ME 7374 Term Project, Fall 2024

Multi-Agent Target Localization Under External Disturbance With Obstacle Avoidance

Milad Alipour Shahraki and Mohammad Hussein Yoosefian Nooshabadi

***Abstract*— In this project, we consider the problem of multi-agent target localization and tracking. The localization performance is quantized using the well-known Fisher Information Matrix (FIM), using which the proper configuration of agents with respect to the target is derived to obtain *optimal* localization. Next, the problem is addressed in two different settings: agents operating under external disturbances and agents navigating through obstacle-rich environments. In the disturbance-prone scenario, an Extended Kalman Filter (EKF) is integrated with a Lyapunov-based feedback controller to estimate and reject disturbances, improving trajectory tracking and localization accuracy. In the obstacle-rich scenario, Control Barrier Functions (CBFs) are employed as a safety filter on the Lyapunov-based controller to guarantee collision-free navigation while maintaining the target-tracking objective. Simulation results confirm the effectiveness of the proposed methods, achieving accurate localization, robust disturbance rejection, and safe navigation.**

I. Introduction

A. Motivation and Background

In dynamic and uncertain environments, the ability of autonomous agents to perform accurate and reliable target localization is often restricted by external disturbances and the presence of obstacles. In scenarios where agents must track a moving target, disturbances such as environmental forces or system noise can significantly degrade their performance, leading to errors in localization. Rejecting these disturbances is crucial to ensuring robust target tracking and coordination among agents, even in the absence of obstacles. On the other hand, in environments where agents are undisturbed but face static or dynamic obstacles, ensuring collision-free navigation becomes the primary challenge. Safe maneuvering while maintaining effective target localization requires safe control strategies that prioritize obstacle/collision avoidance without compromising the system's target tracking objective. These two scenarios—disturbed dynamics without obstacles and undisturbed dynamics with collision avoidance—highlight two distinct challenges for improving the reliability and safety of multi-agent target localization.

B. Literature Review

In recent years, the problem of multi-agent target localization and tracking has found various applications, such as surveillance [1], search-and-rescue [2], information acquisition [3], and military operations [4]. These applications demonstrate the importance of multi-agent target localization and tracking in dynamic and uncertain environments.

External disturbances, such as environmental forces and noise, pose significant challenges to the performance of multi-agent systems, particularly in localization and tracking. In case of bounded external disturbance, robust control methods can be used to mitigate the disturbance effects usually considering the worst-case scenarios [5]. Further, adaptive control methods can also be used to reduce disturbance effects by designing adaptive control gains that can adapt to the disturbance [6]. The main drawback of these robust approaches is that they are overly conservative since they consider the worst-case scenario. Also, they assume that the disturbance is upper-bounded and the bound is known. To overcome these issues, disturbance observers can be used to estimate the disturbance, which then can be taken into account in the control algorithm. Various variants of the Extended Kalman Filter have been widely used for this purpose [7]. In this project, the integration of EKF-based disturbance observers with Lyapunov-based feedback controllers builds upon these approaches to achieve robust localization under external disturbances.

Another major challenge in multi-agent system settings is the presence of obstacles in the environment and the possibility of inter-agent collisions. These challenges can jeopardize safety even in disturbance-free environments. The problem of obstacle avoidance is naturally a non-convex optimization problem. In addition to the trivial nonlinear programming method, other approaches such as mixed-integer programming [8], and artificial potential functions [9] can be used. However, these methods are generally computationally expensive. More recently, the concept of Control Barrier Functions (CBFs) was introduced [10], which uses quadratic programming (QP) to provide safety guarantees for safety-critical systems, including obstacle avoidance applications [11]. In this project, CBFs are used to design a QP-based safety filter for modifying the unsafe controller to avoid obstacles of different radii and inter-agent collisions.

C. Contributions

The contributions of this term project are two-fold:

1) *Disturbance Estimation and Rejection for Improved Localization:* In the first scenario, a novel approach is proposed to address external environmental disturbances and noise affecting multi-agent target localization. An Extended Kalman Filter (EKF) is designed to estimate

{alipourshahraki.m, yoosefiannooshabad.m}@northeastern.edu

the disturbances, which is then incorporated into a Lyapunov-based feedback controller. This integration reduces trajectory tracking errors and ensures robust disturbance rejection, enabling agents to maintain accurate localization even in challenging environments with minimal reliance on external communication.

2) *Collision-Free Navigation with Control Barrier Functions (CBFs):* In the second scenario, where the environment is free from disturbances, a Control Barrier Function (CBF)-based safety filter is developed and applied. The CBF ensures that agents avoid static obstacles and prevent collisions with one another during their trajectories. By dynamically modifying control inputs, the safety filter allows agents to adhere to their planned optimal trajectories while maintaining safety, demonstrating effective navigation in constrained environments.

The rest of this report is organized as follows: Section II describes the problem. In Section III, the main results are presented. Section IV summarizes the simulation and results and Section V, concludes the report and points future directions.

II. Problem Statement

This project focuses on the problem of multi-agent target localization in dynamic environments. Consider the following disturbed dynamics for each of the agents

$$\begin{aligned} \dot{p}_a &= R \begin{bmatrix} v \\ 0 \end{bmatrix} + Cd, \\ \dot{R} &= R\omega^{\times}, \end{aligned} \tag{1}$$

where $p_a \in \mathbb{R}^2$ is the position of the agent, $R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$ is the rotation matrix describing the agent's orientation, $v, \omega \in \mathbb{R}$ are linear and angular velocities, respectively, and $\omega^{\times}$ is the skew-symmetric matrix defined as $\omega^{\times} = \begin{bmatrix} 0 & -\omega \\ \omega & 0 \end{bmatrix}$. Moreover, $d \in \mathbb{R}^q$ is the external disturbance and $C \in \mathbb{R}^{2\times q}$ is a known matrix. We consider the following linear dynamics for the disturbance

$$\dot{d} = Ad, \tag{2}$$

with the known matrix $A \in \mathbb{R}^{q\times q}$ [7].

The goal is to estimate the states (i.e., position and velocity) of a target using an EKF. Since the dynamics of the target is unknown, we use the common kinematic model for an object in 2D

$$\begin{bmatrix} \dot{p}_t \\ \dot{v}_t \end{bmatrix} = \begin{bmatrix} 0_2 & I_2 \\ 0_2 & 0_2 \end{bmatrix} \begin{bmatrix} p_t \\ v_t \end{bmatrix} + \mu, \tag{3}$$

where $p_t, v_t \in \mathbb{R}^2$ are position and velocity of the target and $\mu \sim \mathcal{N}(0, Q)$ is zero-mean Gaussian noise with covariance matrix Q. (3) is a quasi-steady model, meaning that it is assumed that the velocity of the target is slowly changing. This is a common assumption in the literature [12].

In this project, we consider two scenarios that are described in the following subsections.

A. Disturbed agent dynamics without obstacle avoidance

In the first scenario, agents operate in disturbance-prone conditions where external disturbances affect their ability to track the target accurately (as modeled in (1)). An EKF is designed for estimating the disturbance, which is then fed to a Lyapunov-based feedback controller to reduce trajectory tracking errors and reject disturbance effects. Each agent relies solely on information shared with its neighboring agents, and a decentralized information filter is employed for robust estimation of the target's states [13]. In this scenario, we do not consider any obstacles.

B. Undisturbed agent dynamics with collision/obstacle avoidance

In the second scenario, the environment is free of disturbances (i.e., set $d = 0$ in (1)) and agents have access to their true position and orientation. However, agents must navigate safely to avoid static obstacles and prevent collisions with one another. Note that since the goal in this section is to design a guaranteed obstacle avoidance algorithm, it is crucial to assume that the dynamics of the agents are deterministic. After the trajectories are planned and the Lyapunov-based feedback controller is designed to achieve trajectory tracking, a control barrier function (CBF)-based safety filter is applied to ensure collision-free navigation [10]. The CBF modifies the control inputs as needed, allowing agents to maintain safe paths while adhering to their optimal target tracking objectives.

Together, these two scenarios address complementary challenges in multi-agent systems, providing a robust framework for accurate localization and safe operation in dynamic environments.

III. Main Results

In this section, we first derive the configuration of agents with respect to the target that results in optimal state estimation. Then, we see the algorithm used for trajectory generation for individual agents. Next, we see the information EKF used for target localization. Then, the controller and EKF designs for the problem described in Section II-A and the controller and CBF designs for the problem defined in Section II-B are separately discussed.

Throughout this sequel, agents are assumed to be equipped with range-only sensors modeled as

$$z = \underbrace{\|p_t - p_a\|}_{h_r(x)} + n_r, \tag{4}$$

where p_t and $p_a \in \mathbb{R}^2$ are the positions of the target and the agent, respectively, and $n_r \sim \mathcal{N}(0, \sigma^2)$ is zero-mean Gaussian noise with variance σ^2.

a) Optimal configuration of agents: The goal is for the agents to *optimally estimate* the states of the target. The estimation performance is quantified using the D-optimality criterion, defined as the determinant of the Fisher Information Matrix (FIM) [14]. The FIM of a single measurement z

of the variable x is defined as [15, §2]

$$\text{FIM} = \mathbb{E}\left[\left(\frac{\partial \log p(x;z)}{\partial z}\right)\left(\frac{\partial \log p(x;z)}{\partial z}\right)^\top\right], \quad (5)$$

where $p(x;z)$ is the conditional probability density function (pdf). The FIM of n_a agents (i.e., measurements) is the sum of all FIMs.

Substituting (4) into (5) and using the pdf of the Gaussian noise in (4) one can obtain

$$\text{FIM} = \sum_{k=1}^{n_a} \frac{1}{\sigma^2}\begin{bmatrix} \frac{(x_t-x_{a_k})^2}{\delta_k^2} & \frac{(x_t-x_{a_k})(y_t-y_{a_k})}{\delta_k^2} \\ \frac{(x_t-x_{a_k})(y_t-y_{a_k})}{\delta_k^2} & \frac{(y_t-y_{a_k})^2}{\delta_k^2} \end{bmatrix}, \quad (6)$$

where δ_k is the distance from the k^{th} agent to the target, $p_t = \begin{bmatrix} x_t & y_t \end{bmatrix}^\top$ is the position of the target and $p_{a_k} = \begin{bmatrix} x_{a_k} & y_{a_k} \end{bmatrix}^\top$ is the position of the k^{th} agent. Taking the determinant of the FIM in (6) gives

$$\det(\text{FIM}) = \frac{1}{\sigma^4} \sum_{1\le i<j\le n_a} \sin^2 \beta_{ij}, \quad (7)$$

where β_{ij} is the angle between the target and i^{th} and j^{th} agent (or in the case of a single agent, the angle between two different measurements). The goal is to maximize the expression in (7). As discussed in [16], we have the following results.

- $n_a = 2$ *agents:* To obtain optimal estimation, the agents should be perpendicular to each other with respect to the target at all times. In other words $\beta_{12} = \frac{\pi}{2}$.
- $n_a \ge 3$ *agents:* To obtain optimal estimation, the agents should be uniformly distributed on a circle around the target. In other words, at all times we should have $\beta_{ij} = \frac{2\pi}{n_a}$.

As a result, we should generate trajectories for each agent such that the above-mentioned optimal configurations hold at all times.

b) Trajectory generation for each agent: In order to keep the configuration derived above, inspired by the concept of Moving Path Following (MPF) [17], we design the desired trajectory as

$$p_d(t) = r(\gamma(t)) + \hat{p_t}(t), \quad (8)$$

where $\hat{p_t}(t)$ is the estimate of the target's position, obtained using an information EKF, as described in the next section. Also, $r(\gamma(t))$ is defined as

$$r(\gamma) = \begin{bmatrix} r_x \cos(\gamma+\gamma_0) & r_y \sin(\gamma+\gamma_0) \end{bmatrix}^\top, \quad (9)$$

where r_x and r_y are two constants determining the distance of the agent from the target and γ is the *phase* of the agent and will be illustrated in what follows. Using (8) and (9), it is easy to see that if $p_a \to p_d$ then $\|p_a - \hat{p_t}\| \to \|r(\gamma)\| \le (r_x^2 + r_y^2)^{\frac{1}{2}}$, which means that the agents will follow the target with the maximum distance of $(r_x^2 + r_y^2)^{\frac{1}{2}}$. As a result, to track the target, we just need to design a controller that guarantees tracking the desired trajectory defined in (8). Also, note that γ is the angle each agent make with the horizontal line passing through the target. So, to respect the optimal configuration described previously, we should have $|\gamma_{0_1} - \gamma_{0_2}| = \frac{\pi}{2}$ for $n_a = 2$ agents and $\left|\gamma_{0_i} - \gamma_{0_j}\right| = \frac{2\pi}{n_m}$ for $n_a \ge 3$ agents. Also, as we will see, the time-derivative of γ is the angular velocity of the agents. Agents will try to reach a consensus about their angular velocity. In other words, all agents will encircle the target with a given desired angular velocity, ω_d. This will become more clear in the following sections.

c) Information EKF for target estimation: Each agent, estimates the target's state (i.e., $x = \begin{bmatrix} p_t & v_t \end{bmatrix}^\top$) using an information EKF. An information EKF is similar to the standard EKF with the difference that instead of the covariance matrix P and the state x, the information matrix $\Omega = P^{-1}$ and information vector $y = \Omega x$ are used. The filter consists of four steps, which are as follows.

- *Prediction:* Propagate the state of the target and the covariance of the estimation using

$$\begin{aligned} \hat{x}_{t+1|t} &= F_r \hat{x}_t, \\ \Omega_{t+1|t} &= Q - QF_r(\Omega_t + F_r Q F_r^\top)^{-1} F_r^\top Q, \\ y_{t+1|t} &= \Omega_{t+1|t}\hat{x}_{t+1|t}. \end{aligned} \quad (10)$$

- *Correction:* When a new measurement z_{t+1} is received, compute

$$\begin{aligned} \tilde{z}_{t+1} &= z_{t+1} - h_r(\hat{x}_{t+1|t}) + H_r \hat{x}_{t+1|t}, \\ \tilde{y}_{t+1} &= y_{t+1|t} + H_r^\top R \tilde{z}_{t+1}, \\ \tilde{\Omega}_{t+1} &= \Omega_{t+1|t} + H_r^\top R H_r. \end{aligned} \quad (11)$$

- *Communication:* Each agent sends its newly observed information $\{\tilde{y}_{t+1}, \tilde{\Omega}_{t+1}\}$ to its neighboring agents, denoted by $\mathcal{N}_{out}^i(t+1)$, which indicates the agents who receive information from agent i at time step $t+1$.
- *Fusion:* Each agent fuses the information it has with the information it received from other agents through a weighted sum

$$\begin{aligned} y_{t+1} &= \sum_{j\in\mathcal{N}_{in}(t+1)} \pi_j \tilde{y}_{t+1}, \\ \Omega_{t+1} &= \sum_{j\in\mathcal{N}_{in}(t+1)} \pi_j \tilde{\Omega}_{t+1}, \end{aligned} \quad (12)$$

where π_j is the weight associated with the measurement of the j^{th} agent and we have $\sum_{j\in\mathcal{N}_{in}(t+1)} \pi_j = 1$. In this project, we use identical weights for agents' measurements since we assume they have identical sensors.

In the information EKF described above, similar to an ordinary EKF, we derive expressions for F_r and H_r from (3) and (4), respectively

$$\begin{aligned} F_r &= I_4 + dt \begin{bmatrix} 0_2 & I_2 \\ 0_2 & 0_2 \end{bmatrix}, \\ H_r &= \left.\frac{\partial h_r(x)}{\partial x}\right|_{\hat{x}_{t+1|t}} = \begin{bmatrix} \frac{(\hat{p}_{t_{t+1|t}} - p_a)^\top}{\|\hat{p}_{t_{t+1|t}} - p_a\|} & 0 & 0 \end{bmatrix}. \end{aligned}$$

A. Disturbed agent dynamics without obstacle avoidance

As described in Section II-A, in the first scenario, external disturbances act on the agents. We assume that each agent measures its position, and using this measurement estimates its state and the disturbance. Let $x = \begin{bmatrix} x_a & y_a & \theta_a & d \end{bmatrix}^\top \in \mathbb{R}^{3+q}$. Rewrite (1) and (2) to get

$$\begin{bmatrix} \dot{x}_a \\ \dot{y}_a \\ \dot{\theta}_a \\ \dot{d} \end{bmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & C_x \\ 0 & 0 & 0 & C_y \\ 0 & 0 & 0 & 0_q \\ 0_q & 0_q & 0_q & A \end{bmatrix} \begin{bmatrix} x_a \\ y_a \\ \theta_a \\ d \end{bmatrix} + \begin{bmatrix} \cos\theta_a & 0 \\ \sin\theta_a & 0 \\ 0 & 1 \\ 0_q & 0_q \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}}_{f_p(x,u)} + \eta, \quad (13)$$

where $\eta \sim \mathcal{N}(0, Q_a)$ is the zero-mean Gaussian process noise and $C_x \in \mathbb{R}^{1\times q}$ and $C_y \in \mathbb{R}^{1\times q}$ are rows of the matrix C in (1). The agent measures its position

$$z = \underbrace{\begin{bmatrix} x_a \\ y_a \end{bmatrix}}_{h_p(x)} + \zeta, \quad (14)$$

where $\zeta \sim \mathcal{N}(0, R_a)$ is the zero-mean Gaussian measurement noise. The EKF has two steps as follows.

- *Prediction:* Propagate the state and the covariance matrix using

$$\begin{aligned} \hat{x}_{t+1|t} &= f_p(\hat{x}_t, u_t), \\ P_{t+1|t} &= F_p P_t F_p^\top + Q_a. \end{aligned} \quad (15)$$

- *Correction:* Upon receiving a measurement, update the state and the covariance matrix using

$$\begin{aligned} \hat{x}_{t+1} &= \hat{x}_{t+1|t} + K(z_{t+1} - h(\hat{x}_{t+1|t})), \\ P_{t+1} &= (I_{3+q} - KH_p)P_{t+1|t}, \end{aligned} \quad (16)$$

with K being the Kalman gain

$$K = P_{t+1|t} H_p^\top (H_p P_{t+1|t} H_p^\top + R_a)^{-1},$$

and F_p and H_p being the Jacobian matrices defined as

$$F_p = \left.\frac{\partial f_p(x,u)}{\partial x}\right|_{\hat{x}_t, u_t} = I_{3+q} + dt \begin{bmatrix} 0 & 0 & -v_t \sin\hat{\theta}_{a_t} & C_x \\ 0 & 0 & v_t \cos\hat{\theta}_{a_t} & C_y \\ 0 & 0 & 0 & 0_q \\ 0_q & 0_q & 0_q & A \end{bmatrix},$$

$$H_p = \left.\frac{\partial h_p(x)}{\partial x}\right|_{\hat{x}_{t+1|t}} = \begin{bmatrix} I_2 & 0_{2\times(q+1)} \end{bmatrix},$$

where dt is the sampling time.

Each agent iterates through the two above-mentioned steps using some initial condition $\hat{x}_0$ and P_0 to estimate its state vector, which is then fed to the controller.

The next step is to design a Lyapunov-based feedback controller that can follow the generated trajectory despite the existence of the external disturbance.

We define the error equations as follows

$$\begin{aligned} e_p &= R^\top(p_a - p_d) - \epsilon \\ &= R^\top(p_a - r(\gamma) - \hat{p}_t) - \epsilon, \end{aligned} \quad (17)$$

$$e_\gamma = \dot{\gamma} - \omega_d, \quad (18)$$

where e_p is the position error between the tracker agent and the desired trajectory expressed in the tracker's body frame and ω_d is the desired angular velocity. Also, $\epsilon = \begin{bmatrix} \epsilon_x & \epsilon_y \end{bmatrix}^\top$ is an arbitrarily small nonzero vector which the idea behind will become clear next. Let $e = \begin{bmatrix} e_p{}^\top & e_\gamma \end{bmatrix}^\top$ be the complete tracking error vector.

The main objective is to derive a tracking control law for u to drive e to zero.

Taking the time derivative of (17), we get

$$\dot{e}_p = -\omega^\times e_p - \omega^\times \epsilon + \begin{bmatrix} v \\ 0 \end{bmatrix} - R^\top(v_t + r'(\gamma)\dot{\gamma} - Cd). \quad (19)$$

where $r'(\gamma) = \frac{\partial r}{\partial \gamma}$. After some calculations we can rewrite (19) as

$$\dot{e}_p = -\omega^\times e_p + \Lambda u - R^\top(v_t + r'(\gamma)\dot{\gamma} - Cd), \quad (20)$$

where

$$\Lambda := \begin{bmatrix} 1 & \epsilon_y \\ 0 & -\epsilon_x \end{bmatrix}. \quad (21)$$

Therefore, we set $\dot{\gamma} = \omega_d$ and propose the following controller

$$u = \Lambda^{-1}(R^\top(\hat{v}_t + r'(\gamma)\omega_d - C\hat{d}) - K_p e_p), \quad (22)$$

where $\hat{v}_t$ and $\hat{d}$ are the estimates of the target velocity and the external disturbance based on the target states and the disturbance EKFs, respectively. Also, $K_p \in \mathbb{R}^{2\times 2}$ is a diagonal gain matrix.

Remark 1. *In designing the feedback controller, ϵ is used to avoid losing control over ω. Also, from (21), it can be observed that only the ϵ_x has to be non-zero.*

Theorem 1. *For the tracker agents' model in (1), with the designed controller in (22), if the assumptions hold, the closed-loop system corresponding tracking errors will converge to a small neighborhood of the origin in finite time in the case of bounded state and disturbance estimation errors. Also, the tracking errors converge to the origin exponentially if the state and disturbance estimation errors converge to zero.*

Proof. Consider the following candidate Lyapunov function

$$V = \frac{1}{2} e_p^\top e_p. \quad (23)$$

Taking the derivative of (23) and using (20), we have

$$\begin{aligned} \dot{V} &= e_p^\top \dot{e}_p \\ &= e_p^\top(-\omega^\times e_p + \Lambda u - R^\top(v_t + r'(\gamma)\dot{\gamma} - Cd)). \end{aligned} \quad (24)$$

Now substituting the proposed control input (22) and considering that $e_p^\top \omega^\times e_p$ is zero, we have

$$\dot{V} = -e_p^\top K_p e_p - R^\top \tilde{v}_t + R^\top C\tilde{d}, \quad (25)$$

where $\tilde{v}_t = v_t - \hat{v}_t$ and $\tilde{d} = d - \hat{d}$ are target velocity and external disturbance estimation errors, respectively. Now since all the terms in (25) produce scalar values, we can write

$$\dot{V} \leq -\|e_p^\top K_p e_p\| - \|R^\top \tilde{v}_t\| + \|R^\top C\tilde{d}\|. \tag{26}$$

After computing the eigenvalue decomposition of K_p, singular value decomposition of C, and the fact that $\|R\| = 1$, we have

$$\dot{V} \leq -\lambda_{\min}(K_p)\|e_p\|^2 - \|\tilde{v}_t\| + \sigma_{\min}(C)\|\tilde{d}\|, \tag{27}$$

where $\lambda_{\min}(.)$ and $\sigma_{\min}(.)$ denote the minimum eigenvalue and minimum singular value, respectively. If $\tilde{v}_t$ and $\tilde{d}$ are bounded, the closed-loop tracking errors converge to a small neighborhood of the origin in finite time and the system achieves practical finite-time stability. Also, if $\tilde{v}_t \to 0$ and $\tilde{d} \to 0$, we get

$$\dot{V} \leq -2\lambda_{\min}(K_p)V. \tag{28}$$

Therefore, the system achieves exponential stability. □

B. Undisturbed agent dynamics with collision/obstacle avoidance

In this section, since we have a disturbance-free setting we have $d = 0$, and subsequently $\hat{d} = \tilde{d} = 0$. Therefore, the proposed nominal controller for trajectory tracking is similar to (22), expressed as

$$u = \Lambda^{-1}(R^\top(\hat{v}_t + r'(\gamma)\omega_d) - K_p e_p). \tag{29}$$

Remark 2. *In order to study the stability of controller (29), the same argument in Theorem 1 can be used by considering* $\hat{d} = \tilde{d} = 0$.

In what follows we will see how we can achieve collision/obstacle avoidance for the tracker agents.

Let $x = \begin{bmatrix} x_a & y_a & \theta_a \end{bmatrix}^\top$, $u = \begin{bmatrix} v & \omega \end{bmatrix}^\top$, where x_a, y_a, θ_a denote the current coordinates and the orientation of the center of the agents with respect to the x-axis, and v and ω denote the linear and the angular velocity. The coordinates of the head of the agent could be represented as follows

$$\begin{bmatrix} x_h \\ y_h \end{bmatrix} = \begin{bmatrix} x_a + l\cos\theta_a \\ y_a + l\sin\theta_a \end{bmatrix}, \tag{30}$$

where l is the distance between the center and each agent's head. Now considering $d = 0$, (1) can be rewritten in the input-affine form as

$$\dot{x} = f(x) + g(x)u, \tag{31}$$

where

$$f(x) := \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \quad g(x) := \begin{bmatrix} \cos\theta_a & 0 \\ \sin\theta_a & 0 \\ 0 & 1 \end{bmatrix}. \tag{32}$$

We now formally define the concept of safety via CBF.

Definition 1 (CBF [10])**.** *Let $\mathcal{C}$ be the superlevel set of a continuously differentiable function $B : \mathbb{R}^n \to \mathbb{R}$, so that $\mathcal{C} = \{x \in \mathbb{R}^n \mid B(x) \geq 0\}$. B is a* Control Barrier Function (CBF) *for system (31) if there exists a class $\mathcal{K}$ function α such that*

$$\sup_{u \in U} \big(L_f B(x) + L_g B(x)u\big) \geq -\alpha(B(x)) \quad \forall x \in \mathcal{C}, \tag{33}$$

where $L_f B(x)$ and $L_g B(x)$ are the Lie derivatives of $B(x)$ along the system dynamics.

If B is a CBF, define for each $x \in \mathcal{C}$ the set $K_{\text{CBF}}(x) := \big\{u \in U \mid L_f B(x) + L_g B(x)u \geq -\alpha(B(x))\big\}$.

Theorem 2 (See [10])**.** *Let $\mathcal{C}$ be defined as in Definition 1. If B is a CBF for system (31), then any Lipschitz continuous controller $u(x) \in K_{\text{CBF}}(x)$ has the property that if $x(0) \in \mathcal{C}$, then $x(t) \in \mathcal{C}$ for all $t \geq 0$.*

Proof. See [10]. □

The property of $\mathcal{C}$ in Theorem 2 is also called *forward invariance* [18].

We will demonstrate how to design the CBFs for collision avoidance with all obstacles. Consider the agents are sharing an open space with a set of $\mathbb{N}$ static obstacles. For notations, subscript i is used to distinguish each obstacle, which is represented by $O_i \in \mathbb{O} = \{O_0, O_1, \ldots, O_{N-1}\}$. Assuming that the agents and the obstacles are circular shapes with radii of r_a and r_{O_i}, respectively. Hence the safe distance between the robot and obstacle O_i is defined as $r_i = r_a + r_{O_i}$. If the distance between them exceeds this safety threshold, then safety can be guaranteed. If the position of the obstacle O_i is denoted by (x_{O_i}, y_{O_i}), we can design multiple CBFs for each agent in the following form [11]

$$B_i(x) = (x_a + l\cos\theta_a - x_{O_i})^2 + (y_a + l\sin\theta_a - y_{O_i})^2 - r_i^2. \tag{34}$$

In this setting, to provide inter-agent collision avoidance, each agent is also considered as a moving obstacle for all other agents and; therefore, using (34) we provide inter-agent collision avoidance as well.

Now, we need to compute $\dot{B}(x)$ and derive the Lie derivatives. After calculations we obtain

$$L_f B_i(x) = 0, \tag{35}$$

and

$$L_g B_i(x) = \begin{bmatrix} 2\Delta x\cos\theta_a + 2\Delta y\sin\theta_a \\ -2\Delta x l\sin\theta_a + 2\Delta y l\cos\theta_a \end{bmatrix}^\top, \tag{36}$$

where

$$\begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} := \begin{bmatrix} x_a + l\cos\theta_a - x_{O_i} \\ y_a + l\sin\theta_a - y_{O_i} \end{bmatrix}. \tag{37}$$

It can be verified that if $L_g B_i(x)$ to be a zero matrix, we have the following case: $\Delta x\sin\theta_a = \Delta y\cos\theta_a$, without loss of generality, we assume $\cos\theta_a \neq 0$. Then we can simplify this equation to get $\Delta y = \frac{\sin\theta_a}{\cos\theta_a}\Delta x$, next we can get $\Delta x = \Delta y = 0$. However, if both Δx and Δy equal to zero, it would mean that the robot is already inside the obstacle, which is impossible. Therefore, $L_g B_i(x)$ is always non-zero to ensure $B_i(x)$ to be a valid CBF.

Remark 3. *We consider the distance between the head of the agents to the center of the obstacles in (34) to not lose control over ω. If we use the coordinates of the center of the agents instead, the second element of $L_g B_i(x)$ becomes zero and we lose control over ω.*

Through Quadratic Programming (QP), the CBF-QP filter is designed as follows

$$\begin{aligned} \min_{u\in\mathbb{R}^m} \quad & \|u - u_{\text{nominal}}\|^2 \\ \text{subject to} \quad & L_f B_i(x) + L_g B_i(x) u \geq -\alpha(B_i(x)), \\ & u_{\min} \leq u \leq u_{\max}. \end{aligned} \tag{38}$$

where u_{nominal} is the nominal control input designed in (29) to achieve the primary task of trajectory tracking. Therefore, using Remark 2 and Theorem 2 we can guarantee collision-free trajectory tracking for the agents in the disturbance-free setting.

IV. Simulation/Experimental Results

The simulation results for the two scenarios are presented here. It is assumed that the target's trajectory is

$$p_t = \begin{bmatrix} 10\sin(0.02t + \pi) \\ 0.2t \end{bmatrix}.$$

For further numerical values see the following sections and the `runMe.m` code provided with this document.

A. Disturbed agent dynamics without obstacle avoidance

We used the following numerical values for C and A in (1) and (2)

$$A = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix},$$
$$C = \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & -1 & 1 & 0 \end{bmatrix}.$$

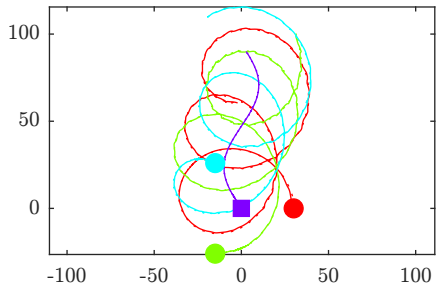


Fig. 1. Agents and target trajectories under external disturbance with disturbance estimation. The magenta-colored rectangle represents the target. Colored circles represent the agents: agent 1 (red), agent 2 (green), and agent 3 (cyan). Dotted lines represent the desired trajectories and solid lines represent the actual trajectories.

It can be observed from Fig. 1, that in the case disturbance estimation is fed to the controller, the agents' and target's positions are estimated with high accuracy since the target's position estimation (purple dotted line) and the actual trajectory (purple solid line), as well as agents' desired trajectories (red, green, and cyan dotted lines) and actual

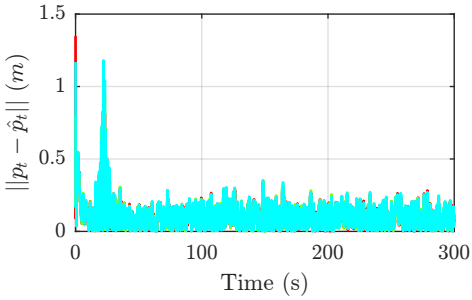


Fig. 2. Target position estimation errors. Colored lines represent the agents: agent 1 (red), agent 2 (green), and agent 3 (cyan).

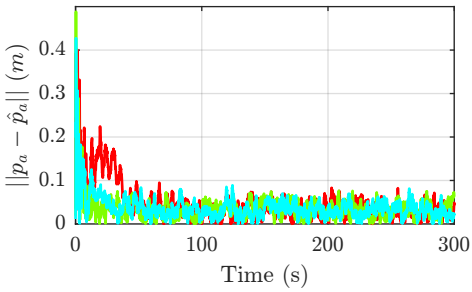


Fig. 3. Agents position estimation errors. Colored lines represent the agents: agent 1 (red), agent 2 (green), and agent 3 (cyan).

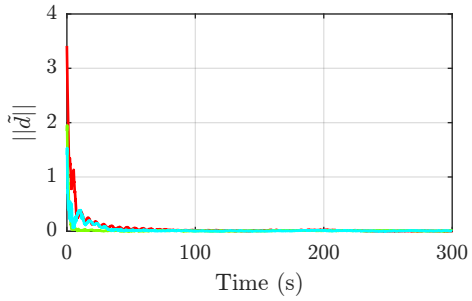


Fig. 4. External disturbance estimation errors. Colored lines represent the agents: agent 1 (red), agent 2 (green), and agent 3 (cyan).

trajectories (red, green, and cyan solid lines) are matched, and the corresponding errors shown in Fig. 2 and Fig. 3 are close to zero. In addition, from Fig. 4, it can be observed that the designed EKF's external disturbance estimation error also goes to zero, which helps the controller to compensate for the disturbance effects properly.

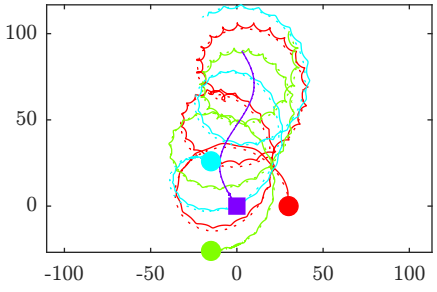


Fig. 5. Agents and target trajectories under external disturbance without disturbance estimation. The magenta-colored rectangle represents the target. Colored circles represent the agents: agent 1 (red), agent 2 (green), and agent 3 (cyan). Dotted lines represent the desired trajectories and solid lines represent the actual trajectories.

In the case of no disturbance estimation, it can be seen from Fig. 5, that the agents' cannot follow their desired trajectories properly due to the external disturbance effects. Consequently, this results in a less accurate estimation of the agents' positions.

B. Undisturbed agent dynamics with collision/obstacle avoidance

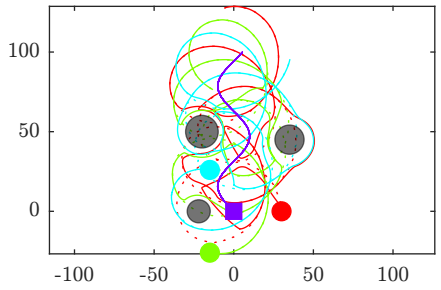


Fig. 6. Agents and target trajectories with obstacles of different radii. The magenta-colored rectangle represents the target. Colored circles represent the agents: agent 1 (red), agent 2 (green), and agent 3 (cyan). Black-colored circles denote the obstacles. Dotted lines represent the desired trajectories and solid lines represent the actual trajectories.

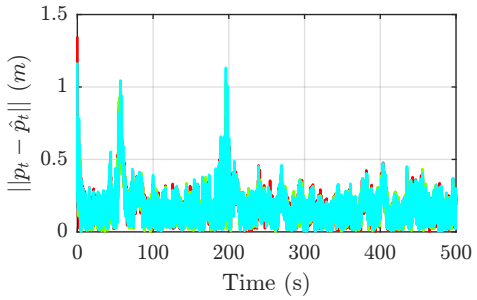


Fig. 7. Target position estimation errors. Colored lines represent the agents: agent 1 (red), agent 2 (green), and agent 3 (cyan).

It can be observed, from Fig. 6, that in the presence of obstacles of various radii in the environment, the agents' desired trajectories (red, green, and cyan dotted lines) generated for achieving optimal target position estimation inevitably intersect with the obstacles. However, it can be seen that by using the designed controller along with the CBF safety filter, the agents can effectively avoid collisions and obstacles and follow the desired trajectories to achieve safe and optimal target position estimation and tracking. This can also be observed from Fig. 7 since the target's position estimation errors are close to zero. Also, upon closer look on Fig. 7, two spikes on the estimation errors are seen that are most likely caused by the agents' reaching obstacles that temporarily cause the agents to deviate from the desired trajectories, leading to the spike in the estimation errors.

V. Conclusion

In this project, we considered the problem of multi-agent target localization and tracking. We first discussed how the relative configuration of the agents with respect to the target affects the quality of estimation and how agents should be relatively located to obtain *optimal estimation*. The problem was solved in two different scenarios. First, the agents were under external disturbance, and second, the agents were navigating in an obstacle-abundant environment. In both scenarios, we designed algorithms (Extended Kalman Filter (EKF) and Control Barrier Functions (CBFs), respectively) that, when integrated into the main controller, were able to successfully accomplish the localization and tracking tasks. Future research directions include combining the two scenarios, i.e., agents operating in an environment full of obstacles and under external disturbance. The application of robust CBF can be explored in this scenario. Also, bounded disturbances can be considered and set-based observers instead of the EKF can be used for disturbance estimation.

References

[1] J. Kim, D. Jang, and H. J. Kim, "Distributed multi-agent target search and tracking with gaussian process and reinforcement learning," *International Journal of Control, Automation and Systems*, vol. 21, no. 9, pp. 3057–3067, 2023.

[2] D. S. Drew, "Multi-agent systems for search and rescue applications," *Current Robotics Reports*, vol. 2, pp. 189–200, 2021.

[3] B. Schlotfeldt, N. Atanasov, and G. J. Pappas, "Adversarial information acquisition," in *Proc. Workshop Adv. Robot. RSS*, 2018, pp. 1–4.

[4] H. Huang, A. V. Savkin, and W. Ni, "Decentralized navigation of a uav team for collaborative covert eavesdropping on a group of mobile ground nodes," *IEEE Transactions on Automation Science and Engineering*, vol. 19, no. 4, pp. 3932–3941, 2022.

[5] M. Naserian, A. Ramazani, A. Khaki-Sedigh, and A. Moarefianpour, "Fast terminal sliding mode control for a nonlinear multi-agent robot system with disturbance," *Systems Science & Control Engineering*, vol. 8, no. 1, pp. 328–338, 2020.

[6] Y. Liu and Y. Jia, "Adaptive leader-following consensus control of multi-agent systems using model reference adaptive control approach," *IET Control Theory & Applications*, vol. 6, no. 13, pp. 2002–2008, 2012.

[7] K. Coleman, H. Bai, and C. N. Taylor, "Extended invariant-ekf designs for state and additive disturbance estimation," *Automatica*, vol. 125, p. 109464, 2021.

[8] D. Ioan, I. Prodan, S. Olaru, F. Stoican, and S.-I. Niculescu, "Mixed-integer programming in motion planning," *Annual Reviews in Control*, vol. 51, pp. 65–87, 2021.

[9] S. Yang, T. Li, Q. Shi, W. Bai, and Y. Wu, "Artificial potential-based formation control with collision and obstacle avoidance for second-order multi-agent systems," in *2020 7th International Conference on Information, Cybernetics, and Computational Social Systems (ICCSS)*. IEEE, 2020, pp. 58–63.

[10] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European control conference (ECC)*. IEEE, 2019, pp. 3420–3431.

[11] J. Huang, Z. Liu, J. Zeng, X. Chi, and H. Su, "Obstacle avoidance for unicycle-modelled mobile robots with time-varying control barrier functions," in *IECON 2023-49th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2023, pp. 1–6.

[12] M. Bibuli, G. Bruzzone, M. Caccia, and L. Lapierre, "Path-following algorithms and experiments for an unmanned surface vehicle," *Journal of Field Robotics*, vol. 26, no. 8, pp. 669–688, 2009.

[13] R. Olfati-Saber, "Distributed kalman filtering for sensor networks," in *2007 46th IEEE Conference on Decision and Control*. IEEE, 2007, pp. 5492–5498.

[14] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[15] T. D. Barfoot, *State estimation for robotics*. Cambridge University Press, 2024.

[16] N. T. Hung, F. F. Rego, and A. M. Pascoal, "Cooperative distributed estimation and control of multiple autonomous vehicles for range-based underwater target localization and pursuit," *IEEE Transactions on Control Systems Technology*, vol. 30, no. 4, pp. 1433–1447, 2021.

[17] T. Oliveira, A. P. Aguiar, and P. Encarnacao, "Moving path following for unmanned aerial vehicles with applications to single and multiple target tracking problems," *IEEE Transactions on Robotics*, vol. 32, no. 5, pp. 1062–1078, 2016.
[18] H. Khalil, *Nonlinear systems*. Prentice Hall, 2002.