* computes the thrust ($\boldsymbol{f}$) and torque inputs ($\boldsymbol{\tau}$) generated by the motors, given the desired angles and altitude.

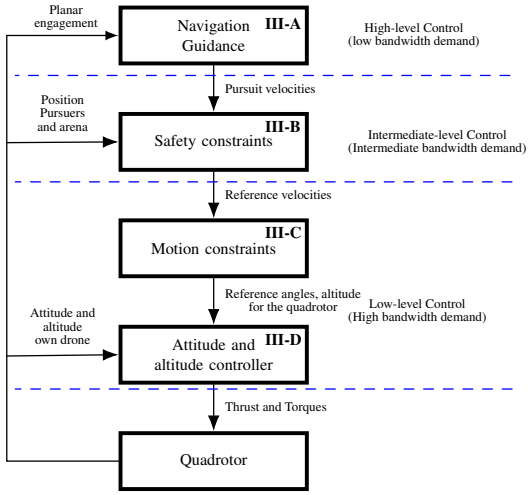

Fig. 4: Control architecture of a quadrotor using guidance laws - Adapted from [12]

A. Navigation and Guidance - Pursuit law

The proposed pursuit law is derived from two established pursuit laws: Proportional Navigation and Group Deviated Pure Pursuit. In the following subsection, we present these methods.

1) Pure pursuit (PP): As stated previously, the PP law is one of the most intuitive guidance laws. The i-th pursuer aims to align its heading, and so its vector velocity, towards the present position of the target ($\psi_i \to \lambda_{iT}$). Therefore, if this perfect alignment is achieved ($\psi_i = \lambda_{iT}$), the relative kinematics equations (2) become:

$$\begin{aligned} \dot{r}_{iT} &= v_T \cos(\alpha_T) - v_i, \\ r_{iT}\dot{\lambda}_{iT} &= v_T \sin(\alpha_T). \end{aligned} \tag{4}$$

To converge towards the target the following condition needs to be satisfied $\dot{r}_{iT} < 0$. This can be achieved only if $v_i > v_T$, for any values of α_T. This condition exposes the limitations when the target is faster than the pursuers.

Finally, an intuitive control law to apply the PP ($\psi_i \to \lambda_{iT}$) is applied as proposed in [11] :

$$f_{\dot{\psi}_i} = -K_{pp}\,(\psi_i - \lambda_{iT}) \tag{5}$$

where K_{pp} defines a positive constant gain.

2) Deviated Pure Pursuit (DPP): The Deviated Pure Pursuit (DPP) is a variant of the PP. Its principle is to lead the pursuer's vector velocity toward an offset angle α_0 relative to the target's current position. This implies that the yaw orientation error is not driven towards zero, but to $\alpha_i = \psi_i - \lambda_{iT} \to \alpha_0$ [11]. Assuming the perfect alignment ($\psi_i - \lambda_{iT} = \alpha_0$), the relative kinematics equations (2) become:

$$\begin{aligned} \dot{r}_{iT} &= v_T \cos(\alpha_T) - v_i \cos(\alpha_0), \\ r_{iT}\,\dot{\lambda}_{iT} &= v_T \sin(\alpha_T) - v_i \sin(\alpha_0). \end{aligned} \tag{6}$$

For the DPP, the condition $\dot{\boldsymbol{r}}_{iT} < 0$ is satisfied if $v_T \cos(\alpha_T) < v_i \cos(\alpha_0)$, . Similar to PP, this pursuit law cannot be guaranteed if $v_T > v_i$.

Finally, an intuitive approach to apply DPP control ($\psi_i \rightarrow \lambda_{iT} + \alpha_0$) is [11]:

$$f_{\dot{\psi}_i} = -K_{dpp}\left(\psi_i - \lambda_{iT} - \alpha_0\right) \tag{7}$$

where K_{dpp} is a positive constant gain.

3) Proportional Navigation Guidance (PNG): Proportional navigation may be the most popular guidance law for interception. Its basic principle is called Parallel Navigation (PN) or constant bearing rule. It consists of keeping a constant bearing angle (λ_{iT}) with respect to the target, and since, in the case of interception, the LOS rate ($\dot{r}_{iT}$) is decreasing, the pursuer and the target are in the collision route.

As described in [11] for the planar case, the parallel navigation rule can be stated as: $\dot{\lambda}_{iT} = 0$ and $\dot{r}_{iT} < 0$. With these assumptions, the relative kinematics equations (2) become:

$$\begin{aligned} v_T \cos(\alpha_T) &= v_i \cos(\alpha_i), \\ v_T \sin(\alpha_T) &< v_i \sin(\alpha_i). \end{aligned} \tag{8}$$

In contrast to PP and DPP, this approach can provide convergence towards the target even for $v_T > v_i$. This will depend on the initial conditions of the engagement.

Finally, the PNG controller can be derived as [11]:

$$f_{\dot{\psi}_i} = K_{png}\dot{\lambda}_{iT} \tag{9}$$

where K_{png} is a positive constant called the navigation gain.

4) Group Deviated Pursuit (GDP): In this work, we seek to extend the pursuit laws to cooperative multi-agent configurations. We use the Group deviated pursuit strategy, developed in [12], that is derived from the DPP law [11].

In this strategy, the idea is that each agent can assume different references of offset angle (7), causing consecutively different trajectories and increasing the possibility of capturing the target. Thus, by considering a variable offset angle of the i-th pursuer (α_i), as a function of the other pursuers, is given by [7]:

$$\alpha_i = \alpha_0 \sum_{j!=i}^{N} \delta_{i,j}\left(\boldsymbol{r}_{ij}, \boldsymbol{r}_{iT}\right) \tag{10}$$

where α_0 is a constant offset angle, $\delta_{i,j}$ is a function indicating the positioning of j-th pursuer in the pursuit frame of the i-th pursuer. The value of $\delta_{i,j}$ is calculated for each neighbor, and it indicates on which side the neighbor j is with $\boldsymbol{r}_{iT}$ (LOS between i-th pursuer and the target). If the j-th pursuer is on the left side, $\delta_{i,j} = 1$, and on the right side $\delta_{i,j} = -1$, and can be defined as:

$$\delta_{i,j}\left(\boldsymbol{r}_{ij}, \boldsymbol{r}_{iT}\right) = \frac{\boldsymbol{r}_{ij} \times \boldsymbol{r}_{iT}}{\left\|\boldsymbol{r}_{ij} \times \boldsymbol{r}_{iT}\right\|} \tag{11}$$

where $\boldsymbol{r}_{ij} = \boldsymbol{r}_{ji} - \boldsymbol{r}_i$ is the distance vector between the pursuers i and j, and $\times$ denotes the cross product.

Similarly to the PNG controller, a controller to implement the GDP can be defined as:

$$f_{\dot{\psi}_i} = -K_{dpp}\left(\psi_i - \lambda_{iT} - \alpha_i\right) \tag{12}$$

where K_{dpp} is a positive constant gain.

5) Group Mixed Pursuit (GMP): As part of the ongoing research on cooperative configuration with drones, we developed a collaborative pursuit law that combines PP with PNG, similar to the approach proposed in [11].

In the PP strategy, pursuers always aim for the target's current location. This strategy leads the pursuer to a "tail chase" pursuit, which is unsuitable against faster targets [12]. When dealing with a moving target, PNG is often used instead. However, PNG has drawbacks, especially when the pursuer is close to the target and the target is capable of quick maneuvers. It can also lead to a loss of sight of the target and requires high bearing angle values when the velocity ratio between the pursuer and the target is low. To address the drawbacks of both methods, a mixed guidance law between GDP and PNG is developed to take advantage of each of them. Since the PN term has a more efficient trajectory, it is preferable to be dominant during the pursuit. The GDP, however, serves two purposes: first, it ambushes, and second, it restricts the pursuers bearing angle to the target, keeping it in sight [12].

To prevent the undesired effect of losing sight of the target, which can occur with PNG, we propose the following modified GDP law [12]:

$$f_{gdp} = K_{dpp}\left(d_{min}\right) \tan\left(\frac{\psi_i - \lambda_{iT} - \alpha_i}{\beta \pi^{-1}}\right) \tag{13}$$

where β represents the FOV angle of the pursuer, and d_{min} is the minimum distance between the pursuers. The $\tan(\cdot)$ function was chosen due to its odd symmetry and its undefined when $\psi_i - \lambda_{iT} - \alpha_i = \beta$. Consequently, this term provides asymptotic values when the heading error approaches the limits of the FOV.

Furthermore, we introduce the gain $K_{dpp}(\cdot)$, which is inversely proportional to the minimum distance between the pursuers, d_{min}, and zero when the distance exceeds a threshold a_0. The gain can be computed as follows:

$$K_{dpp}\left(d_{min}\right) = \begin{cases} K_0 \left(\frac{d_{min} - a_0}{a_0}\right)^2, & \text{if } d_{min} < a_0 \\ 0, & \text{else} \end{cases} \tag{14}$$

where K_0 is a positive gain.

As a result, the GPD term demonstrates close-range performance, which is nullified when the two pursuers are not on a collision course and emphasized by their proximity.

Let us consider the following guidance law, which is composed of two terms, one for the PNG (f_{png}) and the other for the GDP (f_{gdp}), we will call it Group mixed pursuit (GMP). Combining (9) and (13), the GMP law can be written as:

$$f_{\dot{\psi}_i} = \sigma_{png}\left(f_{png}\right) + \sigma_{gdp}\left(f_{gdp}\right) \tag{15}$$

where $\sigma_a(\cdot)$ is a saturation function. Therefore, the σ_{gdp} threshold must be bigger than σ_{png} to allow the second term to overcome the first.

B. Safety constraints

The safety layer is located in the intermediate-level control, and it is used to impose safety constraints in terms of maximum velocities, the arena bounds, and collision avoidance between agents.

In [7], the longitudinal control input of pursuers varies with respect to the magnitude of the desired vector velocity $\boldsymbol{v}_i$ of the pursuer, and it is also related to the target and neighbor distances. Thus, the longitudinal control input is defined as:

$$f_{v_i} = v_{i,\max}\sigma_a\left(\frac{r_{iT} - r_{cap}}{r_{cap}}\right)[C_{rep}, C_{ar,i}]_{min} \tag{16}$$

where r_{cap} is the radius to capture the target, C_{rep} is the safety coefficient that regulates the velocity of the pursuer i when approaching the other pursuers, and $C_{ar,i}$ is the safety coefficient that regulates the velocity when the i-th pursuer is close to the limits of the arena. The values of C_{rep} and $C_{ar,i}$ are limited in the range of $[0, 1]$.

The term C_{rep} represents the collision avoidance and acts as brake coefficient, limiting the velocity input (f_{v_i}) to zero [7]. C_{rep} is defined as:

$$C_{rep} = \left[C_{rep_{i,j}}, \ldots, C_{rep_{i,N}}\right]_{\min} \tag{17}$$

The coefficient of repulsion takes the minimum value of the individual repulsion from i -th pursuer to all neighbors inside the repulsion area, and it is calculated as [7]:

$$C_{rep_{i,j}} = \begin{cases} 1, & \text{if } r_{ij} < r_{col} + a_0 \\ \frac{r_{ij} - a_0}{r_{col}}, & \text{otherwise} \end{cases} \tag{18}$$

where r_{col} is the radius of collision between pursuers and a_0 is constant offset, satisfying $a_0 < r_{col}$.

An additional term $C_{ar,i}$ is also added to generate repulsion when the pursuer is getting closer to the border of the arena. It is computed similarly to $C_{rep_{i,j}}$, but instead of r_{ij}, we use $r_{i,ar}$, which represents the distance of the pursuer i to the closest point on the arena border.

C. Motion constraints

The *Motion constraints* level concentrates on transforming the reference velocities input from *Safety layer*, into commands compatible with the quadrotor, which in this case are the desired angles: ϕ_d, θ_d and ψ_d. Applying the non-holonomic constraint, as described in Sect. II-C.1, the quadrotor states in (1) will have the following tendency:

$$v_i \to f_{v_i}, \quad u_i \to f_{u_i}, \quad f_{\dot{\psi}_i} \to f_{\psi_i} \tag{19}$$

The non-holonomic constraints rely on velocity control, implying a study on the lateral and the longitudinal dynamics. The two can be simplified by linearizing the quadrotor dynamic equation (1). As shown in [14], the altitude dynamics can be represented by two integrators in cascade and the longitudinal/lateral dynamics by four integrators. This simplification is justified by considering small values for the angles/orientation. Then, a control law to assure the condition (19) can be written as:

$$\begin{aligned} \theta_d &= K_\theta\left(f_{vi} - v_i\right), \\ \phi_d &= K_\phi\left(f_{ui} - u_i\right), \\ \psi_d &= \int_0^t f_{\dot{\psi}_i} dt, \end{aligned} \tag{20}$$

where K_θ, K_ϕ are positive gains.

D. Attitude and Altitude Control

For attitude control, three Proportional-Integral-Derivative (PID) controllers were designed for each angle: *roll*, *pitch*, and *yaw*. The goal is to transform the desired angles into torque ($\boldsymbol{\tau}$) applied to the rotational dynamics. A PID controller is also designed to obtain the thrust ($\boldsymbol{f}$) for altitude control. The PID gains tuning is designed with the quadcopter's simplified model, which can be simplified close to the hovering state by linearizing the relationship between rotation velocity and the derivatives of the Euler angles as shown in [14] and [15].

The PID controller for the torque is calculated as follows:

$$\tau_x = k_{p,\phi}\left(\phi_d - \phi\right) + k_{d,\phi}\left(\dot{\phi}_d - \dot{\phi}\right) + k_{i,\phi}\int_0^t \left(\phi_d - \phi\right) dt \tag{21}$$

where τ_x is the generated torque along x-axis, $k_{p,\phi}$, $k_{d,\phi}$ and $k_{i,\phi}$ are the proportional, derivative and integral gains of the roll angle controller, respectively, ϕ_d is the desired roll angle and $\dot{\phi}_d$ is the desired roll rate. The other PID controllers for *pitch*, *yaw*, and altitude have the same structure.

IV. Implementation and Results

For the experimental implementation and validation of the GMP pursuit-evasion strategy, we consider a scenario with three pursuers and a single target moving in a horizontal x-y plane. The drones used in this work are the ***Crazyflie®** 2.1* quadrotor. In the proposed scenario, we assume that the target can move up to two times faster than the pursuers ($v_{T,max} = 2v_{i,max}$). The arena is considered as a circle with a radius ($r_{ar} = 1.2$ m). The maximal episode time in the experiments is set to $t_{end} = 120s$, for which the drones should capture the target. If this timeframe is exceeded, the capture is considered unsuccessful. In the simulations we vary these parameters, including the number of pursuers, to analyze the impact on the capture performance for each of these parameters. The initial conditions are random for each episode, respecting the following conditions:

1) The initial relative distance between pursuers and the target is greater than r_{cap};
2) The initial relative distance between the pursuers (e.g. r_{ij}) is greater than a_0.

The simulation environment was developed in MATLAB/Simulink®, by using the discretized nonlinear dynamic equation (1) with a sampling period of $1\ ms$ and a Runge-Kutta solver. The numerical values to simulate the drones were taken according to the physical parameters

of *Crazyflie 2.1* that are estimated in [16]. In Table I, the constant parameters used in the pursuit law are given.

TABLE I: Pursuit law parameters

Parameter	Value	Parameter	Value
α_0	$22.5°$	r_{col}	$0.2\ m$
β (FOV)	$180°$	a_0	$0.1\ m$
r_{cap}	$0.2\ m$	$v_{T,\,max}$	$1\ m \cdot s^{-1}$

A. Simulation

First, the simulation results for the case of three pursuers and a single target are going to be presented here. The parameters used in this scenario correspond to the experimental setup that we used to implement the proposed method.

Fig. 5 shows the relative distance between pursuers and the target as a function of time. In this configuration, the three pursuers come close to the target at various points. However, as the target possesses superior velocity capabilities and has information about its position, in some instances it manages to avoid the pursuers. Finally, the pursuers capture the target by reaching the threshold distance at 22 s.

Fig. 6 shows the displacement of the drones for different timestamps for this simulation. In this figure, the circle (red) represents the path of the target, and the path of the pursuers is illustrated by the asterisk (black), the square (blue), and the diamond (green). The arrows denote the actual heading (yaw) of each drone. It can be observed that both the target and pursuers stay inside the arena and keep a safe distance, showing the influence of the safety constraints we impose in the control law. It is also important to highlight that the capture time varies depending on the initial conditions (positions of pursuers and target in the inertial frame).

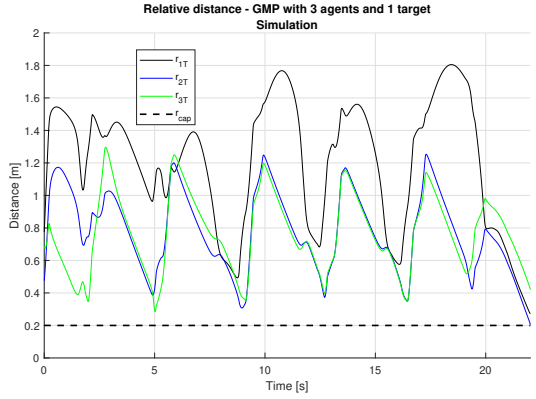

Fig. 5: Distance between pursuers and target - Simulation

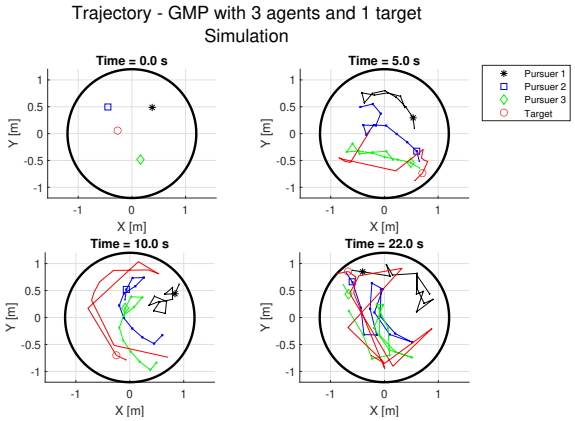

Fig. 6: Trajectories generated by three pursuers and a reactive target - Simulation

We analysed the capture performance by varying the following parameters: the number of pursuers and the arena's size, while the other pursuit law parameters have not been changed. For each varying parameter, we averaged the results over 10 episodes.

Effect of the number of pursuers: The average time to capture the reactive target concerning the number of pursuers is shown in Table II. With an increasing number of pursuers, the average time to capture decreases exponentially. We have noticed that by increasing the number of pursuers, the target tends to move more often closer to the boundaries of the arena to avoid them.

TABLE II: Effects of the number of pursuers in the pursuit

Number of pursuer	Time to capture [s]
3	17.25
5	10.72
8	5.69

Effect of Arena's size: The average time to capture the reactive target concerning the arena's radius in pursuit is shown in Table III. In this scenario, the number of pursuers is fixed at three. From these results, we can conclude that the average time to capture increases linearly with the arena size. The results appear logical since the target has more space to execute evasive maneuvers.

TABLE III: Effects of the arena's size in the pursuit

Arena radius [m]	Time to capture [s]
1	15.67
3	25.25
5	56.17

B. Experimentation

We implemented the proposed pursuit strategies in a real setup using Crazyflie 2.1 quadrotors in a bounded environment, controlled by a ground station and a radio transmitter. The real-world experiments took place in an indoor flight arena equipped with a positioning system, which tracked the pose of all agents. We used the Lighthouse system [17], an optically-based positioning system that allows the Crazyflie

to calculate the position for each drone with an accuracy better than a decimeter precision. The control for each pursuer was calculated on a local computer and transmitted via radio to the drones at 20 Hz.

Fig. 7 and 8 show the results in the real setup for the same scenario used in the simulations with different initial conditions. Similarly to the simulation results in Fig. 5, the three pursuers arrive to come close to the target at various points. In this configuration, two of the pursuers approach the target at around 10 s and finally the pursuer 3 captures the target by reaching the threshold distance at 10.3 s. The video of the experiment is provided on the following link: https://youtu.be/cRKRUOV-lV4.

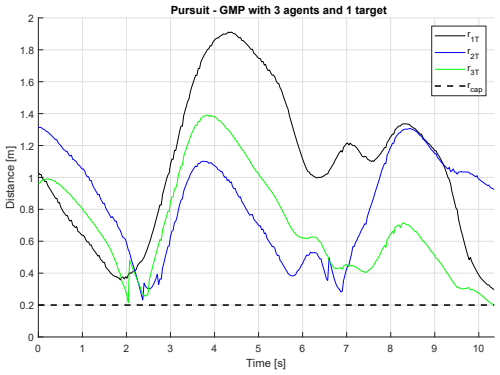

Fig. 7: Distance between pursuers and target - Experimentation

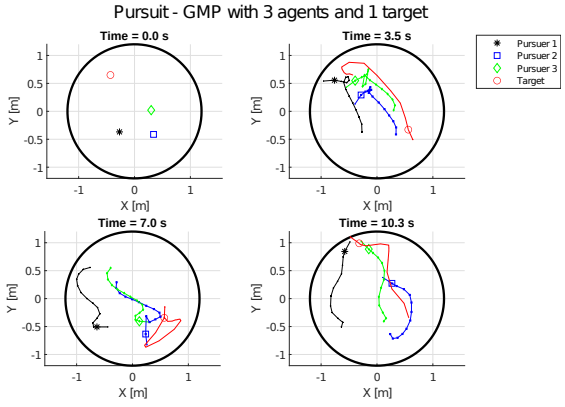

Fig. 8: Trajectories generated by three pursuers and a reactive target - Experimentation

V. Conclusion and Future Works

In this work, a successful implementation of mixed guidance law for pursuit was presented. The results were validated in simulation and real experimental setup, and they confirmed the efficacy of the group mixed pursuit strategy, especially in the case of an agile and faster target.

Interesting perspectives remain to be explored. Naturally, these results should be extended to the 3D pursuit case, which will be done in all three dimensions. Moreover, to improve the performance of the proposed algorithm, more advanced control is planned to be applied, notably using machine learning approaches that combine reinforcement learning with the classical control approaches, such as the one presented here. To adjust the experimental setup to more realistic outdoor scenarios, in future implementations, the drones should be equipped with real cameras that will be used to locate the target.

Acknowledgements

This work has been supported by the French government, through the France 2030 investment plan managed by the Agence Nationale de la Recherche, as part of the "UCA DS4H" project, reference ANR-17-EURE-0004 and by the French National Research Agency through the ANR project DEVIN ANR-23-IAS2-0001.

References

[1] T. Lefebvre and T. Dubot, "Conceptual design study of an anti-drone drone," in *16th AIAA Aviation Technology, Integration, and Operations Conference*, p. 3449, 2016.

[2] S. Park, H. T. Kim, S. Lee, H. Joo, and H. Kim, "Survey on anti-drone systems: Components, designs, and challenges," *IEEE Access*, vol. 9, pp. 42635–42659, 2021.

[3] A. Kamimura and T. Ohira, *Group Chase and Escape: Fusion of Pursuits-Escapes and Collective Motions*. Springer Nature, 2019.

[4] J. Li, M. Li, Y. Li, L. Dou, and Z. Wang, "Coordinated multi-robot target hunting based on extended cooperative game," in *2015 IEEE International Conference on Information and Automation*, (Lijiang, China), pp. 216–221, IEEE, 2015.

[5] F. Belkhouche, B. Belkhouche, and P. Rastgoufard, "Parallel navigation for reaching a moving goal by a mobile robot," *Robotica*, vol. 25, no. 1, pp. 63–74, 2007.

[6] S. T. Fabian, M. E. Sumner, T. J. Wardill, S. Rossoni, and P. T. Gonzalez-Bellido, "Interception by two predatory fly species is explained by a proportional navigation feedback controller," *Journal of The Royal Society Interface*, vol. 15, no. 147, p. 20180466, 2018.

[7] C. de Souza, P. Castillo, and B. Vidolov, "Local interaction and navigation guidance for hunters drones: a chase behavior approach with real-time tests," *Robotica*, vol. 40, no. 8, pp. 2697–2715, 2022.

[8] A. Reinard Rahardian, Y. Y. Nazaruddin, V. Nadhira, and S. Bandong, "Implementation of parallel navigation and pid controller for drone swarm pursuit," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 2513–2518, 2023. 22nd IFAC World Congress.

[9] A. Pierson, A. Ataei, I. C. Paschalidis, and M. Schwager, "Cooperative multi-quadrotor pursuit of an evader in an environment with no-fly zones," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 320–326, 2016.

[10] C. de Souza, R. Newbury, A. Cosgun, P. Castillo, B. Vidolov, and D. Kulić, "Decentralized multi-agent pursuit using deep reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 6, no. 3, pp. 4552–4559, 2021.

[11] N. A. Shneydor, *Missile guidance and pursuit: kinematics, dynamics and control*. Elsevier, 1998.

[12] C. De Souza, *Hunter drones : drones cooperation for tracking an intruder*. Phd thesis, Université de Technologie de Compiègne, 2021.

[13] T. Tomić, C. Ott, and S. Haddadin, "External wrench estimation, collision detection, and reflex reaction for flying robots," *IEEE Transactions on Robotics*, vol. 33, no. 6, pp. 1467–1482, 2017.

[14] J. Li and Y. Li, "Dynamic analysis and pid control for a quadrotor," in *2011 IEEE International Conference on Mechatronics and Automation*, pp. 573–578, 2011.

[15] G. Qingji, Y. Fengfa, and H. Dandan, "Research of precision flight control for quadrotor uav," in *Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference*, pp. 2369–2374, 2014.

[16] J. Forster, *System Identification of the Crazyflie 2.0 Nano Quadrocopter*. Master thesis, Swiss Federal Institute of Technology (ETH) Zurich, 2015.

[17] A. Taffanel, B. Rousselot, J. Danielsson, K. McGuire, K. Richardsson, M. Eliasson, T. Antonsson, and W. Hönig, "Lighthouse positioning system: Dataset, accuracy, and precision for uav research," in *2021 IEEE International Conference on Robotics and Automation - Real World Swarms Workshop*, 2021.