

Cooperative Receding Horizon 3D Coverage Control with a Team of Networked Aerial Agents

Savvas Papaioannou, Panayiotis Kolios, Theocharis Theocharides,
Christos G. Panayiotou and Marios M. Polycarpou

Abstract—This work proposes a receding horizon coverage control approach which allows multiple autonomous aerial agents to work cooperatively in order cover the total surface area of a 3D object of interest. The cooperative coverage problem which is posed in this work as an optimal control problem, jointly optimizes the agents’ kinematic and camera control inputs, while considering coupling constraints amongst the team of agents which aim at minimizing the duplication of work. To generate look-ahead coverage trajectories over a finite planning horizon, the proposed approach integrates visibility constraints into the proposed coverage controller in order to determine the visible part of the object with respect to the agents’ future states. In particular, we show how non-linear and non-convex visibility determination constraints can be transformed into logical constraints which can easily be embedded into a mixed integer optimization program.

I. INTRODUCTION

The recent technological advances in unmanned aerial vehicle (UAV) technology has led to a transformative impact in various application domains [1]–[4]. Specifically, during the last decade the problem of coverage path planning (CPP) [5] with UAVs has gained a lot of attention. Notably, the work in [6] proposes a coverage path planning method for a single UAV agent that optimizes the UAV’s turning motion in planar convex polygonal areas, whereas in [7] the authors consider a fixed-wing UAV, and utilize algorithms for the Chinese postman problem in order to compute an Eulerian path that covers all cells in the region of interest. An information theoretic terrain coverage planning approach is proposed in [8] for a single fixed-wing UAV agent, whereas in [9] a spatiotemporal clustering-based coverage approach is proposed for assigning regions of interest that need to be covered to a team of heterogeneous UAVs. The problem of multi-UAV CPP is also investigated in [10] emphasizing on the energy efficiency of the mission. The multi-agent CPP problem for terrain coverage with workload balancing has recently been investigated in [11], and in [12], [13] the coverage problem is investigated for 3D cuboid-like objects of interest. Despite the continuous advancements in this domain, there is still work to be done until this

The authors are with the KIOS Research and Innovation Centre of Excellence (KIOS CoE) and the Department of Electrical and Computer Engineering, University of Cyprus, Nicosia, 1678, Cyprus. E-mail:{papaioannou.savvas, pkolios, ttheocharides, christosp, mpolycar}@ucy.ac.cy This work was undertaken as part of the GLIMPSE project EXCELLENCE/0421/0586 which is co-financed by the European Regional Development Fund and the Republic of Cyprus through the Research and Innovation Foundation’s RESTART 2016-2020 Programme for Research, Technological Development and Innovation and supported by the European Union’s Horizon 2020 research and innovation programme under grant agreement No 739551 (KIOS CoE), and from the Government of the Republic of Cyprus through the Cyprus Deputy Ministry of Research, Innovation and Digital Policy.

technology reaches the required level of maturity to enable autonomous coverage missions. The majority of UAV-based CPP approaches discussed above, and found in the literature, steer their focus towards covering mainly 2D planar areas and terrains [14]–[16], and not 3D objects. Moreover, coverage planning approaches which rely on simple geometric patterns (e.g., back-and-forth and zig-zag motions) [17], usually fail to generalize in 3D environments. Existing 3D coverage planning techniques often require specialized types of objects e.g., cuboids-like structures [12], and mainly utilize UAVs equipped with fixed, and uncontrollable sensors [18] thus reducing the problem to a standard path-planning problem which does not accounts for the complexities of coverage control in 3D settings.

To tackle some of the challenges discussed above, in this work we propose a coverage controller which enables a team of autonomous UAV agents to compute cooperative finite-length look-ahead trajectories by jointly optimizing their kinematic and camera control inputs in order to cover in 3D the total surface area of an object of interest. In order to generate the look-ahead coverage plans we simulate the physical behavior of light in order to determine the visible parts of the object with respect to the future states of the agents, and we show how non-linear and non-convex visibility determination constraints can be transformed into logical constraints which can be embedded into a mixed integer optimization problem.

The rest of the paper is organized as follows. Section II develops the system model, Section III formulates the problem tackled, and Section IV discusses the details of the proposed approach. Finally, Section V evaluates the proposed approach, and Section VI concludes the paper, and discusses future directions.

II. PRELIMINARIES

A. Agent Kinematic Model

We assume that a cooperative team of N autonomous networked aerial agents denoted by $j \in [1, \dots, N]$, operate inside a bounded 3D environment $\mathcal{E} \subset \mathbb{R}^3$, with discrete-time kinematics given by the following linear state-space model:

$$x_k^j = \begin{bmatrix} \mathbf{1}_{3 \times 3} & \Delta T \times \mathbf{1}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & (1 - \gamma) \times \mathbf{1}_{3 \times 3} \end{bmatrix} x_{k-1}^j + \begin{bmatrix} \mathbf{0}_{3 \times 3} \\ \frac{\Delta T}{m} \times \mathbf{1}_{3 \times 3} \end{bmatrix} u_k^j, \quad (1)$$

which is abbreviated hereafter as $x_k^j = Ax_{k-1}^j + Bu_k^j$, where $x_k^j = [(x_k^{j,P})^\top, (x_k^{j,V})^\top]^\top \in \mathcal{X} \subset \mathbb{R}^6$ denotes agent’s j kinematic state at time-step k , which is composed of position (i.e., $x_k^{j,P} \in \mathbb{R}^3$), and velocity (i.e., $x_k^{j,V} \in \mathbb{R}^3$) components, in the 3D cartesian coordinate system. It is assumed that the agents are controllable, and that can be

commanded to execute a certain direction and speed through the control input $u_k^j \in \mathcal{U} \subset \mathbb{R}^3, j \in [1, \dots, N]$ which denotes the applied input force. In Eq. (1) the parameter ΔT denotes the sampling interval, γ models the air resistance coefficient, and finally m is agent's j mass, which without loss of generality is assumed to be the same for all agents. In this work it is assumed that all agents maintain a wireless communication link with a central mobile base-station which is used for information exchange.

B. Agent Camera Model

Each agent $j \in [1, \dots, N]$ is equipped with a camera system attached to a gimbal device which allows the 3D rotation of the camera's finite field-of-view (FOV). In this work, the camera's FOV is modelled as a regular right pyramid which exhibits four triangular lateral faces and a rectangular base. The camera optical center is positioned directly above the centroid of the rectangular FOV base. In essence, the camera's FOV is determined in this work by the parameter set (ℓ, w, r) , where ℓ and w are the length and width of the FOV rectangular base respectively, and r (i.e., the height of the pyramid) determines the FOV range. Let the five FOV vertices of a downward facing camera, centered at the origin of the 3D cartesian coordinate system to be given by the 3-by-5 matrix \mathcal{V}_0 as:

$$\mathcal{V}_0 = \begin{bmatrix} -\frac{\ell}{2} & \frac{\ell}{2} & \frac{\ell}{2} & -\frac{\ell}{2} & 0 \\ \frac{w}{2} & \frac{w}{2} & -\frac{w}{2} & -\frac{w}{2} & 0 \\ -r & -r & -r & -r & 0 \end{bmatrix}. \quad (2)$$

The FOV is rotated in 3D space by commanding the gimbal controller to execute sequentially two elemental rotations i.e., one rotation by angle $\theta \in [0, \pi)$ around the y -axis, followed by a rotation $\phi \in [0, 2\pi)$ around the z -axis. Therefore, at each time-step k the agent j with position $x_k^{j,\mathbf{P}}$ can rotate the camera's FOV anywhere inside the surveillance region, via the following geometric transformation:

$$\mathcal{V}_k^j(\theta_k^j, \phi_k^j, x_k^{j,\mathbf{P}})^i = R_z(\phi_k^j)R_y(\theta_k^j)\mathcal{V}_0^i + x_k^{j,\mathbf{P}}, \forall i \in [1, \dots, 5] \quad (3)$$

where \mathcal{V}_0^i denotes the i th column of \mathcal{V}_0 , and therefore $\mathcal{V}_k^j(\theta_k^j, \phi_k^j, x_k^{j,\mathbf{P}})^i$ is the corresponding rotated and translated vertex of the FOV. The parameters θ_k^j and ϕ_k^j are the input rotation angles, and $R_y(\alpha)$, $R_z(\alpha)$ represent the basic 3-by-3 rotation matrices [19] which rotate vectors by an angle α around the y - and z -axis respectively. We assume that the gimbal device is bounded to operate within a predefined finite set of admissible input rotation angles $\Xi = \Theta \times \Phi = \{(\theta, \phi) | \theta \in \Theta, \phi \in \Phi\}$, where \times denotes the Cartesian product on the finite sets Θ and Φ . Therefore, the camera FOV of each agent j can take, at each time-step k , one out of $|\Xi|$ possible configurations ($|\Xi|$ denotes the cardinality of the set Ξ).

Finally, it is assumed that at each time-step k a finite set of (straight) light-rays, which model the direction of the propagation of light, enter the camera's optical center and cause matter to be imaged. The set of light-rays captured through the agent's camera FOV $\mathcal{V}_k^j(\theta_k^j, \phi_k^j, x_k^{j,\mathbf{P}})$ is denoted in this work as $\mathcal{L}_k^j(\theta_k^j, \phi_k^j, x_k^{j,\mathbf{P}}) = \{\Lambda_{k,1}^j, \dots, \Lambda_{k,n}^j\}$, where $\Lambda_{k,i}^j$ denotes the individual light-ray in the set which is

further given by the line-segment:

$$\Lambda_{k,i}^j = \lambda_{k,i}^j + d(x_k^{j,\mathbf{P}} - \lambda_{k,i}^j), \quad \forall d \in [0, 1], \quad (4)$$

where $x_k^{j,\mathbf{P}}$ is the light-ray's end point which enters the camera's optical center at time-step k , $\lambda_{k,i}^j$ is a fixed point on the camera's FOV base denoting the ray's origin, and d is a scalar. Note here that every FOV state $\mathcal{V}_k^j(\theta_k^j, \phi_k^j, x_k^{j,\mathbf{P}})$ generates a different set of light-rays $\mathcal{L}_k^j(\theta_k^j, \phi_k^j, x_k^{j,\mathbf{P}})$.

The goal of the agents is to cover with their cameras the total surface area $\partial\mathcal{O}$ of a known object of interest $\mathcal{O} \in \mathcal{E}$. This object's surface area has been triangulated into a finite set of non-overlapping triangular facets $\tau \in \mathcal{T}$, where \mathcal{T} is the object's surface triangle mesh. Consequently, our aim becomes the generation of cooperative coverage trajectories which cover all facets $\tau \in \mathcal{T}$.

III. PROBLEM FORMULATION

Given a team of N cooperative agents $j = [1, \dots, N]$, find the joint kinematic (i.e., input force $u_k^j, \forall j$) and camera (i.e., rotation angles $(\theta_k^j, \phi_k^j), \forall j$) control inputs over a sufficiently large planning horizon of length K' time-steps which result in the optimal coverage of the total surface area \mathcal{T} of the object of interest \mathcal{O} . A high-level formulation of the coverage problem discussed above is shown in Problem (P1), posed as an optimal control problem. As shown in Problem (P1), we are seeking to find the agents' joint control inputs $\{u_k^j, \theta_k^j, \phi_k^j\}_{j=1}^N$ over a finite planning horizon $k = [1, \dots, K']$ of length K' time-steps which optimize a mission related objective function (e.g., the coverage elapsed time) denoted as $\mathcal{F}_{\text{mission}}$, and shown in Eq. (5a), subject to the set of coverage constraints shown in Eq. (5b)-(5c).

(P1) Coverage Problem

$$\arg \min_{\{u_k^j, \theta_k^j, \phi_k^j\}_{j=1}^N} \mathcal{F}_{\text{mission}} \quad (5a)$$

subject to: $k \in [1, \dots, K'], j \in [1, \dots, N]$

$$x_k^j = A^k x_0^j + \sum_{\kappa=1}^k A^{k-\kappa} B u_{\kappa}^j \quad \forall k, \forall j \quad (5b)$$

$$\exists \kappa \leq K' : \tau \in \bigcup_{j=1}^N \Delta(\mathcal{V}_{\kappa}^j(\theta_{\kappa}^j, \phi_{\kappa}^j, x_{\kappa}^{j,\mathbf{P}})) \quad \forall \tau \in \mathcal{T} \quad (5c)$$

The constraint in Eq. (5b) is due to the agent's kinematic model as discussed in Sec. II-A which has been obtained from the recursive application of Eq. (1) for time-steps $k = [1, \dots, K']$, with a known initial state $x_0^j = \bar{x}^j$. The next constraint shown in Eq. (5c) ensures that the total surface area \mathcal{T} of the object of interest is cooperatively covered by the agents during the mission. In order to achieve this we require that for each facet $\tau \in \mathcal{T}$ there exists a time-step $\kappa \leq K'$ for which τ resides within the convex-hull (indicated by the Δ operator) of some agent's camera FOV. In essence we require that each facet τ is covered by some agent j . However, in order to be able plan accurate look-ahead coverage trajectories, we first need to determine which parts of the object are actually visible given the future states of the agents (i.e., determine the visible area given the predicted camera poses). In that sense, the constraint in Eq. (5c) is actually incomplete since it does not indicate what

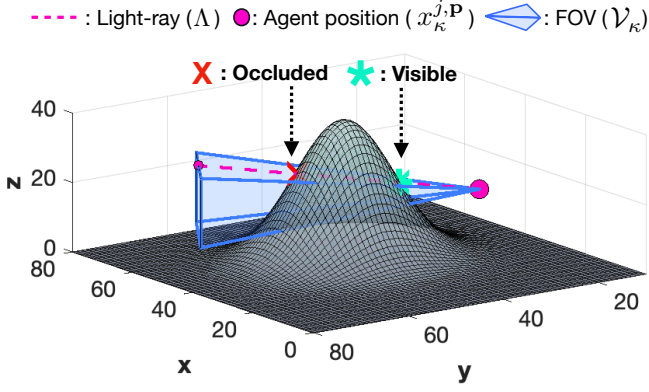


Fig. 1. Illustrative example of the visibility determination problem, where $x_{\kappa}^{j,P}$ is the future predicted agent position at time-step $k + \kappa|k$, and \mathcal{V}_{κ} is the predicted FOV configuration at the same time-step. With the FOV pose shown in the figure the facet indicated with \times is not visible as there is no light-ray that can be traced back to it i.e., all light-rays are blocked. On the other hand the facet indicated with \star can be observed as shown.

is the observable FOV. In the next section we show how we have incorporated visibility determination constraints into Problem (P1), in an effort to generate cooperative look-ahead coverage plans.

IV. COOPERATIVE RECEDING HORIZON 3D COVERAGE CONTROL

The coverage planning problem discussed in the previous section is quite challenging to be solved efficiently. In particular, observe that a feasible solution to this problem is directly coupled with the length of the planning horizon K' i.e., if K' is too short, then no feasible solution may exist, while if K' is too long then the computational complexity increases unnecessarily. For the reasons discussed above, Problem (P1) is re-formulated as a receding horizon optimal control problem in where the joint control inputs over all agents $j = [1, \dots, N]$ are computed in an on-line fashion at each time-step k inside a rolling finite planning horizon K i.e., $\{u_{k+\kappa|k}^j, \theta_{k+\kappa|k}^j, \phi_{k+\kappa|k}^j\}, \forall \kappa \in [1, \dots, K]$. As a result, at each time-step k the agents plan finite-length look-ahead coverage trajectories $x_{k+\kappa|k}^j, \kappa = [1, \dots, K]$, where the notation $x_{\kappa|k}$ denotes the predicted agent state at time-step $\kappa \geq k$, which was computed at time-step k .

A. Visibility Determination

In order to generate the agents' future coverage trajectories over a finite planning horizon K , we first need to have a way of determining which parts of the object's surface area are visible given the future planned states of the agents at time-steps $k + \kappa|k, \kappa \in [1, \dots, K]$, abbreviated for simplicity as κ . Intuitively, the facet $\tau \in \mathcal{T}$, is visible through the agent's j camera FOV $\mathcal{V}_{\kappa}^j(\theta_{\kappa}^j, \phi_{\kappa}^j, x_{\kappa}^{j,P})$ at time-step κ when: a) τ resides inside the convex-hull of the agent's camera FOV i.e., $\tau \in \Delta(\mathcal{V}_{\kappa}^j(\theta_{\kappa}^j, \phi_{\kappa}^j, x_{\kappa}^{j,P}))$, and b) there exists a light-ray $\Lambda_{\kappa,i}^j \in \mathcal{L}_{\kappa}^j(\theta_{\kappa}^j, \phi_{\kappa}^j, x_{\kappa}^{j,P})$ which enters the camera's optical center and can be traced back to facet τ . On the other hand, when no light-ray can be traced back to τ , indicates that the specific facet is not visible since the propagation of light is blocked. Specifically, the notion of visibility can now be defined as follows: The facet $\tau \in \mathcal{T}$ is visible through the

agent's j camera FOV $\mathcal{V}_{\kappa}^j(\theta_{\kappa}^j, \phi_{\kappa}^j, x_{\kappa}^{j,P})$ at the future time-step κ when:

$$\exists \Lambda_{\kappa,i}^j \in \mathcal{L}_{\kappa}^j(\theta_{\kappa}^j, \phi_{\kappa}^j, x_{\kappa}^{j,P}) : \Lambda_{\kappa,i}^j \oplus \mathcal{T} = \tau, \quad (6)$$

where the operator \oplus returns the facet $\tau \in \mathcal{T}$ which intersects last with the light-ray $\Lambda_{\kappa,i}^j$; otherwise it returns \emptyset if no facet τ intersects with the light-ray. Let us denote the equation of the plane which contains facet τ as $\alpha_{\hat{\tau}} \cdot x = \beta_{\hat{\tau}}$, where $\hat{\tau} \in [1, \dots, |\mathcal{T}|]$ is the index pointing to facet $\tau \in \mathcal{T}$, $\alpha_{\hat{\tau}} \in \mathbb{R}^3$ is the unit outward normal vector to the plane containing τ , $x \in \mathbb{R}^3$, $\beta_{\hat{\tau}}$ is a scalar, and the notation $a \cdot b$ denotes the dot product of the vectors a and b . Subsequently, the operation $\Lambda_{\kappa,i}^j \oplus \tau$ finds the intersection point (if exists) between the light-ray $\Lambda_{\kappa,i}^j$ which is given by Eq. (4), and the plane which contains facet τ as follows:

$$\alpha_{\hat{\tau}} \cdot [\lambda_{\kappa,i}^j + d(x_{\kappa}^{j,P} - \lambda_{\kappa,i}^j)] = \beta_{\hat{\tau}} \implies \quad (7a)$$

$$d = \frac{\beta_{\hat{\tau}} - \alpha_{\hat{\tau}} \cdot \lambda_{\kappa,i}^j}{\alpha_{\hat{\tau}} \cdot (x_{\kappa}^{j,P} - \lambda_{\kappa,i}^j)}, \quad (7b)$$

where Eq. (7a) is the result of the substitution of $\Lambda_{\kappa,i}^j$ for x in the equation of the plane which contains facet τ , and then in Eq. (7b) we solve for d . Consequently, if the denominator of Eq. (7b) is equal to zero, the light-ray $\Lambda_{\kappa,i}^j$ and the facet τ are parallel which results in either no visibility (i.e., $\beta_{\hat{\tau}} - \alpha_{\hat{\tau}} \cdot \lambda_{\kappa,i}^j \neq 0$) or distorted view (i.e., when $\beta_{\hat{\tau}} - \alpha_{\hat{\tau}} \cdot \lambda_{\kappa,i}^j = 0$). In essence, we are interested in the scenario where there exists a single point intersection between the light-ray $\Lambda_{\kappa,i}^j$ and the plane which contains the facet τ i.e., when $\alpha_{\hat{\tau}} \cdot (x_{\kappa}^{j,P} - \lambda_{\kappa,i}^j) \neq 0$ and $d \in [0, 1]$ which indicates the visibility of τ by the light-ray $\Lambda_{\kappa,i}^j$ i.e., the light-ray is not blocked and traces back to τ , as illustrated in Fig. 1.

The procedure discussed above must be evaluated at each time-step inside the planning horizon for all pairwise combinations of light-rays $\Lambda_{\kappa,i}^j \in \mathcal{L}_{\kappa}^j(\theta_{\kappa}^j, \phi_{\kappa}^j, x_{\kappa}^{j,P})$, and facets $\tau \in \mathcal{T}$ which not only is computationally expensive, but also requires the integration of non-convex and non-linear constraints which are challenging to be handled efficiently during optimization. In order to bypass this challenge, in this work we follow an alternative procedure which allows us to first learn a set of state-dependant constraints for determining visibility, and subsequently incorporating these constraints into the coverage planning optimization problem as logical constraints.

To achieve this the environment \mathcal{E} is first decomposed into a 3D grid $\tilde{\mathcal{E}}$ consisting of a finite number of disjoint cells i.e., $\tilde{\mathcal{E}} = \{c_1, \dots, c_{|\tilde{\mathcal{E}}|}\}$, such that $\bigcup_{i=1}^{|\tilde{\mathcal{E}}|} c_i = \tilde{\mathcal{E}}$. Then, within each cell $c \in \tilde{\mathcal{E}}$, we randomly sample the agent states i.e., (x^P, θ, ϕ) and then the visibility determination procedure discussed above i.e., Eq. (7b) is performed for various configurations of the FOV state $\mathcal{V}(\theta, \phi, x^P)$ to identify the visible facets τ . Let us denote with $\mathcal{L}_{\hat{c}}$ the set of light-rays that have been obtained from the application of N_r random joint combinations of the parameter set $(x^P, \theta, \phi)_i, i = [1, \dots, N_r]$ inside cell c i.e., $\tilde{\mathcal{L}}_{\hat{c}} = \bigcup_{i=1}^{N_r} \{\Lambda : \Lambda \in \mathcal{L}_{\hat{c}}^i(\theta_i, \phi_i, x_i^P)\}$, where $\hat{c} \in [1, \dots, |\tilde{\mathcal{E}}|]$ is the index pointing to cell $c \in \tilde{\mathcal{E}}$, $\mathcal{L}_{\hat{c}}^i(\theta_i, \phi_i, x_i^P)$ is the set of light-rays given by the camera pose obtained with the i_{th} set of control inputs (θ_i, ϕ_i) , for

the agent location $x_i^{\mathbf{P}}$ sampled within cell c . We then learn the following logical visibility determination constraints:

$$\rho_{\hat{c}, \hat{\tau}} = 1 \iff \exists \Lambda \in \tilde{\mathcal{L}}_{\hat{c}} : \Lambda \oplus \mathcal{T} = \tau, \forall \hat{c}, \hat{\tau} \quad (8)$$

Once the constraints above are learned, we can determine the visibility of facet τ , by utilizing the binary variable $\rho_{\hat{c}, \hat{\tau}}$ which is activated when there exists a light-ray Λ which traces back to facet τ when the agent is within the cell c .

B. Coverage Objective Function

Let us assume that each facet $\tau \in \mathcal{T}$ can be uniquely identified by the agent $x_{k+\kappa|k}^j$, and thus its state (i.e., observed/covered or unobserved) can be associated with the binary variable $\hat{b}_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j \in \{0, 1\}$ which indicates that facet τ (indexed by $\hat{\tau} \in [1, \dots, |\mathcal{T}|]$), is planned to be observed and covered by agent j , with the FOV rotation $\xi \in \Xi$ (indexed by $\hat{\xi} \in [1, \dots, |\Xi|]$) at the future time-step $k + \kappa|k, \kappa \in [1, \dots, K]$ of the planning horizon (abbreviated simply as κ throughout the paper). As a reminder the FOV rotation $\xi \in \Xi$ maps to a certain set of rotation angles (θ, ϕ) i.e., $\xi = (\theta, \phi) \in \Theta \times \Phi$ as discussed in Sec. II-B. The cooperative coverage objective function $\mathcal{F}_{\text{mission}}$ to be optimized over the planning horizon $k + \kappa|k, \kappa \in [1, \dots, K]$, shown in Eq. (10a), can now be defined as $\mathcal{F}_{\text{mission}} = \sum_{j=1}^N F^j$, where F^j is agent's j own coverage objective function which is further given by:

$$F^j = \mathcal{D}(x_{k+1|k}^{j, \mathbf{P}}, \tau_{\star}^j) - \sum_{\kappa=1}^K \sigma(\kappa) \sum_{\hat{\xi}=1}^{|\Xi|} \sum_{\hat{\tau}=1}^{|\mathcal{T}|} \hat{b}_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j, \quad (9)$$

which essentially is minimized when all facets $\tau \in \mathcal{T}$ are planned to be covered inside the planning horizon i.e., $\hat{b}_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j = 1, \forall \hat{\tau}$, assuming a sufficiently large enough value of K . Otherwise, F^j incentivizes the agent to generate a trajectory which covers as many facets as possible inside the planning horizon. The time-dependent term $\sigma(\kappa)$ penalizes facets that are covered later in the horizon i.e., $\sigma(\kappa) = K - (\kappa - 1)$ or any other penalty scheme can be used. Finally, the term $\mathcal{D}(x_{k+1|k}^{j, \mathbf{P}}, \tau_{\star}^j) = \|x_{k+1|k}^{j, \mathbf{P}} - \tau_{\star}^j\|_2^2$ drives the agent j towards the centroid of its nearest unobserved facet τ_{\star}^j , in order to make sure that the mission can progress particularly in the events where no facets can be reached for coverage inside the planning horizon.

C. Constraints

The generation of look-ahead trajectories is achieved with the constraints shown in Eq. (10b)-(10c), by appropriately selecting the control inputs $u_{k+\kappa|k}^j$ inside the horizon for all agents, according to the kinematic model as discussed in Sec. II-A. The constraint shown in Eq. (10d) rotates and translates agent's j camera FOV inside the planning horizon. More specifically, $\mathcal{V}_{\hat{\xi}, k+\kappa|k}^j$ denotes the $\hat{\xi}_{\text{th}}$ configuration of the camera's FOV vertices at the future time-step κ . The set of all possible $|\Xi|$ FOV rotations is precomputed as $\tilde{\mathcal{V}}_{\hat{\xi}, k+\kappa|k}^{j, i} = R_z(\phi_{k+\kappa|k}^j) R_y(\theta_{k+\kappa|k}^j) \mathcal{V}_0^i, \forall i \in [1, \dots, 5], \forall \hat{\xi} \in [1, \dots, |\Xi|]$, where $(\theta_{k+\kappa|k}^j) \in \Theta, (\phi_{k+\kappa|k}^j) \in \Phi) \in \Xi$, and then translated to the agent's position $x_{k+\kappa|k}^{j, \mathbf{P}}$ as shown in Eq. (10d). Next, the binary variable $\nu_{\hat{\xi}, k+\kappa|k}^j \in \{0, 1\}$

(P2) Cooperative 3D Coverage

$$\arg \min_{\{u_{k+\kappa|k}^j, \theta_{k+\kappa|k}^j, \phi_{k+\kappa|k}^j\}_{j=1}^N} \mathcal{F}_{\text{mission}}, \quad (10a)$$

subject to: $\kappa \in [1, \dots, K]$

$$x_{k+\kappa|k}^j = A x_{k+\kappa-1|k}^j + B u_{k+\kappa|k}^j \quad \forall \kappa, j \quad (10b)$$

$$x_{k|k}^j = x_{k|k-1}^j \quad \forall \kappa, j \quad (10c)$$

$$\mathcal{V}_{\hat{\xi}, k+\kappa|k}^j = \tilde{\mathcal{V}}_{\hat{\xi}, k+\kappa|k}^j + x_{k+\kappa|k}^{j, \mathbf{P}} \quad \forall \kappa, \hat{\xi}, j \quad (10d)$$

$$\sum_{\hat{\xi} \in [1, \dots, |\Xi|]} \nu_{\hat{\xi}, k+\kappa|k}^j = 1 \quad \forall \kappa, j \quad (10e)$$

$$b_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j = 1 \iff \tau \in \Delta(\mathcal{V}_{\hat{\xi}, k+\kappa|k}^j) \quad \forall \hat{\tau}, \kappa, \hat{\xi}, j \quad (10f)$$

$$\bar{b}_{\tau, \xi, k+\kappa|k}^j = \nu_{\hat{\xi}, k+\kappa|k}^j \wedge (b_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j \wedge \rho_{\hat{c}, \hat{\tau}} \wedge \tilde{\rho}_{\hat{c}}^j) \quad \forall \hat{\tau}, \kappa, \hat{\xi}, \hat{c}, j \quad (10g)$$

$$\hat{b}_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j \leq \bar{b}_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j + \mathcal{Q}(\tau) \quad \forall \hat{\tau}, \kappa, \hat{\xi}, j \quad (10h)$$

$$\sum_j \sum_{\hat{\xi}} \sum_{\kappa} \hat{b}_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j \leq 1 \quad \forall \hat{\tau} \quad (10i)$$

$$x_{k+\kappa|k}^{j, \mathbf{P}} \notin \Delta(\psi) \quad \forall \psi \in \Psi, \kappa, j \quad (10j)$$

$$x_{k+\kappa|k}^{j, \mathbf{P}} \notin \bigcup_{i \neq j=1}^N \Delta(\mathcal{A}^i) \quad \forall \kappa, j \quad (10k)$$

indicates which of the $|\Xi|$ camera FOV configurations is active at time-step κ , to avoid double counting the same facet with different FOV configurations. This is achieved with the constraint shown in Eq. (10e).

To determine whether facet τ resides inside the agent's j , $\hat{\xi}_{\text{th}}$ camera FOV configuration at time-step κ i.e., $\tau \in \Delta(\mathcal{V}_{\hat{\xi}, k+\kappa|k}^j)$ first observe that an arbitrary point $p \in \mathcal{E}$ which belongs to the convex-hull defined by the camera FOV vertices \mathcal{V} satisfies the following system of linear inequalities: $\alpha_n \cdot p \leq \beta_n, \forall n = [1, \dots, 5]$, where $\alpha_n \cdot p = \beta_n$ is the equation of the plane which contains the n_{th} face of the FOV (with 5 faces in total), α_n is the unit outward normal vector to the plane containing the n_{th} FOV face, and β_n is a constant. Any point $p \in \mathcal{E}$ which satisfies the aforementioned system of inequalities is contained within the convex-hull of \mathcal{V} , and therefore can be potentially observed by the agent (provided it is visible). Subsequently, the binary variable $b_{\hat{\tau}, \hat{\xi}, k+\kappa|k}^j \in \{0, 1\}$ shown in Eq. (10f) is activated when facet τ resides inside the $\hat{\xi}_{\text{th}}$ camera FOV configuration of agent j at time-step κ . This functionality can be defined as shown below:

$$\alpha_{n, \hat{\xi}, \kappa}^j \cdot \tau + \sigma_{n, \hat{\tau}, \hat{\xi}, \kappa}^j (M - \beta_{n, \hat{\xi}, \kappa}^j) \leq M, \quad \forall n, \hat{\tau}, \hat{\xi}, \kappa, \quad (11a)$$

$$5b_{\hat{\tau}, \hat{\xi}, \kappa}^j - \sum_{n=1}^5 \sigma_{n, \hat{\tau}, \hat{\xi}, \kappa}^j \leq 0, \quad \forall \kappa, \hat{\xi}, \hat{\tau}. \quad (11b)$$

where $k + \kappa|k$ is abbreviated as κ , $\alpha_{n, \hat{\xi}, \kappa}^j$ and $\beta_{n, \hat{\xi}, \kappa}^j$ are the plane equation coefficients, containing the n_{th} FOV face of the $\hat{\xi}_{\text{th}}$ FOV configuration, of the j agent at time-step κ , $\sigma_{n, \hat{\tau}, \hat{\xi}, \kappa}^j \in \{0, 1\}$ is an auxiliary binary variable which is activated when the n_{th} inequality is true i.e., $\alpha_{n, \hat{\xi}, \kappa}^j \cdot \tau \leq$

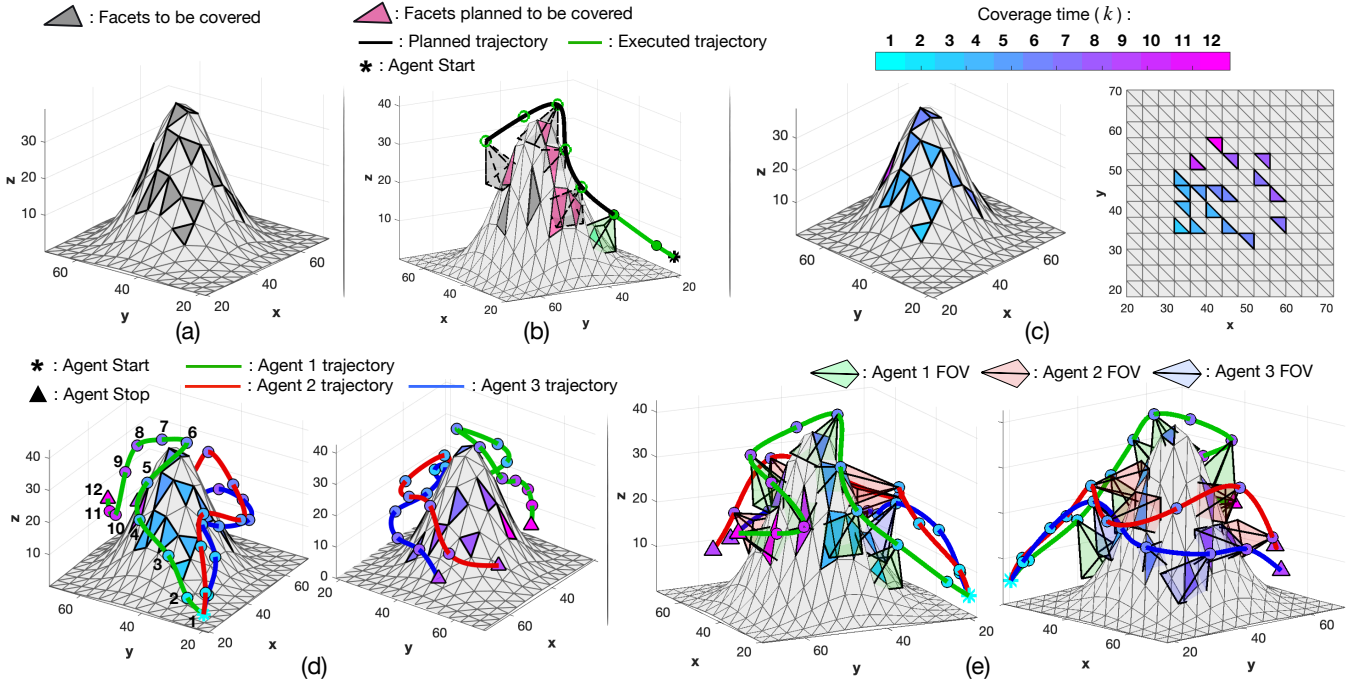


Fig. 2. The figure shows an illustrative example of the proposed cooperative receding horizon 3D coverage controller with a team of 3 agents.

$\beta_{n,\hat{\xi},\kappa}^j$, and M is a large positive constant that ensures the constraint is valid when $o_{n,\hat{\tau},\hat{\xi},\kappa}^j = 0$. Finally, $b_{\hat{\tau},\hat{\xi},\kappa}^j$ is activated to satisfy Eq. (11b) when $o_{n,\hat{\tau},\hat{\xi},\kappa}^j = 1, \forall n$ thus indicating that facet τ resides inside the ξ_{th} camera FOV configuration of agent j . For brevity facet τ is represented by its centroid in this formulation.

The logical conjunction shown in Eq. (10g) makes sure that agent j does not activate more than one camera FOV configurations at each time-step κ , with the binary variable $\nu_{\hat{\xi},k+\kappa|k}^j$. The constraint in Eq. (10g) also checks whether the facet τ which resides inside the agent's FOV is visible via the learned visibility determination variable $\rho_{\hat{c},\hat{\tau}}$. Subsequently, the logical conjunction $\nu_{\hat{\xi},k+\kappa|k}^j \wedge (b_{\hat{\tau},\hat{\xi},k+\kappa|k}^j \wedge \rho_{\hat{c},\hat{\tau}} \wedge \tilde{\rho}_{\hat{c}}^j)$ becomes true when at time-step κ facet τ resides within the convex-hull of the active FOV configuration ξ (as indicated by the variables $\nu_{\hat{\xi},k+\kappa|k}^j$ and $b_{\hat{\tau},\hat{\xi},k+\kappa|k}^j$), and at the same time the agent position $x_{k+\kappa|k}^{j,\mathbf{P}}$ resides within the cell c (as indicated by the binary variable $\tilde{\rho}_{\hat{c}}^j$) from which it has been determined that the facet τ is visible via the learned visibility variable $\rho_{\hat{c},\hat{\tau}}$. The constraint shown in Eq. (10h) is used for avoiding the duplication of work (i.e., avoiding to cover facets that have been covered in the past). To achieve this, the function $Q : \mathcal{T} \rightarrow \{0,1\}$ keeps track of all the facets that have been covered by the agents up to the current time-step k . Therefore any facet τ that has been covered by any agent j results in $Q(\tau) = 1$. Consequently, the binary variable $b_{\hat{\tau},\hat{\xi},k+\kappa|k}^j$ in Eq. (10h) is maximized for facet τ either through $Q(\tau)$ or via $\tilde{b}_{\hat{\tau},\hat{\xi},k+\kappa|k}^j$. For this reason, the agent j has no incentive to plan a coverage trajectory for facet τ inside the planning horizon, when $Q(\tau) = 1$ since the binary variable $b_{\hat{\tau},\hat{\xi},k+\kappa|k}^j$ is maximised through $Q(\tau)$.

The constraint shown in Eq. (10i) makes sure that during

the planning horizon the facet τ is not planned to be covered more than once by the same or multiple agents, and then the constraints shown in Eq. (10j) ensure that the agents avoid collisions with the obstacles $\psi \in \Psi$ in the environment, including the object of interest. This is achieved by enforcing the agent position $x_{k+\kappa|k}^{j,\mathbf{P}}$ to reside outside the convex-hull of all obstacles $\psi \in \Psi$ at all time-steps κ . Assuming that the convex-hull of obstacle ψ , which is represented as a triangle mesh $\mathcal{T}_{\hat{\psi}}$ (where $\hat{\psi} \in [1, \dots, |\Psi|]$ is the index of ψ) is given by the intersection of $n_{\hat{\psi}}$ half-spaces, where the i_{th} half-space is associated with the plane equation $\alpha_i^{\hat{\psi}} \cdot p = \beta_i^{\hat{\psi}}, i \in [1, \dots, n_{\hat{\psi}}], p \in \mathcal{E}$, which divides the 3D space into two parts, the obstacle avoidance constraints for all obstacles can be defined as follows:

$$\alpha_i^{\hat{\psi}} \cdot x_{k+\kappa|k}^{j,\mathbf{P}} + M z_{k+\kappa|k,i}^{j,\hat{\psi}} > \beta_i^{\hat{\psi}}, \forall j, \kappa, i, \hat{\psi}, \quad (12)$$

$$\sum_{i=1}^{n_{\hat{\psi}}} z_{k+\kappa|k,i}^{j,\hat{\psi}} \leq (n_{\hat{\psi}} - 1), \forall j, \kappa, \hat{\psi} \quad (13)$$

where $z_{k+\kappa|k,i}^{j,\hat{\psi}} \in \{0,1\}$ is a binary variable which indicates when activated that $\alpha_i^{\hat{\psi}} \cdot x_{k+\kappa|k}^{j,\mathbf{P}} > \beta_i^{\hat{\psi}}$ is not true. Therefore when $z_{k+\kappa|k,i}^{j,\hat{\psi}} = 1, \forall i \in [1, \dots, n_{\hat{\psi}}]$ indicates that agent j resides within the convex-hull of obstacle ψ at time-step κ . Consequently, a collision is avoided with obstacle ψ at time-step κ when $\exists i \in [1, \dots, n_{\hat{\psi}}] : z_{k+\kappa|k,i}^{j,\hat{\psi}} = 0$ which is achieved with the constraint in Eq. (13). The same principle is applied to implement collision avoidance constraints amongst the team of agents as shown in Eq. (10k), which requires that during all time-steps the agent's j positional state must reside outside the convex-hull of agent's i safety area $\mathcal{A}^i, i \neq j, i \in [1, \dots, N]$, where \mathcal{A}^i is the inscribed dodecahedron around $x_{k+\kappa|k}^{i,\mathbf{P}}$ which approximates a spherical safety area

with certain radius around the agent [20]. Finally, the mission is terminated when $\sum_{\tau \in \mathcal{T}} \mathcal{Q}(\tau) = |\mathcal{T}|$.

V. EVALUATION

A. Simulation Setup

For the evaluation of the proposed approach we assume agents with identical capabilities. Subsequently, the agent's j kinematic model parameters ΔT , γ , and m are set to 1s, 0.2, and 1.05kg respectively. The agent velocity $x_k^{j,v}$ is bounded within the interval $[-12, 12]$ m/s, whereas the kinematic control input u_k^j is bounded within the interval $[-10, 10]$ N. The agent camera FOV model parameters (ℓ, w, r) are set to (10, 10, 16)m, and the gimbal rotation angles θ and ϕ take their values from the finite sets $\Theta = \{30, 90, 150\}$ deg, and $\Phi = \{30, 105, 180, 255, 330\}$ respectively, leading to $|\Xi| = 15$ possible camera FOV configurations. The 3D environment \mathcal{E} is bounded in each dimension in the interval $[0, 100]$ m, and the object of interest to be covered is given by the Gaussian function $f(x, y) = A \exp\left(-\left(\frac{(x-x_o)^2}{2\sigma_x^2} + \frac{(y-y_o)^2}{2\sigma_y^2}\right)\right)$, with $(x_o, y_o) = (45, 45)$, $\sigma_x^2 = \sigma_y^2 = 80$, and $A = 40$, which has been Delaunay triangulated into $|\mathcal{T}| = 220$ triangular facets. For the visibility determination constraints we have used 50 light rays i.e., $|\mathcal{L}_k^j(\theta_k^j, \phi_k^j, x_k^{j,p})| = 50$, N_r was set to 100, and the procedure described in Sec. IV-A was conducted on a discretized version of the environment $\tilde{\mathcal{E}}$, composed of $|\tilde{\mathcal{E}}| = 1000$ non-overlapping 3D cuboid cells. The planning horizon in the following experiments has been set to $K = 5$.

B. Simulation Experiment

An illustrative example of the proposed approach with 3 UAV agents is shown in Fig. 2. Specifically, Fig. 2(a) shows the object of interest to be covered. Without loss of generality, and in order to aid the analysis and visual clarity of this demonstration, we only require that a random subset of facets $\tilde{\mathcal{T}} \subseteq \mathcal{T}$ needs to be covered by the agents, instead of the full triangle mesh \mathcal{T} , thus we randomly sample 18 facets ($|\tilde{\mathcal{T}}| = 18$) as shown in Fig. 2(a) marked with dark gray color. Figure 2(b) shows the controller's output for agent 1 at time-step $k = 3$, with the executed trajectory shown in green color, and the predicted trajectory shown in black. The figure illustrates the generated finite-length look-ahead trajectory (i.e., kinematic and camera states) of agent 1 inside the planning horizon $k + \kappa|k$, $k = 3, \kappa = [1, \dots, 5]$. Then Fig. 2(c) shows the time-steps at which the facets $\tau \in \tilde{\mathcal{T}}$ have been covered by the agents (both in 3D and top-down view), color-coded based on the coverage time. Figure 2(d) shows in detail (front and back view of the object of interest) the kinematic trajectories of the 3 agents during the coverage mission, indicating the time-steps at which the agents cover the object's facets with different color. Finally, Fig. 2(e) shows the camera FOV configurations of each agent used during the coverage mission.

VI. CONCLUSION

We have proposed a cooperative coverage controller for 3D environments which allows a team of networked UAV agents to work cooperatively in order to cover the total surface area of an object of interest. We have formulated the coverage planning problem as a receding horizon optimal

control problem which jointly optimizes the kinematic and camera control inputs over all agents, under duplication of work constraints and visibility determination constraints. Future work will investigate the extension of the proposed approach to a distributed system, and study how uncertainty can be handled using robust control techniques.

REFERENCES

- [1] S. Papaioannou, P. Kolios, and G. Ellinas, "Downing a rogue drone with a team of aerial radio signal jammers," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, pp. 2555–2562.
- [2] S. Papaioannou, P. Kolios, T. Theocharides, C. G. Panayiotou, and M. M. Polycarpou, "Unscented optimal control for 3d coverage planning with an autonomous uav agent," in *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2023, pp. 703–712.
- [3] R. S. de Moraes and E. P. de Freitas, "Multi-UAV based crowd monitoring system," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 56, no. 2, pp. 1332–1345, 2020.
- [4] S. M. S. M. Daud, M. Y. P. M. Yusof, C. C. Heo, L. S. Khoo, M. K. C. Singh, M. S. Mahmood, and H. Nawawi, "Applications of drone in disaster management: A scoping review," *Science & Justice*, vol. 62, no. 1, pp. 30–42, 2022.
- [5] E. Galceran and M. Carreras, "A survey on coverage path planning for robotics," *Robotics and Autonomous Systems*, vol. 61, no. 12, pp. 1258–1276, 2013.
- [6] Y. Li, H. Chen, M. J. Er, and X. Wang, "Coverage path planning for UAVs based on enhanced exact cellular decomposition method," *Mechatronics*, vol. 21, no. 5, pp. 876–885, 2011.
- [7] A. Xu, C. Viriyasuthee, and I. Rekleitis, "Efficient complete coverage of a known arbitrary environment with applications to aerial operations," *Autonomous Robots*, vol. 36, no. 4, pp. 365–381, 2014.
- [8] L. Paull, C. Thibault, A. Nagaty, M. Seto, and H. Li, "Sensor-driven area coverage for an autonomous fixed-wing unmanned aerial vehicle," *IEEE Transactions on Cybernetics*, vol. 44, no. 9, pp. 1605–1618, 2014.
- [9] J. Chen, C. Du, Y. Zhang, P. Han, and W. Wei, "A clustering-based coverage path planning method for autonomous heterogeneous UAVs," *IEEE Transactions on Intelligent Transportation Systems*, 2021.
- [10] M. Theile, H. Bayerlein, R. Nai, D. Gesbert, and M. Caccamo, "UAV coverage path planning under varying power constraints using deep reinforcement learning," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1444–1449.
- [11] L. Collins, P. Ghassemi, E. T. Esfahani, D. Doermann, K. Dantu, and S. Chowdhury, "Scalable coverage path planning of multi-robot teams for monitoring non-convex areas," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7393–7399.
- [12] S. Papaioannou, P. Kolios, T. Theocharides, C. G. Panayiotou, and M. M. Polycarpou, "UAV-based receding horizon control for 3d inspection planning," in *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, 2022, pp. 1121–1130.
- [13] —, "Distributed search planning in 3-d environments with a dynamically varying number of agents," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 7, pp. 4117–4130, 2023.
- [14] —, "Integrated guidance and gimbal control for coverage planning with visibility constraints," *IEEE Transactions on Aerospace and Electronic Systems*, 2022, doi:10.1109/TAES.2022.3199196.
- [15] —, "Integrated ray-tracing and coverage planning control using reinforcement learning," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, 2022, pp. 7200–7207.
- [16] C. S. Tan, R. Mohd-Mokhtar, and M. R. Arshad, "A comprehensive review of coverage path planning in robotics using classical and heuristic algorithms," *IEEE Access*, vol. 9, pp. 119310–119342, 2021.
- [17] T. M. Cabreira, C. Di Franco, P. R. Ferreira, and G. C. Buttazzo, "Energy-aware spiral coverage path planning for UAV photogrammetric applications," *IEEE Robotics and automation letters*, vol. 3, no. 4, pp. 3662–3668, 2018.
- [18] R. Almadhoun, T. Taha, L. Seneviratne, and Y. Zweiri, "A survey on multi-robot coverage path planning for model reconstruction and mapping," *SN Applied Sciences*, vol. 1, no. 8, pp. 1–24, 2019.
- [19] G. Taubin, "3d rotations," *IEEE Computer Graphics and Applications*, vol. 31, no. 6, pp. 84–89, 2011.
- [20] S. Papaioannou, P. Kolios, and G. Ellinas, "Distributed estimation and control for jamming an aerial target with multiple agents, doi: 10.1109/TMC.2022.3207589," *IEEE Transactions on Mobile Computing*, 2022.