



# Multiagent Planning and Control for Swarm Herding in 2-D Obstacle Environments Under Bounded Inputs

Vishnu S. Chipade , *Student Member, IEEE*, and Dimitra Panagou , *Senior Member, IEEE*

**Abstract**—This article presents a method for herding a swarm of adversarial agents toward a safe area in a 2-D obstacle environment. The team of defending agents (defenders) aims to block the path of a swarm of risk-averse, adversarial agents (attackers) and guide it to a safe area while navigating in an obstacle-populated environment. To achieve this, a closed formation (StringNet) of defenders is formed around the adversarial swarm. A combination of open-loop, near time-optimal controllers (that result in forming the defenders' formation), and state-feedback controllers with finite-time convergence guarantees under bounded inputs (that guide the formation around attackers and toward the safe area) synthesize the herding strategy. For demonstration purpose, we consider that the attacking swarm moves under a flocking model, which however is unknown to the defenders. Collision-free trajectory generation for the defenders, as well as their convergence to the desired formations, is proved formally, and simulations are provided to demonstrate the efficacy of the proposed approach. An implementation of the proposed approach on quadrotor vehicles simulated in the Gazebo simulator is also provided.

**Index Terms**—Autonomous agents, cooperative robots, motion and path planning, multirobots systems.

## I. INTRODUCTION

### A. Motivation

**T**HEORY and technology of robotic swarms have seen rapid growth recently. Swarms of ground, marine, or aerial robots are being deployed to accomplish search and rescue missions [1], [2], monitoring and mapping in agricultural [3] and marine [4] environments, and cooperative transportation [5], [6]. For more applications of robotic swarms, refer to the review paper [7].

Nevertheless, ubiquity and rapid advancements of swarm technology pose significant threat to safety-critical infrastructure, such as government facilities, airports, and military bases. The presence of adversarial swarms nearby such entities, with

the aim of causing physical damage or collecting critical information, can lead to catastrophic consequences. This necessitates solutions for the protection of safety-critical infrastructure against such attacks, particularly in crowded urban areas.

Counteracting an adversarial swarm by means of physical interception, as studied in [8]–[10], at low altitudes in an urban environment may not be desirable due to human presence. Under the assumption of risk-averse and self-interested adversarial agents (attackers) that tend to move away from the defending agents (defenders) and from other dynamic objects, *herding* can be used as an indirect way of guiding the attackers to some safe area.

### B. Related Work

Herding has been studied earlier in the literature, see for instance [11]–[15]. The approach in [11] uses an  $n$ -wavefront algorithm to herd a flock of birds away from an airport, where the birds on the boundary of the flock are influenced based on the locations of the airport and a safe area. The framework is extended to include stability and performance guarantees for a bird flock under a directed star communication graph [12] and experiments [13].

The herding method in [14] utilizes a circular-arc formation of herders to influence the nonlinear dynamics of the herd based on a potential-field approach, and designs a point-offset controller to guide the herd close to a specified location. In [15], biologically inspired strategies are developed for confining a group of agents; the authors develop strategies based on the “wall” and “encirclement” methods that dolphins use to capture a school of fish. In addition, they compute regions from which this confinement is possible, but the results are limited to constant-velocity motion. A similar approach called *herding by caging* is adopted in [16], where a cage of high potential is formed around the attackers. A rapidly exploring random tree (RRT) approach is used to find a motion plan for the robots; however, the cage is assumed to have already been formed around the agents, whereas the caging of the agents thereafter is only ensured with constant velocity motion under additional assumptions on the distances between the agents. Forming such a cage could be more challenging in case of self-interested, risk-averse attackers under nonconstant velocity motion.

In [17] and [18], the authors discuss herding using a switched-system approach; the herder (defender) chases targets (evaders/attackers) sequentially by switching among them so that certain dwell-time conditions are satisfied to guarantee

Manuscript received December 23, 2020; accepted March 17, 2021. Date of publication May 3, 2021; date of current version December 6, 2021. This work was supported in part by the Center for Unmanned Aircraft Systems (C-UAS) and in part by the National Science Foundation Industry/University Cooperative Research Center under NSF Award 1738714 along with significant contributions from C-UAS industry members. This article was recommended for publication by Associate Editor M. Schwager and Editor P. Robuffo Giordano upon evaluation of the reviewers' comments. (*Corresponding author: Vishnu S. Chipade.*)

The authors are with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI 48109 USA (e-mail: vishnuc@umich.edu; dpanagou@umich.edu).

Color versions of one or more figures in this article are available at <https://doi.org/10.1109/TRO.2021.3072026>.

Digital Object Identifier 10.1109/TRO.2021.3072026

stability of the resulting trajectories. However, the assumption that only one of the targets is influenced by the herder at any time might be limiting and nonpractical in real applications. Deptula *et al.* [19] use approximate dynamic programming to obtain suboptimal control policies for the herder to chase a target agent to a goal location. A game-theoretic formulation is used in [20] to address the herding problem by constructing a virtual barrier similar to Pierson and Schwager [14], but the computational complexity because of the discretization of the state and control-action space limits its applicability.

Most of the previously discussed studies do not consider obstacles in the environment. Some approaches on pursuit-evasion [21] and shepherding [22] do consider obstacles in the environment, however they use single integrator motion models for the agents that limit their applicability to real systems. In our prior work [23], we developed a vector-field-based strategy for herding a *single* attacker to a safe area while avoiding static rectangular obstacles in the environment. Obstacle avoidance is ensured using vector fields defined around superellipses that contain the obstacles; in fact, superellipse offers a better over-approximation of a rectangle compared to circle or ellipse.

Furthermore, all aforementioned approaches assume some form of potential field to model the repulsive motion of the attackers with respect to the defenders, and develop herding strategies for the defenders based on this potential field. Hence, if the attacker's strategy is not known, such approaches may fail to create proper barriers around the attackers. More recently, in [24], we considered herding strategies for defending a safety-critical area (protected area) from a swarm of attackers in a 2-D environment. We proposed a method termed as "StringNet Herding," in which a closed formation of strings (StringNet) is formed by the defenders to surround the swarm of attackers. It is assumed that the string between two defenders serves as a barrier through which the attackers cannot escape (e.g., a physical straight-line barrier). The StringNet is then controlled to herd the swarm of attackers to a safe area. We provided state-feedback, finite-time control laws for defenders moving under double integrator dynamics with drag term (damped double integrator) to, first, form the StringNet around the attackers, and then, herd the enclosed attackers to the safe area while maintaining the StringNet formation. In contrast to the potential field based herding methods, the "StringNet Herding" approach only assumes that the attackers aim to avoid collisions with the defenders, yet the particular form of the repulsive field or their collision avoidance strategy does not need to be known *a priori*. To demonstrate the proposed approach, we adopt a flocking behavior for the attackers, which however is not known to the defenders.

### C. Overview of the Proposed Approach

The main contribution of this article is advancing the "StringNet Herding" strategy in each of its phases: Gathering, seeking, enclosing, and herding. Recall that the defenders must form the StringNet around the attackers before the attackers reach the protected area.

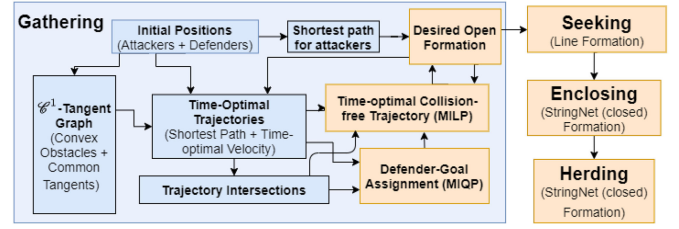


Fig. 1. Overview of the herding approach.

1) *Gathering Phase*: In this phase, compared to the conference version, we design a near time-optimal<sup>1</sup> motion plan for the defenders. Inspired by the work in [25], we compute near time-optimal, collision-free trajectories for the defenders toward desired open-formation positions in the expected (i.e., the shortest) path of the attackers to the protected area. To this end, we use path-velocity decomposition [25]: first, we design shortest paths from the current positions of the defenders to the desired positions that lie on the desired formation. The shortest paths are obtained using a special representation of the environment called  $\mathcal{C}^1$ -tangent graph [26] (inspired from tangent graph [27]). Then, we compute velocity profiles that minimize the time to traverse the shortest paths under bounded acceleration [26]. The desired formation positions and the corresponding shortest paths, along with the near time-optimal velocity profiles, are then assigned to the defenders by solving a mixed integer quadratic program (MIQP), which minimizes the total travel time and total length of path intersections. Collision avoidance on the chosen paths and under the given near time-optimal velocity profiles for the defenders is accomplished by initial-time scheduling, similar to the approach in [28], using a mixed integer linear program (MILP). The desired formation is chosen by iteratively solving the MIQP so that it is as far as possible from the protected area, and the defenders are able to gather at the desired formation before the attackers can reach there.

2) *Seeking Phase*: After the defenders have converged to their desired formation, they start moving closer to the attacking swarm (i.e., as we say, they seek the attackers) while maintaining the desired formation.

3) *Enclosing and Herding*: Once the distance of the formation center to the center-of-mass of the attacking swarm is below a certain threshold, the defenders enclose the attackers by completing the StringNet formation, and herd the attacking swarm to the safe area.

In each phase, we assume that the agents have a known, circular footprint, similar to our work [24], and we explicitly consider the problem of interagent collision avoidance, in contrast to earlier work, for instance [14], [16]. The block diagram in Fig. 1 summarizes the approach.

### D. Summary of Our Contributions

Compared to the prior literature and our earlier work in the conference paper [24], the novelties and the main contributions of this article are as follows.

<sup>1</sup>i.e., the actual travel time  $\tau$  satisfies  $\tau^* \leq \tau \leq (1 + \varepsilon)\tau^*$  where  $\tau^*$  is the optimal travel time and  $\varepsilon \ll 1$  is a small, positive constant.

- 1) We develop near time-optimal controllers to guide the defenders to an optimal desired open formation (in order to later form the StringNet) while avoiding collisions. The contributions compared to the conference version are as follows.
  - 1) The formulation of an MIQP to assign desired formation positions to the defenders, so that the total travel time and the total length of the path intersections on the corresponding trajectories are minimized.
  - 2) An iterative scheme that uses the MIQP formulation to find an optimal desired formation for the defenders to gather in the expected path of the attackers.
- 2) We develop *state-feedback, finite-time convergent, bounded* control laws for the defenders moving under *damped double integrator* dynamics such that their formation seeks, encloses, and herds the attackers to the safe area while avoiding the convex polygonal obstacles. Compared to prior work, we provide explicit guarantees on the time of convergence under the proposed bounded controllers.

### E. Organization

The rest of this article is structured as follows. Section II discusses problem setup and preliminaries. The “StringNet Herding” method is discussed in Section III. A formal safety and convergence analysis is provided in Section IV, followed by MATLAB simulations in Section V-A. The results on implementation of the proposed approach on quadrotor vehicles using physics-based Gazebo simulator are provided in Section V-B. Finally, Section VI concludes this article.

## II. PROBLEM SETUP AND PRELIMINARIES

### A. Notation

We use  $\mathbf{r}$ ,  $\mathbf{v}$ , and  $\mathbf{u}$  to denote position, velocity, and input acceleration vector, respectively. We use  $\xi$  and  $\eta$  to denote desired position and velocity vector, respectively. We use both  $\rho$  and  $\varrho$  to denote radius. The variables  $ai$ ,  $ac$ ,  $dj$ , and  $df$  used as subscripts of the aforementioned variables correspond to the  $i$ th attacker, attackers’ center of mass (ACoM),  $j$ th defender, and the defenders’ desired formation, respectively. Similarly, subscripts  $pa$ ,  $sa$ ,  $ok$ , and  $ct$  denote the protected area, safe area, the  $k$ th obstacle, and the  $\mathcal{C}^1$ -tangent graph, respectively. We use subscript  $d$  to denote common variables that correspond to all the defenders. Similarly, subscripts  $a$  and  $o$  denote common variables corresponding to all the attackers and all the obstacles, respectively. We use  $sn$  and  $sb$  as subscripts to denote StringNet and string barrier, respectively. Any variable with superscript  $g$ ,  $s$ ,  $e$ , and  $h$  correspond to gathering, seeking, enclosing, and herding phase, respectively. We denote by  $\delta_{j,k}$  and  $\delta_{f,k}$  virtual  $\delta$ -agents (defined later) on the obstacle  $\mathcal{O}_k$  corresponding to the  $j$ th defender and the defenders’ formation, respectively. Similarly,  $\delta_{j,c}$  and  $\delta_{j,p}$  denote the  $\delta$ -agents corresponding to the  $j$ th defender on the connectivity region of the attackers and the protected area, respectively. The bar notation ( $\bar{\cdot}$ ) used on a variable denotes some form of upper limit of the corresponding variable. Similarly, underbar notation ( $\underline{\cdot}$ ) denotes some form of

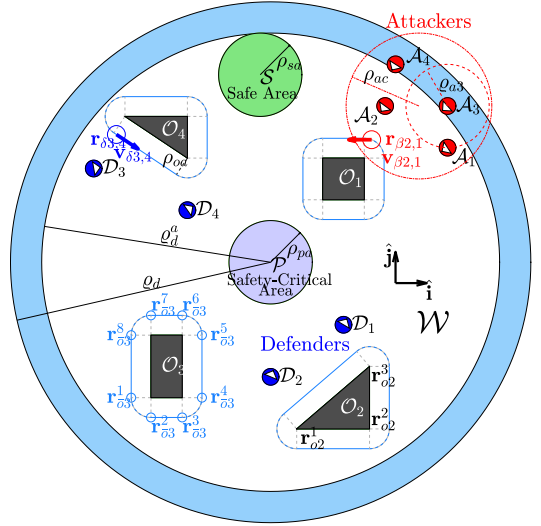


Fig. 2. Problem formulation.

upper limit of the corresponding variable. The tilde notation ( $\tilde{\cdot}$ ) used on a vector variable denotes quantity associated with some form of error.  $\|\cdot\|$  denotes the Euclidean norm of its argument.  $|\cdot|$  denotes absolute value of a scalar argument.

### B. Problem Setup

We consider  $N_a$  attackers denoted as  $\mathcal{A}_i$ ,  $i \in I_a = \{1, 2, \dots, N_a\}$ , and  $N_d$  defenders denoted as  $\mathcal{D}_j$ ,  $j \in I_d = \{1, 2, \dots, N_d\}$ , operating in an environment  $\mathcal{W} \subseteq \mathbb{R}^2$  with a protected area  $\mathcal{P} \subset \mathcal{W}$  defined as  $\mathcal{P} = \{\mathbf{r} \in \mathbb{R}^2 \mid \|\mathbf{r} - \mathbf{r}_{pa}\| \leq \rho_{pa}\}$ , and a safe area  $\mathcal{S}$  defined as  $\mathcal{S} = \{\mathbf{r} \in \mathbb{R}^2 \mid \|\mathbf{r} - \mathbf{r}_{sa}\| \leq \rho_{sa}\}$ , where  $(\mathbf{r}_{pa}, \rho_{pa})$  and  $(\mathbf{r}_{sa}, \rho_{sa})$  are the centers and radii of the protected and safe area, respectively. The agents  $\mathcal{A}_i$  and  $\mathcal{D}_j$  are modeled as discs of radii  $\rho_a > 0$  and  $\rho_d > 0$ , for all  $i \in I_a$  and  $j \in I_d$ , respectively, and move under double integrator dynamics with a quadratic drag term (damped double integrator)

$$\dot{\mathbf{r}}_{ai} = \mathbf{v}_{ai}, \quad \dot{\mathbf{v}}_{ai} = \mathbf{u}_{ai} - C_D \|\mathbf{v}_{ai}\| \mathbf{v}_{ai} \quad (1)$$

$$\dot{\mathbf{r}}_{dj} = \mathbf{v}_{dj}, \quad \dot{\mathbf{v}}_{dj} = \mathbf{u}_{dj} - C_D \|\mathbf{v}_{dj}\| \mathbf{v}_{dj} \quad (2)$$

$$\|\mathbf{u}_{ai}\| \leq \bar{u}_a, \quad \|\mathbf{u}_{dj}\| \leq \bar{u}_d \quad (3)$$

where  $C_D > 0$  is the known, constant drag coefficient,  $\mathbf{r}_{ai} = [x_{ai} \ y_{ai}]^T$  and  $\mathbf{r}_{dj} = [x_{dj} \ y_{dj}]^T$  are the position vectors of  $\mathcal{A}_i$  and  $\mathcal{D}_j$ , respectively;  $\mathbf{v}_{ai} = [v_{x_{ai}} \ v_{y_{ai}}]^T$  and  $\mathbf{v}_{dj} = [v_{x_{dj}} \ v_{y_{dj}}]^T$  are the velocity vectors, respectively, and  $\mathbf{u}_{ai} = [u_{x_{ai}} \ u_{y_{ai}}]^T$ ,  $\mathbf{u}_{dj} = [u_{x_{dj}} \ u_{y_{dj}}]^T$  are the accelerations, which serve also as the control inputs, respectively, all resolved in a global inertial frame  $\mathcal{F}_{gi}(\hat{\mathbf{i}}, \hat{\mathbf{j}})$  (see Fig. 2). The accelerations  $\mathbf{u}_{ai}$  and  $\mathbf{u}_{dj}$  are bounded by  $\bar{u}_a$  and  $\bar{u}_d$  as given in (3) such that  $\bar{u}_a < \bar{u}_d$ . The dynamics in (1) and (2) takes into account the air drag experienced by the agents modeled as a quadratic function of the velocity. Note also that the damped double integrator model inherently poses a speed bound on each agent under a limited acceleration control, i.e.,  $\|\mathbf{v}_{ai}\| < \bar{v}_a = \sqrt{\frac{\bar{u}_a}{C_D}}$  and  $\|\mathbf{v}_{dj}\| < \bar{v}_d = \sqrt{\frac{\bar{u}_d}{C_D}}$ , and does not require an explicit constraint on the velocity of the agents



while designing bounded controllers, as in earlier literature. We also consider the following.

**Assumption 1:** There is a navigation system that senses the position  $\mathbf{r}_{ai}$  and velocity  $\mathbf{v}_{ai}$  of the attacker  $\mathcal{A}_i$  that lies inside a circular sensing zone  $\mathcal{Z}_d = \{\mathbf{r} \in \mathbb{R}^2 \mid \|\mathbf{r} - \mathbf{r}_{pa}\| \leq \varrho_d\}$  for all  $i \in I_a$ , where  $\varrho_d > 0$  is the radius of the defenders' sensing zone. The navigation system communicates the sensed information to the defenders  $\mathcal{D}_j$ , for all  $j \in I_d$ . Every attacker  $\mathcal{A}_i$  has a local sensing zone  $\mathcal{Z}_{ai} = \{\mathbf{r} \in \mathbb{R}^2 \mid \|\mathbf{r} - \mathbf{r}_{ai}\| \leq \varrho_{ai}\}$ , where  $\varrho_{ai} > 0$  is the radius of the attacker  $\mathcal{A}_i$ 's sensing zone (see Fig. 2).

This navigation system can include sensors, such as radars, lidars, and cameras, which are spatially distributed around the protected area and provide measurements of positions and velocities of the attackers and the defenders.

We consider  $N_o$  static, convex polygonal obstacles denoted as  $\mathcal{O}_k, k \in I_o = \{1, 2, \dots, N_o\}$ , (grey colored polygons in Fig. 2), described as the convex hull of their vertices

$$\mathcal{O}_k = \text{Conv}(\{\mathbf{r}_{ok}^1, \mathbf{r}_{ok}^2, \dots, \mathbf{r}_{ok}^{M_k}\}) \quad (4)$$

where  $\text{Conv}(Q)$  is the convex hull of the points given in the set  $Q$ ,  $\mathbf{r}_{ok}^\ell = [x_{ok}^\ell, y_{ok}^\ell]^T$  are the positions of the vertices for all  $\ell \in \{1, 2, \dots, M_k\}$ ,  $M_k$  is the total number of vertices of  $\mathcal{O}_k$ ,  $k \in I_o$ . The boundary of  $\mathcal{O}_k$  is denoted by  $\partial\mathcal{O}_k$ . Inspired from Hegde and Panagou [29] and Esquivel and Chiang [30], the boundaries  $\partial\mathcal{O}_k$  are inflated by a size of  $\rho_{\bar{o}}$  ( $> \rho_d$ ) to account for safety and agent size. The inflated obstacles are denoted by  $\bar{\mathcal{O}}_k$ , and are given as (see Fig. 2):  $\bar{\mathcal{O}}_k = \mathcal{O}_k \oplus B(\rho_{\bar{o}})$ , where  $\oplus$  denotes the Minkowski sum of the sets and  $B(\rho_{\bar{o}})$  denotes a ball of radius  $\rho_{\bar{o}}$  centered at the origin. The boundary  $\partial\bar{\mathcal{O}}_k$  of the inflated obstacle  $\bar{\mathcal{O}}_k$  is a  $C^1$  curve for all  $\rho_{\bar{o}} > 0$ .

The attackers aim to reach the protected area  $\mathcal{P}$ . They are assumed to stay within a connectivity region of radius  $\rho_{ac}$  ( $< \bar{\rho}_{sn}$ ) centered at the ACoM, see Fig. 2, where  $\bar{\rho}_{sn}$  is the radius of the maximum circular footprint of a formation that can pass through the obstacle-free space in the environment. The defenders aim to herd the attackers to the safe area  $\mathcal{S}$  before they reach  $\mathcal{P}$ . Formally, we consider the following.

**Problem 1 (Herdin):** Design control actions  $\mathbf{u}_{dj} \forall j \in I_d$ , such that: the defenders form a "StringNet" formation around the flock of attackers in finite time, and the StringNet formation herds the flock of attackers to the safe area  $\mathcal{S}$  while avoiding the obstacles  $\mathcal{O}_k$ .

### C. Preliminaries

We define  $\text{sig}^\alpha(\mathbf{x}) = \mathbf{x}/\|\mathbf{x}\|^{\alpha-1}$ . The Euclidean distance between agent  $i$  and  $j$  is denoted as  $R_i^j = \|\mathbf{r}_i - \mathbf{r}_j\|$ . A blending function  $\sigma_i^j : [0, \infty) \rightarrow [0, 1]$  [29], characterized by a user defined doublet  $(\underline{R}_i^j, \bar{R}_i^j)$  with  $0 < \underline{R}_i^j < \bar{R}_i^j$ , is defined in (5) as a function of the distance  $R_i^j$

$$\sigma_i^j(R_i^j) = \begin{cases} 1, & R_i^j \leq \underline{R}_i^j \\ \sum_{l=0}^3 A_{i,l}^j (R_i^j)^l, & \underline{R}_i^j \leq R_i^j \leq \bar{R}_i^j \\ 0, & \bar{R}_i^j \leq R_i^j \end{cases} \quad (5)$$

where the coefficients  $A_{i,3}^j, A_{i,2}^j, A_{i,1}^j$ , and  $A_{i,0}^j$  are chosen as in [24] so that  $\sigma_i^j$  in (5) is a  $C^1$  function. The blending function

$\sigma_i^j(R_i^j)$  is used to keep certain terms in the controller of agent  $i$  active only in a local neighborhood around the agent/object  $j$ . The user-defined parameter  $\bar{R}_i^j$  specifies the maximum distance below which the blending function is nonzero and, hence, the controller term multiplied to it is active only within the distance  $\bar{R}_i^j$  from object  $j$ . The parameter  $\underline{R}_i^j < \bar{R}_i^j$  is chosen to allow smooth transition from value 0 outside the circle of radius  $\bar{R}_i^j$  to value 1 inside the circle of radius  $\underline{R}_i^j$  centered at the object  $j$ . The argument  $R_i^j$  of the blending function would be omitted whenever clear from the context. A saturation function  $\Omega_{\bar{u}} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$  is defined as

$$\Omega_{\bar{u}}(\mathbf{g}) = \min(\|\mathbf{g}\|, \bar{u})\mathbf{g} \|\mathbf{g}\|^{-1} \quad (6)$$

where  $\bar{u} > 0$  is the saturation limit. We define potential function as follows.

**Definition 1 (Potential Function [31]):** The potential  $W_i^j$  is a continuously differentiable, nonnegative function of the distance  $R_i^j$  between agents  $i$  and  $j$ , such that:  $W_i^j(R_i^j) \rightarrow \infty$  as  $R_i^j \rightarrow \underline{R}_i^{j,\infty}$  and  $R_i^j \rightarrow \bar{R}_i^{j,\infty}$ , and  $W_i^j$  attains its unique minimum when agents  $i$  and  $j$  are located at a desired distance  $\check{R}_i^j$ . Here,  $\underline{R}_i^{j,\infty}$ ,  $\bar{R}_i^{j,\infty}$ , and  $\check{R}_i^j$  are positive numbers such that  $(\underline{R}_i^{j,\infty} < \check{R}_i^j < \bar{R}_i^{j,\infty})$ .

We choose a potential function  $W_i^j : \mathcal{I}_i^j \rightarrow [0, \infty)$  as

$$W_i^j(R_i^j) = \ln \left( \frac{w_1 R_i^j - w_2}{R_i^j - w_0} + \frac{R_i^j - w_0}{w_1 R_i^j - w_2} \right) \quad (7)$$

where  $\mathcal{I}_i^j = (\underline{R}_i^{j,\infty}, \bar{R}_i^{j,\infty})$ ,  $w_0 = \underline{R}_i^{j,\infty}$ ,  $w_1 = \frac{\check{R}_i^j - \underline{R}_i^{j,\infty}}{\bar{R}_i^{j,\infty} - \check{R}_i^j} > 0$ , and  $w_2 = w_1 \bar{R}_i^{j,\infty}$ , with  $\check{R}_i^j \in \mathcal{I}_i^j$  being the desired distance between agent  $i$  and agent  $j$ . Note that as  $R_i^j \rightarrow \underline{R}_i^{j,\infty}$  or  $R_i^j \rightarrow \bar{R}_i^{j,\infty}$ , the value of the potential  $W_i^j \rightarrow \infty$ . We choose this form because it serves two purposes: first, to generate collision avoidance control for the defenders, and second, to demonstrate flocking motion of the attackers (discussed later) because  $W_i^j(R_i^j)$  has a unique minimum. The derivative of  $W_i^j$  is defined as

$$\frac{\partial W_i^j}{\partial R_i^j} = \frac{-(w_2 - w_1 w_0) K_1}{(w_0 - R_i^j)(w_1 R_i^j - w_2) K_2}$$

where  $K_1 = (w_1^2 - 1)(R_i^j)^2 - (2w_1 w_2 - 2w_0)R_i^j + w_2^2 - w_0^2$  and  $K_2 = (w_1^2 + 1)(R_i^j)^2 - (2w_1 w_2 + 2w_0)R_i^j + w_2^2 + w_0^2$ .

A control action used by agent  $i$  to avoid collision with agent  $j$  based on the potential function  $W_i^j$  is defined as

$$\mathbf{u}_p(\mathbf{x}_i^j) = -\zeta_i^j(\mathbf{v}_i - \mathbf{v}_j) - \mu_i^j \frac{\mathbf{r}_i - \mathbf{r}_j}{R_i^j} \begin{cases} \frac{\partial W_i^j}{\partial R_i^j}, & \text{if } R_i^j \in \bar{\mathcal{I}}_i^j \\ -\nu, & \text{if } R_i^j < \underline{R}_i^{j,\nu} \\ \nu, & \text{if } R_i^j > \bar{R}_i^{j,\nu} \end{cases} \quad (8)$$

where the joint state vector  $\mathbf{x}_i^j = [\mathbf{r}_i^T, \mathbf{v}_i^T, \mathbf{r}_j^T, \mathbf{v}_j^T]^T$ ,  $\zeta_i^j$  and  $\mu_i^j$  are positive control gains,  $\nu$  is a very large number;  $\underline{R}_i^{j,\nu}$ ,  $\bar{R}_i^{j,\nu} \in \mathcal{I}_i^j$  such that  $\frac{\partial W_i^j}{\partial R_i^j}|_{\underline{R}_i^{j,\nu}} = -\nu$ ,  $\frac{\partial W_i^j}{\partial R_i^j}|_{\bar{R}_i^{j,\nu}} = \nu$  and  $\underline{R}_i^{j,\nu} < \bar{R}_i^{j,\nu}$ ;  $\bar{\mathcal{I}}_i^j = [\underline{R}_i^{j,\nu}, \bar{R}_i^{j,\nu}]$ .

We consider a flocking motion model for the attackers for demonstration purposes. In our previous work [24], we developed a bounded controller that yields flocking behavior for the attackers in the presence of rectangular obstacles. For collision

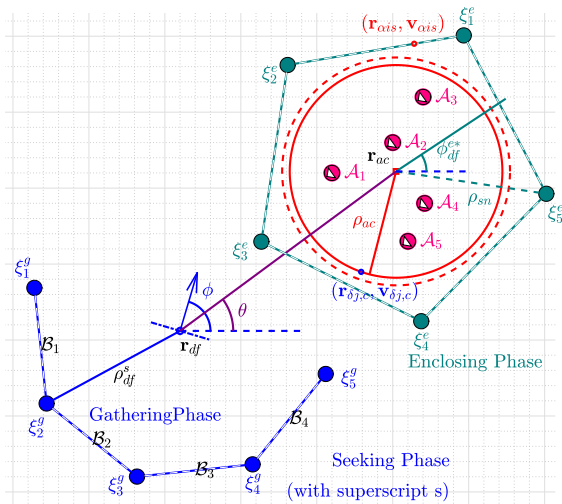


Fig. 3. Desired positions of the defenders.

avoidance with the obstacles during flocking, an attacker considers a virtual  $\beta$ -agent<sup>2</sup> on the superelliptic boundaries around a rectangular obstacle, and uses a potential function based control, similar to that in (8), corresponding to the  $\beta$ -agent. The  $\beta$ -agent strategy can be extended for avoiding convex polygonal obstacles  $\mathcal{O}_k$  with  $\mathcal{C}^1$  boundaries  $\partial\bar{\mathcal{O}}_k$  (see Fig. 2), refer to Chipade and Panagou [24] and references therein for more details on the flocking controller. Next, we describe “StringNet Herding.”

### III. STRINGNET HERDING

To herd the flock of attackers to the safe area  $S$ , we build upon the “StringNet Herding” method proposed in our earlier work [24]. StringNet is a closed net of connections, called strings, formed by the defenders, as shown in Fig. 3. The strings can be thought of as physical mechanisms through which the attackers cannot pass once the defenders become connected through the strings; for example, these could be extendable physical strings or bars connecting the defenders. A closely related practical implementation of a similar concept can be found in [32], where a large rope encircled on a pulley attached to a controlled motor is used to build a bridge. This string barrier (connection) between two defenders can have a maximum length of  $\bar{R}_{sb} > 0$  and is assumed to be a straight line.

The underlying graph for the “StringNet” is defined as follows.

*Definition 2 (StringNet):* The StringNet  $\mathcal{G}_{sn} = (\mathcal{V}_{sn}, \mathcal{E}_{sn})$  is a cycle graph consisting of: the defenders as the vertices,  $\mathcal{V}_{sn} = \{\mathcal{D}_1, \mathcal{D}_2, \dots, \mathcal{D}_{N_d}\}$ , and a set of edges,  $\mathcal{E}_{sn} = \{(\mathcal{D}_j, \mathcal{D}_{j'}) \in \mathcal{V}_{sn} \times \mathcal{V}_{sn} | \mathcal{D}_j \xleftrightarrow{s} \mathcal{D}_{j'}\}$ , where the operator  $\xleftrightarrow{s}$  denotes a string connection between the defenders.

$^2\beta$ -agent is a virtual agent located at the projection point of an attacker's position on the boundary around the obstacle. The  $\beta$ -agent moves along the boundary and its velocity is equal to the projection of the attacker's velocity on the unit tangent vector to the boundary at the current location of the  $\beta$ -agent.

The StringNet Herding consists of four phases: Gathering, seeking, enclosing: StringNet formation, and herding. These phases are discussed as follows.

### A. Gathering

The defenders may be initially scattered throughout the workspace. Once the attackers are sensed in the sensing zone  $\mathcal{Z}_d$ , the defenders are tasked to herd them away from the protected area  $\mathcal{P}$  toward the safe area  $\mathcal{S}$ . The gathering is initiated when at least one of the attackers detected in  $\mathcal{Z}_d$  enters a circular region of radius  $\varrho_d^a$  ( $< \varrho_d$ ) (see Fig. 2). All the attackers that are detected inside the annular region between the circles of radius  $\varrho_d^a$  and  $\varrho_d$  centered at  $\mathbf{r}_{pa}$  are to be herded by the defenders.

It is important for the defenders to form the StringNet around the attackers before the attackers reach the protected area. In the gathering phase, the aim of the defenders is to converge to an open formation  $\mathcal{F}_d^g$  on the expected path of the attackers (i.e., the shortest path of the attackers to the protected area) before the attackers reach there. The open formation  $\mathcal{F}_d^g$  is characterized by the positions  $\xi_l^g \quad \forall l \in I_d$ , as shown in Fig. 3. Once the defenders arrive at these positions, the defenders get connected by strings in the following sense: the defender at  $\xi_l^g$  gets connected to the defender at  $\xi_{l+1}^g$  for all  $l = \{1, 2, \dots, N_d - 1\}$  (see Fig. 3, blue formation). The open formation is formed such that the normal to the line joining  $\xi_1^g$  and  $\xi_{N_d}^g$  (which we refer to as the orientation vector of the formation  $\mathcal{F}_d^g$ ) faces the attackers on their expected path, see for example the blue formation in Fig. 3. The design of  $\mathcal{F}_d^g$  and the corresponding positions  $\xi_l^g \quad \forall l \in I_d$  is discussed later in the section.

In order for the defenders to converge to the formation  $\mathcal{F}_d^g$  as early as possible, we design a near time-optimal motion plan. Given initial positions for the  $N_d$  defenders, and desired goal positions on the formation  $\mathcal{F}_d^g$ , the objective is to find defender-goal pairings and corresponding near time-optimal, collision-free trajectories, which the defenders follow to reach their assigned goals. The solution to this problem consists of the following steps.

- 1) Finding near time-optimal trajectories for all the pairs of the defenders' initial positions and the desired positions on  $\mathcal{F}_d^g$ .
- 2) Finding optimal pairings among the defenders and the desired positions on  $\mathcal{F}_d^g$  to minimize total travel time and total length of the portions of the chosen trajectories that have a possible collision.
- 3) Modifying the derived trajectories to avoid collisions.

The near time-optimal trajectory between two given points is obtained by path-velocity decomposition [25], which consists of finding the shortest path and a near time-optimal velocity profile along the shortest path.

Additionally, we want to find the location of a desired formation that is as far as possible from the protected area, at which the defenders are able to gather before the attackers reach there. This problem becomes computationally intractable as the number of defenders increases.

To reduce the computational requirements, we generate shortest paths between *a priori* known grid points offline. We discretize the obstacle free space  $\mathcal{W}_{\text{free}}$  into a grid of square size  $\Delta w$ . We then define a  $\mathcal{C}^1$ -tangent graph  $\mathcal{G}_{\text{ct}} = \{\mathcal{V}_{\text{ct}}, \mathcal{E}_{\text{ct}}\}$  [26] using the grid points and the obstacles. The  $\mathcal{C}^1$ -tangent graph is inspired from the idea of tangent graph [27]. The  $\mathcal{C}^1$ -tangent graph consists of  $\mathcal{C}^1$  paths between any two nodes. For an obstacle environment with convex polygonal obstacles  $\mathcal{O}_k$ , the construction of the  $\mathcal{C}^1$ -tangent graph involves finding common tangents of each pair of boundaries  $\partial\bar{\mathcal{O}}_k$  and tangents to these boundaries from all the grid points. The points at which these tangents touch the boundaries  $\partial\bar{\mathcal{O}}_k$  and the grid points serve as the nodes on  $\mathcal{G}_{\text{ct}}$ , and the tangents serve as the edges on  $\mathcal{G}_{\text{ct}}$ . More details on  $\mathcal{C}^1$ -tangent can be found in [26]. The shortest paths joining these nodes and the near time-optimal velocity profiles over these paths are also obtained assuming terminal speeds to be zero<sup>3</sup> and considering the dynamics in (2) and (3) (refer to Chipade and Panagou [26]).

The case of multiple defenders traveling along their near-optimal trajectories may give rise to collisions. Since the defenders have finite size, finding trajectory intersections involves checking the intersections of the tubes that would be traversed by the defenders. Any collision between two defenders following given velocity profiles can only happen on the intersecting segments of the tubes over the respective paths. Finding intersecting segments of given two shortest paths involves checking intersections of all the tubes corresponding to the straight-line and circular segments on the two paths. The intersection can be found in  $O(N_{\mathbf{P}_1} N_{\mathbf{P}_2})$  time, where  $N_{\mathbf{P}_1}$  and  $N_{\mathbf{P}_2}$  are the number of segments on any two paths  $\mathbf{P}_1$  and  $\mathbf{P}_2$ . We have the following lemma about the tube intersections.

**Lemma 1 (Lemma 3 in [26]):** Let  $\mathbf{P}_1$  be the shortest path between the points  $\mathbf{r}_{11}$  and  $\mathbf{r}_{12}$ , and  $\mathbf{P}_2$  be the shortest path between the points  $\mathbf{r}_{21}$  and  $\mathbf{r}_{22}$  obtained using  $\mathcal{C}^1$ -tangent graph. Then,  $\mathbf{P}_1$  and  $\mathbf{P}_2$  intersect at most once.

To find the trajectory intersections, we discretize the intersecting segments of the two paths, and perform a pairwise check on these discrete locations for a possible collision between the two agents moving with specified velocity profiles. This yields the collision time interval for the two agents under the specified velocity profiles and the corresponding lengths of the intersecting segments.

In summary, for the  $\mathcal{C}^1$ -tangent graph  $\mathcal{G}_{\text{ct}}$  based on the discretized grid, we offline compute the shortest paths between all pairs of the nodes, velocity profiles on these paths, and trajectory intersections for each pair of trajectories. The worst-case memory requirement for all the aforementioned data for an environment with  $N_o$  obstacles and  $N^2$  grid points is  $O(\bar{N}^3)$ , where  $\bar{N} = 4(N_o^2 - N_o + N^2 N_o)$ .

<sup>3</sup>The final speed is required to be zero for the defenders to get connected via strings. The initial speed being zero is not a conservative assumption because if a defender has nonzero speed, one can apply acceleration opposite to its velocity to make the speed zero and assume the initial position for that defender to be the position at which this speed will become zero.

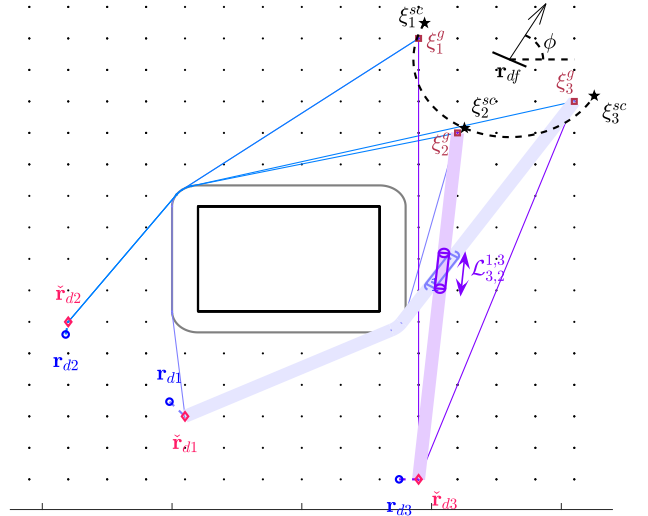


Fig. 4. Grid for motion planning in gathering phase.

We find an approximate shortest path between any two points by connecting them to the closest node on the offline computed  $\mathcal{C}^1$ -tangent graph  $\mathcal{G}_{\text{ct}}$  (see Fig. 4).

In the next sections, we discuss how the defenders are assigned to the goal locations on a given formation  $\mathcal{F}_d^g$  and how the desired formation  $\mathcal{F}_d^g$  is chosen.

1) *Defender-Goal Pairing Assignment:* For a given formation  $\mathcal{F}_d^g$ , we find the near time-optimal trajectories for all possible defender-goal pairs and intersections of all the pairs of the obtained time-optimal trajectories from the offline stored data.

We formulate an MIQP to find the defender-goal assignment that minimizes the total time of travel for all the defenders and the total length of the intersecting segments of the intersecting trajectories. This is to ensure that minimum alterations are required in the assigned trajectories for collision avoidance (discussed later).

Let  $\Xi_{jl}$  be the decision variable whose value is 1 if the trajectory between the initial position of defender  $\mathcal{D}_j$ ,  $\mathbf{r}_{dj}(0)$ , and the goal  $\xi_l^g$  is chosen in the defender-goal assignment, i.e.,  $\xi_l^g$  is assigned to  $\mathcal{D}_j$ , and 0 otherwise for all  $j, l \in I_d$ . Let  $\mathcal{L}_{j,l}^{j',l'}$  be the length of the path along the trajectory between  $\mathbf{r}_{dj}(0)$  and  $\xi_l^g$  that has possible collision with the trajectory between  $\mathbf{r}_{dj'}(0)$  and  $\xi_{l'}^g$  (see Fig. 4). Let  $\mathcal{T}_{dj}^g$  denote the total time required by  $\mathcal{D}_j$  to travel the near time-optimal trajectory corresponding to  $\xi_l^g$  without considering collision avoidance with other defenders. The MIQP is formulated as

$$\begin{aligned}
 &\text{Minimize} \quad \sum_{j,l \in I_d} \mathcal{T}_{dj}^g \Xi_{jl} + \sum_{j,j',l,l' \in I_d} \mathcal{L}_{j,l}^{j',l'} \Xi_{jl} \Xi_{j'l'} \\
 &\text{Subject to} \quad \sum_{l \in I_d} \Xi_{jl} = 1 \quad \forall j \in I_d \\
 &\quad \sum_{j \in I_d} \Xi_{jl} = 1 \quad \forall l \in I_d \\
 &\quad \Xi_{jl}, \Xi_{j'l'} \in \{0, 1\} \quad \forall j, l, j', l' \in I_d
 \end{aligned} \tag{9}$$



where the cost is the total time of travel for all the defenders and the total length of the intersecting segments of the intersecting trajectories. The first constraint ensures that each defender is only assigned to a single desired position and the second constraint ensures that each desired position is only assigned to a single defender. This MIQP can be solved using, for instance, the off-the-shelf MIP solver GUROBI [33]. The solution  $\Xi^* = \{\Xi_{11}^*, \Xi_{12}^*, \dots, \Xi_{j1}^*, \Xi_{j2}^*, \dots, \Xi_{N_d(N_d-1)}^*, \Xi_{N_d N_d}^*\}$  to this MIQP gives the defender-goal assignment. We define a bijective mapping  $a : I_d \rightarrow I_d$

$$a(j) = \arg \max_{l \in I_d} \Xi_{jl}^* \quad (10)$$

such that the desired position  $\xi_{a(j)}^g$  is assigned to  $\mathcal{D}_j$ .

2) *Collision Avoidance by Start-Time Scheduling*: Defenders may collide with other defenders on their assigned trajectories. To avoid this case, we adopt the MILP formulation developed in [28], which performs initial-time scheduling so that the agents start their motion at different initial times, and move on the given paths with specified velocity profiles to avoid collisions. In [34], it is established that the given time-optimal trajectories of multiple robots without considering collision avoidance remain time-optimal under initial time-scheduling if Lemma 1 holds. In our case, the trajectories would still remain near time-optimal after initial-time scheduling.

3) *Desired Open Formation*: So far, we discussed how to obtain a near time-optimal motion plan for the defenders to gather at the given desired formation  $\mathcal{F}_d^g$ . Next, we discuss how to choose  $\mathcal{F}_d^g$  to ensure that the attackers are enclosed inside a StringNet formation (discussed later) provided it is possible, i.e., the attackers have not reached the protected area  $\mathcal{P}$ , and can be herded to a safe area thereafter.

For a given center  $\mathbf{r}_{df}$  and orientation  $\phi$ , consider a semicircular formation  $\mathcal{F}_d^{sc}$  formed by the points  $\xi_l^{sc}$

$$\xi_l^{sc}(\mathbf{r}_{df}, \phi) = \mathbf{r}_{df} + \rho_{df}^g \hat{\mathbf{o}}(\phi_l^g) \quad (11)$$

where  $\phi_l^g = \phi + \frac{\pi}{2} + \frac{\pi(l-1)}{N_d-1}$ , for all  $l \in I_d$ , where  $\hat{\mathbf{o}}(\phi_l^g) = [\cos(\phi_l^g), \sin(\phi_l^g)]^T$  is the unit vector of orientation  $\phi_l^g$  with respect to the  $x$ -axis. We choose the desired positions for the defenders  $\xi_l^g, l \in I_d$ , as the points on the grid that are closest to the points  $\xi_l^{sc}$  given by (11)

$$\xi_l^g = \mathbf{f}_l^g(\mathbf{r}_{df}, \phi) \triangleq \min_{q \in \mathcal{V}_{ct}^{\text{free}}} \|\xi_l^{sc}(\mathbf{r}_{df}, \phi) - \mathbf{r}_q\| \quad (12)$$

where  $\mathcal{V}_{ct}^{\text{free}} \subset \mathcal{V}_{ct}$  contains the nodes that are not on the obstacle boundaries (see Fig. 4). The radius  $\rho_{df}^g$  satisfies  $\rho_{df}^g \sqrt{2 - 2\cos(\frac{\pi}{N_d-1})} > \sqrt{2}\Delta w$  to ensure that there is a unique grid point that is the closest to  $\xi_l^{sc}$ . Similarly, we select the closest grid points  $\check{\mathbf{r}}_{dj}(0)$  to the initial positions of the defenders  $\mathbf{r}_{dj}(0)$  and select the desired open formation as discussed below.

We first calculate the shortest path for the ACoM,  $\mathbf{r}_{ac} = \frac{1}{N_a} \sum_{i=1}^{N_a} \mathbf{r}_{ai}$ , to the protected area. Let this path be  $\mathbf{P}_{ac}$  and its length be  $\Gamma_{ac}$ . The path  $\mathbf{P}_{ac}$  is associated with mappings  $\mathcal{P}_{ac} : [0, \Gamma_{ac}] \rightarrow \mathbb{R}^2$  and  $\vartheta_{ac} : [0, \Gamma_{ac}] \rightarrow [0, 2\pi]$ . Here,  $\mathcal{P}_{ac}(\gamma_{ac})$  gives the Cartesian coordinates, and  $\vartheta_{ac}(\gamma_{ac})$  gives the direction

---

**Algorithm 1: Desired Open Formation.**


---

**Input:**  $\check{\mathbf{r}}_d(0), \mathbf{r}_a(0) = \{\mathbf{r}_{a1}(0), \mathbf{r}_{a2}(0), \dots, \mathbf{r}_{aN_a}(0)\},$   
 $\mathbf{r}_{pa}, \bar{\mathcal{O}} = \{\mathcal{O}_k | k \in I_o\}, \mathcal{G}_{ct}$   
1: Find ACoM:  $\mathbf{r}_{ac}(0) = \sum_{i=1}^{N_a} \frac{\mathbf{r}_{ai}(0)}{N_a};$   
2:  $\mathbf{P}_{ac} = \text{shortestPath}(\bar{\mathcal{O}}, \mathcal{G}_{ct}, \mathbf{r}_{ac}(0), \mathbf{r}_{pa});$   
3:  $T_{lead}(\gamma_{ac}) = \frac{\gamma_{ac}}{\bar{v}_{ac}} - \mathcal{T}(\check{\mathbf{r}}_d(0), \xi^g(\gamma_{ac}));$   
4:  $\gamma_{ac}^* = \text{bisectionSearch}(T_{lead}(\gamma_{ac}) - \Delta T_d^g, \rho_{pa}, \Gamma_{ac});$   
5:  $\mathbf{r}_{df} = \mathcal{P}_{ac}(\gamma_{ac}^*); \quad \phi = \vartheta_{ac}(\gamma_{ac}^*) - \pi;$   
6: **return**  $\xi^g(\gamma_{ac}^*), \mathbf{r}_{df}, \phi$

---

of the tangent to the path at the location reached after traveling  $\Gamma_{ac}$  distance along the path from the initial position. On this path, we use bisection method to find the point  $\mathbf{r}_{df}$  around which  $\mathcal{F}_d^g$  should be designed such that the minimum time to gather, i.e., the minimum time for the defenders to reach their desired positions on  $\mathcal{F}_d^g$  from  $\check{\mathbf{r}}_d(0) = \{\check{\mathbf{r}}_{dj}(0) \mid \forall j \in I_d\}$ , is smaller by at least  $\Delta T_d^g (> 0)$  than the time for the ACoM to reach this point. Here,  $\Delta T_d^g$  is a user-defined time to account for the time required by the defenders to travel from the actual initial position  $\mathbf{r}_{dj}(0)$  to the closest grid point  $\check{\mathbf{r}}_{dj}(0)$  and the time to get connected by strings once arrived at the desired formation. Algorithm 1 gives a detailed iterative scheme to find the desired positions on the open formation  $\mathcal{F}_d^g$ .

In Algorithm 1,  $\mathcal{T}(\check{\mathbf{r}}_d(0), \xi^g(\gamma_{ac}))$  is the minimum time, obtained by solving the MIQP (9) and the MILP [28], for the defenders to reach the formation  $\mathcal{F}_d^g$  defined by  $\xi^g(\gamma_{ac}) = \{\xi_l^g(\mathcal{P}_{ac}(\gamma_{ac}), \vartheta_{ac}(\gamma_{ac}) - \pi) | l \in I_d\}$ ; The function  $\text{shortestPath}(\bar{\mathcal{O}}, \mathcal{G}_{ct}, \mathbf{r}_{ac}(0), \mathbf{r}_{pa})$  finds the shortest path between  $\mathbf{r}_{ac}(0)$  and  $\mathbf{r}_{pa}$  on the  $C^1$ -tangent graph  $\mathcal{G}_{ct}$  defined for the environment with obstacles  $\bar{\mathcal{O}}$ . The term  $\frac{\gamma_{ac}}{\bar{v}_{ac}}$  gives the minimum time the attackers require to travel the distance  $\gamma_{ac}$  when they travel at the maximum possible speed  $\bar{v}_a$ .  $T_{lead}(\gamma_{ac})$  is the time lead or advantage the defenders have over the attackers during the gathering phase. The function  $\text{bisectionSearch}(T_{lead}(\gamma_{ac}) - \Delta T_d^g, \rho_{pa}, \Gamma_{ac})$  returns  $\gamma_{ac}^* \in [0, \Gamma_{ac} - \rho_{pa}]$  such that  $T_{lead}(\gamma_{ac}^*) = \Delta T_d^g$ .

The initial position  $\mathbf{r}_{dj}(0)$  is connected to the closest grid point  $\check{\mathbf{r}}_{dj}(0)$  by the largest circular arc that is tangential to the shortest path obtained between  $\check{\mathbf{r}}_{dj}(0)$  and its goal position  $\xi_{a(j)}^g$  (see Fig. 4). This gives us an approximate shortest path between  $\mathbf{r}_{dj}(0)$  and  $\xi_{a(j)}^g$ . The near time-optimal velocity profile for the circular arc is added to that obtained for the grid points  $\check{\mathbf{r}}_{dj}(0)$  and  $\xi_{a(j)}^g$ .

If there exists more than one shortest paths for the attackers to reach the protected area, then Algorithm 1 is run for all such paths, and the gathering formation on these paths for which the defenders require the shortest time to gather is chosen as the final gathering formation.

Using the near time-optimal velocity profiles on the chosen paths, all the defenders reach their desired positions on  $\mathcal{F}_d^g$  within finite time  $\mathcal{T}$ . Upon accomplishment of the gathering phase, the defenders execute the seeking phase. Next, we describe the seeking phase.

### B. Seeking

In practice, the attackers may deviate from their optimal trajectories computed during the gathering phase, i.e., be outside the circle  $\|\mathbf{r} - \mathbf{r}_{pa}\| = \|\mathbf{r}_{df}(\mathcal{T}) - \mathbf{r}_{pa}\|$  for  $\mathbf{r} \in \mathbb{R}^2$ . This requires defenders to move close to the attackers in order to enclose them. Recall that the defenders have already converged to the open formation  $\mathcal{F}_d^g$ . Then, during the seeking phase their goal is to move as a desired open, rigid formation  $\mathcal{F}_d^s$  centered at  $\mathbf{r}_{df}$ . The formation of the defenders seeks to: (i) get closer to the attacking swarm (i.e.,  $\|\mathbf{r}_{df}(t) - \mathbf{r}_{ac}\| < E_{\text{trans}}^s$ , for all  $t > T_{df}^s$ , where  $E_{\text{trans}}^s$  is a user-defined parameter,  $T_{df}^s > \mathcal{T}$  is some finite time at which the seeking phase would be completed), and (ii) maintain the orientation  $\phi$  of the formation  $\mathcal{F}_d^s$  toward the attacking swarm (see Fig. 3). We first generate desired trajectories for each defender assuming rigid-body motion of the desired formation, and then design finite-time convergent controllers to track the desired trajectories.

To generate desired trajectories that can be tracked by the defenders, we first consider that the center  $\mathbf{r}_{df}$  is governed by the same damped double integrator dynamics by which the defenders' motion is governed

$$\dot{\mathbf{r}}_{df} = \mathbf{v}_{df}, \quad \dot{\mathbf{v}}_{df} = \mathbf{u}_{df} - C_D \|\mathbf{v}_{df}\| \mathbf{v}_{df} \quad (13)$$

where  $\mathbf{v}_{df}$  is the velocity of the formation  $\mathcal{F}_d^s$ . To achieve the aforementioned objective (i), we design the control input

$$\mathbf{u}_{df} = \Omega_{\bar{u}_{df}^{s1}} \left( \sum_{k \in I_o} \sigma_{df}^{\delta_{f,k}} \mathbf{u}_p(\mathbf{x}_{df}^{\delta_{f,k}}) - k_1(\mathbf{r}_{df} - \mathbf{r}_{ac}) \right) + \Omega_{\bar{u}_{df}^{s2}} (C_D \|\mathbf{v}_{df}\| \mathbf{v}_{df} - k_2(\mathbf{v}_{df} - \mathbf{v}_{ac})) \quad (14)$$

where  $k_1, k_2 > 0$  are control gains,  $\mathbf{x}_{df}^{\delta_{f,k}} = [\mathbf{r}_{df}^T, \mathbf{v}_{df}^T, \mathbf{r}_{\delta_{f,k}}^T, \mathbf{v}_{\delta_{f,k}}^T]^T$ , where  $\mathbf{r}_{\delta_{f,k}}$  and  $\mathbf{v}_{\delta_{f,k}}$  are the position and velocity of the virtual  $\delta$ -agent on the obstacle  $\mathcal{O}_k$  corresponding to the defenders' formation  $\mathcal{F}_d^s$ , respectively. The summation term in (14) is for avoiding collision with obstacles. The avoidance control for obstacle  $\mathcal{O}_k$  is activated only when the  $\delta$ -agent is within a distance of  $\bar{R}_{df}^{\delta_{f,k}}$  from the formation center using the blending function  $\sigma_{df}^{\delta_{f,k}}$  characterized by  $(\bar{R}_{df}^{\delta_{f,k}}, \bar{R}_{df}^{\delta_{f,k}})$  where  $\bar{R}_{df}^{\delta_{f,k}} < \bar{R}_{df}^{\delta_{f,k}}$ . Two separate saturation functions  $\Omega_{\bar{u}_{df}^{s1}}$  and  $\Omega_{\bar{u}_{df}^{s2}}$ , with saturation limits  $\bar{u}_{df}^{s1}$  and  $\bar{u}_{df}^{s2}$ , respectively, are used for the terms that correspond to the potentials and velocities, respectively, to ensure that the desired formation moves with a bounded velocity, i.e.,  $\|\mathbf{v}_{df}\| < \bar{v}_{df}^s \triangleq \sqrt{\frac{\bar{u}_{df}^{s1} + \bar{u}_{df}^{s2}}{C_D}}$  since each defender's velocity is also bounded. We add the quadratic term  $C_D \|\mathbf{v}_{df}\| \mathbf{v}_{df}$  to the controller (14) to compensate for the drag term in the dynamics (13) whenever the controller is unsaturated to facilitate the convergence analysis of the resulting trajectories. In order for the desired formation to track the attackers, it has to move at least as fast as the attackers. To ensure this, we consider the following.

**Assumption 2:** The maximum acceleration of the desired formation is larger than the maximum acceleration of the attackers, i.e.,  $\bar{u}_a < \bar{u}_{df}^{s1} + \bar{u}_{df}^{s2}$ .

The desired positions  $\xi_l^s$  on  $\mathcal{F}_d^s$  are chosen to be regularly spaced on a semicircle of radius  $\rho_{df}^s$  centered at  $\mathbf{r}_{df}$

$$\xi_l^s = \mathbf{f}_l^s(\mathbf{r}_{df}, \phi_l^s) = \mathbf{r}_{df} + \rho_{df}^s \hat{\mathbf{o}}(\phi_l^s) \quad (15)$$

where  $\phi_l^s = \phi + \frac{\pi}{2} + \frac{\pi(l-1)}{N_d-1}$ ;  $\phi \in [0, 2\pi]$  is the orientation of  $\mathcal{F}_d^s$ , as shown in Fig. 3. The semicircular shape is chosen because it ensures largest area in the interior and the largest distance between the end positions  $\xi_1^s$  and  $\xi_{N_d}^s$  for the formation  $\mathcal{F}_d^s$  and requires the defenders to travel minimal distance to complete the closed regular polygonal formation around the attackers (discussed later in Section III-C). To achieve the objective (ii) of the seeking phase, the desired positions  $\xi_l^s$ ,  $l \in I_d$ , on the formation  $\mathcal{F}_d^s$ , in addition to moving with the velocity of the center of  $\mathcal{F}_d^s$ , have a rotational motion around the center  $\mathbf{r}_{df}$ . This motion is governed by

$$\begin{aligned} \dot{\xi}_l^s &= \eta_l^s = \mathbf{v}_{df} + \rho_{df}^s \dot{\phi} \hat{\mathbf{o}}(\phi_l^s + \frac{\pi}{2}) \\ \dot{\eta}_l^s &= \dot{\mathbf{v}}_{df} + \rho_{df}^s \left( \ddot{\phi} \hat{\mathbf{o}}(\phi_l^s + \frac{\pi}{2}) - \dot{\phi}^2 \hat{\mathbf{o}}(\phi_l^s) \right) \\ \ddot{\phi} &= -k_d^\phi (\phi - \theta) - k_d^\phi (\dot{\phi} - \dot{\theta}) \end{aligned} \quad (16)$$

where  $\theta = \tan^{-1}(\frac{y_{ac} - y_{df}}{x_{ac} - x_{df}})$  is the angle between the line joining the center  $\mathbf{r}_{ac}$ , and  $\mathbf{r}_{df}$ . The third equation in (16) aims to align the orientation of the formation  $\mathcal{F}_d^s$  toward the attackers by considering proportional-derivative-type feedback controller in which  $k_d^\phi, k_d^\phi > 0$  are control gains.

To maintain the desired formation, the defenders need to track their desired trajectories. To ensure timely convergence, inspired by Bhat and Bernstein [35], a finite-time convergent controller for  $\mathcal{D}_j \quad \forall j \in I_d$ , to track  $(\xi_{a(j)}^s, \eta_{a(j)}^s)$  is designed as

$$\mathbf{u}_{dj} = \dot{\eta}_{a(j)}^s + \Omega_{\bar{u}_d^{s1}} \left( \mathbf{u}_{dj}^{s1} \right) + \Omega_{\bar{u}_d^{s2}} \left( \mathbf{u}_{dj}^{s2} \right) \quad (17)$$

where  $\bar{u}_d^{s1}, \bar{u}_d^{s2} > 0$  are saturation limits and

$$\begin{aligned} \mathbf{u}_{dj}^{s1} &= -k_0 \text{sig}^{\alpha_1}(\mathbf{p}_{dj}) + \mathbf{u}_{dj}^{\text{col}} \\ \mathbf{u}_{dj}^{s2} &= -k_0 \text{sig}^{\alpha_2}(\mathbf{v}_{dj} - \eta_{a(j)}^s) + C_D \|\mathbf{v}_{dj}\| \mathbf{v}_{dj} \\ \mathbf{p}_{dj} &= \mathbf{r}_{dj} - \xi_{a(j)}^s + \frac{1}{k_0(2-\alpha_2)} \text{sig}^{2-\alpha_2}(\mathbf{v}_{dj} - \eta_{a(j)}^s) \\ \mathbf{u}_{dj}^{\text{col}} &= \sum_{j' \in I_d^j} \sigma_{dj}^{\delta_{j'}} \mathbf{u}_p(\mathbf{x}_{dj}^{\delta_{j'}}) + \sum_{k \in I_o} \sigma_{dj}^{\delta_{j,k}} \mathbf{u}_p(\mathbf{x}_{dj}^{\delta_{j,k}}) \\ &\quad + \sigma_{dj}^{\delta_{j,c}} \mathbf{u}_p(\mathbf{x}_{dj}^{\delta_{j,c}}) + \sigma_{dj}^{\delta_{j,p}} \mathbf{u}_p(\mathbf{x}_{dj}^{\delta_{j,p}}) \end{aligned} \quad (18)$$

where  $\alpha_1, \alpha_2 \in (0, 1)$ ;  $k_0 > 0$ ;  $\bar{u}_d^{s1}$ , and  $I_d^j = I_d \setminus \{j\}$ . In (18),  $\mathbf{u}_{dj}^{\text{col}}$  is the control term for collision avoidance and the terms in the expression of  $\mathbf{u}_{dj}^{\text{col}}$  correspond to collision avoidance from the other defenders, the obstacles, the connectivity region around the attackers, and the protected area, respectively. The superscripts  $\delta_{j,k}$ ,  $\delta_{j,c}$ , and  $\delta_{j,p}$  refer to the virtual  $\delta$ -agents corresponding to  $\mathcal{D}_j$  on the obstacle  $\mathcal{O}_k$ , attackers' connectivity region, and the protected area, respectively (see Figs. 2 and 3).  $\sigma_{dj}^{\delta_{j,k}}$ ,  $\sigma_{dj}^{\delta_{j,c}}$ , and  $\sigma_{dj}^{\delta_{j,p}}$  are blending functions for  $\mathcal{D}_j$  corresponding to  $\mathcal{D}_{j'}$ ,  $\mathcal{O}_k$ , attackers' connectivity region, and the protected area, respectively, characterized by  $(\bar{R}_{dj}^{\delta_{j,k}}, \bar{R}_{dj}^{\delta_{j,k}})$ ,  $(\bar{R}_{dj}^{\delta_{j,c}}, \bar{R}_{dj}^{\delta_{j,c}})$ , and  $(\bar{R}_{dj}^{\delta_{j,p}}, \bar{R}_{dj}^{\delta_{j,p}})$ , respectively, for all  $j, j' \in I_d$ ;  $k \in I_o$ . Again, the quadratic term  $C_D \|\mathbf{v}_{dj}\| \mathbf{v}_{dj}$  is added to compensate the drag in dynamics (2) when the controller is unsaturated. The radius  $\rho_{df}^s$  of the formation  $\mathcal{F}_d^s$  satisfies the following assumption.

**Assumption 3:** The radius  $\rho_{df}^s$  is such that the control terms corresponding to the collision avoidance from other defenders and the connectivity region are not active for any defender ( $\sigma_{dj}^{\delta_{j'}} = 0$  for all  $j, j' \in I_d$ ), i.e.,  $\rho_{df}^s \sqrt{2 - 2 \cos(\frac{\pi}{N_d-1})} > \bar{R}_{dj}^{\delta_{j,k}}$



and  $\rho_{df}^s > \rho_{ac} + \bar{R}_d^{\delta_c} + \bar{S}$ . Here,  $\bar{S} = \frac{\log(2)}{2C_D}$  is the distance an attacker, moving with maximum speed under (1), would travel before it comes to rest despite having applied maximum acceleration in opposite direction to its velocity. The safety parameter for the formation is then chosen as  $\bar{R}_{df}^{\delta_{f,k},\infty} > \rho_{df}^s + \rho_d + \bar{R}_d^{\delta_o} + \bar{S}_d$ , where  $\bar{S}_d$  is an additional safety distance that needs to be taken into account due to the damped double integrator dynamics and is defined formally later in Theorem 2.

Assumption 3 ensures that once the trajectory of  $\mathcal{D}_j \forall j \in I_d$ , converges to  $(\xi_{a(j)}^s, \eta_{a(j)}^s)$ , the input becomes  $\mathbf{u}_{dj} = \dot{\eta}_{a(j)}^s + C_D \|\eta_{a(j)}^s\| \eta_{a(j)}^s$  and the tracking errors  $\mathbf{r}_{dj} - \xi_{a(j)}^s$  and  $\mathbf{v}_{dj} - \eta_{a(j)}^s$  stay zero thereafter.

Once the defenders get close to the attackers with their orientation toward the attackers, i.e.,  $\|\mathbf{r}_{df} - \mathbf{r}_{ac}\| < E_{trans}^s$  and  $|\phi - \theta| < E_{rot}^s$ , the seeking phase is accomplished and the enclosing phase is initiated. Here,  $E_{trans}^s > \rho_{ac}$  and  $E_{rot}^s > 0$  are user-defined parameters. Next, we discuss the enclosing phase.

### C. Enclosing: StringNet Formation

In this phase, we want to ensure that the defenders converge to a StringNet formation (i.e., a closed formation of strings) around the attackers. The defenders should also ensure that they do not enter the connectivity region around the ACoM to avoid collisions with the attackers, and that they do not collide with each other.

To trap the attackers inside StringNet, a desired regular polygon formation  $\mathcal{F}_d^e$  is designed around the connectivity region of the attackers centered at ACoM, as shown in Fig. 3. Regular polygonal formation is chosen for the defenders as it provides the largest area in the interior of the StringNet formation formed by a given number of defenders. The desired positions  $\xi_l^e$  on the formation  $\mathcal{F}_d^e$  (green circles in Fig. 3) are chosen on the circle with radius  $\rho_{sn}$  centered at  $\mathbf{r}_{ac}$  and move with the velocity of ACoM

$$\dot{\xi}_l^e = \eta_l^e = \mathbf{v}_{ac}, \quad \text{where } \xi_l^e = \mathbf{r}_{ac} + \rho_{sn} \hat{\mathbf{o}}(\phi_l^e) \quad (19)$$

where  $\phi_l^e = \phi + \frac{\pi(2l-1)}{N_d}$  and  $\mathbf{v}_{ac} = \dot{\mathbf{r}}_{ac}$ , i.e., the formation  $\mathcal{F}_d^e$  moves with the speed of ACoM, so that the defenders are reactive to the motion of the attackers. Let us denote the StringNet associated with the formation  $\mathcal{F}_d^e$  by  $\mathcal{G}_{sn}$ . The defenders start tracking their desired positions around the attackers using the controller

$$\mathbf{u}_{dj} = \Omega_{\bar{u}_d^{e1}}(\mathbf{u}_{dj}^{e1}) + \Omega_{\bar{u}_d^{e2}}(\mathbf{u}_{dj}^{e2}) \quad (20)$$

where  $\bar{u}_d^{e1}$  and  $\bar{u}_d^{e2}$  are the control saturation limits for the enclosing phase and

$$\begin{aligned} \mathbf{u}_{dj}^{e1} &= -k_1(\mathbf{r}_{dj} - \xi_{a(j)}^e) + \mathbf{u}_{dj}^{\text{col}} \\ \mathbf{u}_{dj}^{e2} &= -k_2(\mathbf{v}_{dj} - \eta_{a(j)}^e) + C_D \|\mathbf{v}_{dj}\| \mathbf{v}_{dj}. \end{aligned} \quad (21)$$

The StringNet is considered to have been formed when the defenders tracking the position  $\xi_1^e$  and  $\xi_{N_d}^e$  satisfy  $\|\mathbf{r}_{d_{j_1}} - \xi_1^e\| < b_d$  and  $\|\mathbf{r}_{d_{j_{N_d}}} - \xi_{N_d}^e\| < b_d$ , where  $j_1 = a^{-1}(1)$  and  $j_{N_d} = a^{-1}(N_d)$  where  $a$  is given in (10). Here,  $b_d > 0$  is the minimum radius of the ball around the desired trajectory within which the

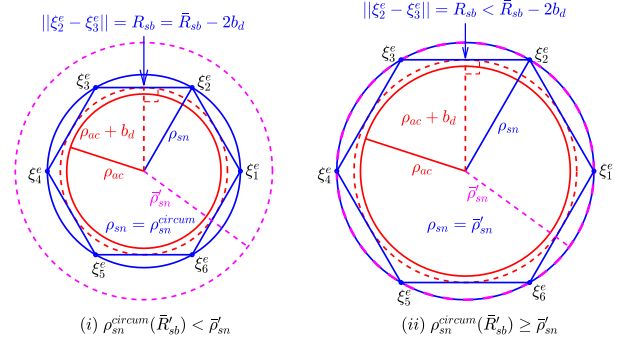


Fig. 5. Geometry for minimum number of defenders. (a)  $\rho_{sn}^{\text{circum}}(R'_{sb}) < \rho'_{sn}$ . (b)  $\rho_{sn}^{\text{circum}}(R'_{sb}) \geq \rho'_{sn}$ .

defender's trajectory will converge in the absence of the knowledge about the attackers' control acceleration. The parameter  $b_d$  is formally defined in Section IV in Theorem 5. The radius  $\rho_{sn}$  should satisfy  $\rho_{ac} + b_d + \bar{R}_d^d + \bar{S} < \rho_{sn} \leq \bar{\rho}_{sn} - b_d$  to ensure that the StringNet has at least a circular area of radius  $\rho_{ac}$  free inside it.

During the StringNet formation phase and moving it to the safe area, we want to ensure that: the defenders do not collide with the attackers, and minimum number of defenders are used such that the connectivity region of the attackers is completely contained inside the StringNet. Let  $R_{sb}$  be the distance between the defenders on the StringNet. Let  $\rho_{sn}^{\text{circum}}(R_{sb}) = \sqrt{(\rho_{ac} + b_d)^2 + (0.5R_{sb})^2}$  be the radius of the circumscribed circle of the regular polygon formed by the defenders with side length equal to  $R_{sb}$ . Note that the inscribed circle of this regular polygon has radius  $\rho_{ac} + b_d$  (see Fig. 5), i.e., the attackers can be completely contained inside this polygon. However, the value of  $\rho_{sn}^{\text{circum}}(R_{sb})$  cannot be larger than  $\rho'_{sn} = \bar{\rho}_{sn} - b_d$  due to limited free space between the obstacles. To ensure minimum number of defenders to achieve the StringNet, we need to place the defenders as far as possible. We choose  $R_{sb}$  while satisfying  $\rho_{sn}^{\text{circum}}(R_{sb}) < \rho'_{sn}$  as follows:

$$R_{sb} = \begin{cases} \bar{R}'_{sb}, & \text{if } \rho_{sn}^{\text{circum}}(\bar{R}'_{sb}) < \rho'_{sn} \\ 2\sqrt{(\rho'_{sn})^2 - (\rho_{ac} + b_d)^2}, & \text{otherwise} \end{cases} \quad (22)$$

where  $\bar{R}'_{sb} = \bar{R}_{sb} - 2b_d$  is the maximum possible safe distance at which the constraint on the length of the string barrier is not violated. With this choice of  $R_{sb}$ , we choose  $\rho_{sn} = \rho_{sn}^{\text{circum}}(R_{sb})$ . The two cases in (22) are visualized in Fig. 5. Since the defenders are placed as far as possible, this results into minimum number of defenders required to herd the given number of attackers with connectivity region of radius  $\rho_{ac}$  and this number is:  $N_d^m = \lceil \frac{\pi}{\cos^{-1}(\frac{\rho_{ac} + b_d}{\rho_{sn}^{\text{circum}}(R_{sb})})} \rceil$ , where  $\lceil \cdot \rceil$  gives the smallest integer greater than  $\cdot$ .

1) *Discussion on the Number of Defenders Needed for Attackers' Encirclement:* Since the attackers want to avoid defenders, they need to start moving away from the defenders well before they are at  $\bar{R}_a^{d,m} = \bar{S} + \rho_a + \rho_d < \bar{R}_{sb}$  distance away from the defenders for their safety under the given damped

**Algorithm 2:** Overall Approach.

- 
- 1 Find the desired open formation  $\mathcal{F}_d^g$  defined by  $\xi^g$ ;
  - 2 Gather at  $\xi^g$  under the near time-optimal, collision-free motion plan
  - 3 **while**  $\|\mathbf{r}_{df} - \mathbf{r}_{ac}\| > E_{trans}^s \& |\phi - \theta| > E_{rot}^s$  **do**
  - 4   Seek the attacking swarm;
  - 5 **while**  $\left(\|\mathbf{r}_{dj} - \xi_{a(j)}^e\| > b_d \forall j \in I_d\right) \& R_{d1}^{dN_d} > \bar{R}_{sb}$  **do**
  - 6   Enclose the attacking swarm;
  - 7 Herd the enclosed attackers to  $\mathcal{S}$ .
- 

double integrator dynamics. Consider the worst case, when the attackers only intend to start moving away from the defenders when they are  $\underline{R}_a^{d,m}$  distance away and hope to escape and hit the protected area otherwise. In the absence of actual barrier between two defenders, such as in [16], the defenders will have to be at most  $2\underline{R}_a^{d,m}$  away from each other in order to create a closed virtual barrier around the attackers, assuming the worst case. In this case, the number of the defenders required to enclose the attackers with overall circular footprint of  $\rho_{ac}$  and herd them to the safe area would be  $N_d^{\text{no,bar}} = \pi(\tan^{-1}(\frac{\underline{R}_a^{d,m}}{\rho_{ac} + b_d}))^{-1}$ , which is much larger than  $N_d^m$ , which is needed in our approach, for large values of  $\rho_{ac}$ .

**D. Herding: Moving the StringNet to Safe Area**

Once the defenders form the StringNet  $\mathcal{G}_{sn}$  around the attackers, they track a desired, rigid, closed formation  $\mathcal{F}_d^h$  that herds the enclosed attackers to the safe area while avoiding the static obstacles. Similar to the seeking phase, the dynamics of the center  $\mathbf{r}_{df}$  of the desired formation  $\mathcal{F}_d^h$  are governed by (13), where the control action is

$$\mathbf{u}_{df} = \Omega_{\bar{u}_{df}^{h1}} \left( -k_1(\mathbf{r}_{df} - \mathbf{r}_{sa}) + \sum_{k \in I_o} \sigma_{df}^{\delta f, k} \mathbf{u}_p(\mathbf{x}_{df}^{\delta f, k}) \right) \quad (23)$$

where  $\bar{u}_{df}^{h1}$  is a control design parameter such that  $\bar{u}_{df}^{h1} < \bar{u}_a$ , which ensures that during the herding phase, the defenders' desired formation does not move faster than the attackers can move. The desired positions  $\xi_l^h$  of the defenders on  $\mathcal{F}_d^h$  are chosen to be on a circle of radius  $\rho_{sn}$  centered at  $\mathbf{r}_{df}$  and are governed by

$$\begin{aligned} \dot{\xi}_l^h &= \dot{\eta}_l^h = \dot{\mathbf{r}}_{df} = \mathbf{v}_{df} \\ \dot{\eta}_l^h &= \dot{\mathbf{v}}_{df} \\ \xi_l^h &= \mathbf{r}_{df} + \rho_{sn} \hat{\mathbf{o}}(\phi_l^h), \text{ where } \phi_l^h = \phi + \frac{\pi(2l-1)}{N_d}. \end{aligned} \quad (24)$$

The control action for  $\mathcal{D}_j \ \forall j \in I_d$ , to track  $(\xi_{a(j)}^h, \eta_{a(j)}^h)$  during the herding phase is designed as

$$\mathbf{u}_{dj} = \dot{\eta}_{a(j)}^h + \Omega_{\bar{u}_d^{h1}}(\mathbf{u}_{dj}^{h1}) + \Omega_{\bar{u}_d^{h2}}(\mathbf{u}_{dj}^{h2}) \quad (25)$$

where  $\mathbf{u}_{dj}^{h1}$  and  $\mathbf{u}_{dj}^{h2}$  are same as  $\mathbf{u}_{dj}^{s1}$  and  $\mathbf{u}_{dj}^{s2}$  obtained after replacing the superscript  $s$  by  $h$  in (18), respectively.

The overall approach is summarized in Algorithm 2.

**IV. SAFETY AND CONVERGENCE ANALYSIS**

In this section, we formally prove the safety of the defenders and their convergence to their respective desired trajectories so that they achieve their goals in each phase.

**A. Gathering Phase**

By construction, the trajectories obtained by solving the MIQP (9) and the MILP in [28] are safe and near time-optimal, i.e., defenders reach their respective goal locations within a finite time  $\mathcal{T}$  on these trajectories without colliding with each other.

**B. Seeking Phase**

In this section, we prove the safety and the convergence of the defenders during the seeking phase. The analysis for the seeking phase is divided into three theorems; Theorem 2 proves the safety of the defenders moving under the tracking controller (17). Theorem 3 proves the finite-time convergence of the tracking controller to the desired trajectories, and Theorem 4 proves that the desired trajectories of the defenders converge to the desired neighborhood of the attackers' swarm within a finite time.

The defenders should be able to move faster than their desired positions in order to track these desired positions. For this, we make the following assumption.

*Assumption 4:* First, the defenders can move faster than their desired positions change in time during the seeking phase, i.e.,  $\bar{u}_d > \bar{\eta}^s + \bar{u}_d^{s1} + \bar{u}_d^{s2}$ , where  $\bar{\eta}^s = \bar{u}_{df}^{s1} + \bar{u}_{df}^{s2} + \rho_{df}^s(\sqrt{(\bar{\phi})^2 + (\bar{\phi})^4})$  is the maximum value of  $\|\dot{\eta}_l^s\|$ , for all  $l \in I_d$ , where  $\bar{\phi}$  and  $\bar{\phi}$  are the maximum values of  $|\dot{\phi}|$  and  $|\phi|$  during the seeking phase, and second, saturation limit on the terms corresponding to position and collision avoidance feedback is larger than that for the velocity feedback terms, i.e.,  $\bar{u}_d^{s1} > \bar{u}_d^{s2}$ .

Next, prove the safety of the defenders.

*Theorem 2:* Let Assumption 4 hold. Then, under the control law (17), each defender  $\mathcal{D}_j$ ,  $j \in I_d$ , remains always at least a safety distance  $R_d^{d,m}$  apart from every other defender  $\mathcal{D}_{j'}$ ,  $j' \in I_d \setminus \{j\}$ .

*Proof:* We want to show that at any time, the distance between two defenders remains greater than a specified safety distance  $R_d^{d,m}$ . Consider the relative dynamics between two defenders  $\mathcal{D}_j$  and  $\mathcal{D}_{j'}$

$$\begin{aligned} \dot{\mathbf{e}}_r &= \dot{\mathbf{r}}_{dj} - \dot{\mathbf{r}}_{dj'} = \mathbf{e}_v \\ \dot{\mathbf{e}}_v &= \mathbf{u}_{dj} - \mathbf{u}_{dj'} - C_D \|\mathbf{v}_{dj}\| \mathbf{v}_{dj} + C_D \|\mathbf{v}_{dj'}\| \mathbf{v}_{dj'} \\ &= \Delta \dot{\eta}^s + \Omega_{\bar{u}_d^{s1}}(\mathbf{u}_{dj}^{s1}) + \Omega_{\bar{u}_d^{s2}}(\mathbf{u}_{dj}^{s2}) - \Omega_{\bar{u}_d^{s1}}(\mathbf{u}_{dj'}^{s1}) \\ &\quad - \Omega_{\bar{u}_d^{s2}}(\mathbf{u}_{dj'}^{s2}) - C_D(\|\mathbf{v}_{dj}\| \mathbf{v}_{dj} - \|\mathbf{v}_{dj'}\| \mathbf{v}_{dj'}) \end{aligned} \quad (26)$$

where  $\Delta \dot{\eta}^s = \dot{\eta}_{a(j)}^s - \dot{\eta}_{a(j')}^s = 2\rho_{df}^s \sin(\frac{\pi(a(j') - a(j))}{2(N_d - 1)}) (\dot{\phi} \hat{\mathbf{o}}(\phi_{j,j'}^s) + \dot{\phi}^2 \hat{\mathbf{o}}(\phi_{j,j'}^s + \frac{\pi}{2})),$  where  $\phi_{j,j'}^s = \frac{\phi_{a(j)}^s + \phi_{a(j')}^s}{2}$ . Only the component of the relative dynamics along the line joining  $\mathcal{D}_j$  and  $\mathcal{D}_{j'}$  affects the distance between them. Define  $\mathbf{e}_v^\parallel = (\mathbf{v}_{dj} - \mathbf{v}_{dj'}) \cdot \hat{\mathbf{e}}_r = v_{dj}^\parallel - v_{dj'}^\parallel$ , where  $\hat{\mathbf{e}}_r \triangleq \frac{\mathbf{e}_r}{\|\mathbf{e}_r\|}$  is the unit vector parallel to  $\mathbf{e}_r$ ,  $v_{dj}^\parallel$  and  $v_{dj'}^\parallel$  are the components of

the velocities  $\mathbf{v}_{dj}$  and  $\mathbf{v}_{dj'}$ , respectively. For  $R_{dj}^{dj'} \leq \underline{R}_{dj}^{dj',\nu}$ , the vector  $\frac{\partial V_{dj}^{dj'}}{\partial R_{dj}^{dj'}}$  has magnitude  $\nu$  and is directed away from  $\mathcal{D}_j$  and toward  $\mathcal{D}_{j'}$ .  $\nu$  is chosen to be a very large number such that it is at least  $(\frac{1}{\epsilon})$  times greater than the rest of the terms in  $\mathbf{u}_{dj}^{s1}$ , where  $0 < \epsilon \ll 1$ . This gives us  $(\Omega_{\bar{u}_d^{s1}}(\mathbf{u}_{dj}^{s1})) \cdot \hat{\mathbf{e}}_r = (1 - \epsilon)\bar{u}_d^{s1}$  and  $(\Omega_{\bar{u}_d^{s1}}(\mathbf{u}_{dj'}^{s1})) \cdot \hat{\mathbf{e}}_r = -(1 - \epsilon)\bar{u}_d^{s1}$  for all  $R_{dj}^{dj'} \leq \underline{R}_{dj}^{dj',\nu}$ . This yields

$$\begin{aligned} \dot{e}_v &= 2\dot{u}_d^{s1} + \Delta \dot{\eta}^s \cdot \hat{\mathbf{e}}_r - k_0(\text{sig}^{\alpha_2}(\mathbf{v}_{dj}) - \text{sig}^{\alpha_2}(\mathbf{v}_{dj'})) \cdot \hat{\mathbf{e}}_r \\ &\geq 2(\dot{u}_d^{s1}) - 2\rho_{df}^s \sqrt{(\ddot{\phi})^2 + (\dot{\phi})^4} + K \end{aligned} \quad (27)$$

where  $\dot{u}_d^{s1} = (1 - \epsilon)\bar{u}_d^{s1}$  and  $K$  is the minimum value of the last term in the expression of  $\dot{e}_v$ . The gains  $k_d^\phi$  and  $k_d^{\dot{\phi}}$  are chosen such that  $\Upsilon = \frac{\rho_{df}^s \sqrt{(\ddot{\phi})^2 + (\dot{\phi})^4}}{\dot{u}_d^{s1}} < 1$ .

If  $e_v \geq 0$  at  $R_{dj}^{dj'} = \underline{R}_{dj}^{dj',\nu}$ , then  $R_{dj}^{dj'}$  would stay equal or larger than  $\underline{R}_{dj}^{dj',\nu}$ . If  $e_v < 0$ , we have  $v_{dj} - v_{dj'} < 0$  implying  $K > 0$  resulting into

$$\dot{e}_v \geq 2(\dot{u}_d^{s1}) - 2\rho_{df}^s \sqrt{(\ddot{\phi})^2 + (\dot{\phi})^4} = 2(1 - \Upsilon)(\dot{u}_d^{s1}). \quad (28)$$

Using comparison lemma [36] and performing integration on (28), one can establish that  $R_{dj}^{dj'} \geq \underline{R}_{dj}^{dj',\nu} - S_d$  when  $e_v$  again becomes larger than 0 within the time  $\frac{e_v}{2(1-\Upsilon)(\dot{u}_d^{s1})}$ , where  $S_d(e_v) = \frac{(e_v)^2}{4(1-\Upsilon)(\dot{u}_d^{s1})}$ . In the worst-case head-to-head motion between the defenders,  $e_v = -2\bar{v}_d$ . In this case, we have  $\bar{S}_d = S_d(-2\bar{v}_d) = \frac{\bar{u}_d}{(1-\Upsilon)\bar{C}_D(\bar{u}_d^{s1})}$ . For the safety distance  $R_d^{d,m} (> 2\rho_d)$  between the defenders, we choose  $\underline{R}_{dj}^{dj',\nu} > \bar{S}_d + R_d^{d,m}$  such that  $R_{dj}^{dj'} > R_d^{d,m}$  for all times, ensuring safety of the defenders. ■

Analysis similar to that in Theorem 2 can be performed for the collision avoidance from the  $\delta$ -agents on the static obstacles and on the connectivity region of the attackers; and parameters of the corresponding blending functions are chosen to include the additional safety distance of  $\bar{S}_d$ . Next, we define feasible initial conditions.

**Definition 3 (Feasible initial conditions):** Any initial condition that does not belong to the set  $\mathcal{M}_0 = \{\mathbf{r}_{dj}, \mathbf{v}_{dj} \in \mathbb{R}^2 \mid \forall j \in I_d, \mathbf{v}_{dj} = \mathbf{0}, \mathbf{u}_{dj} = \mathbf{0} \text{ as per (17)}\}$ , or any initial condition from which the defenders' trajectories approach  $\mathcal{M}_0$ ; the latter depends on the desired states of the defenders. A formal construction of this set is left open for future research.

For these feasible initial conditions, the finite-time trajectory tracking by the defenders is proven as follows.

**Theorem 3:** Under Assumption 4 and feasible initial conditions (see Definition 3), all defenders  $\mathcal{D}_j$ ,  $j \in I_d$ , track their desired trajectories  $(\xi_{a(j)}^s, \eta_{a(j)}^s)$  in finite time under the bounded control action given in (17) while avoiding collisions, if  $\alpha_1 = \frac{\alpha_2}{2-\alpha_2}$ .

*Proof:* Safety for the defenders (in terms of no collisions between the defenders with each other, the static obstacles, and the connectivity region of the attackers) under the control

action (17) has been proved in Theorem 2. For proving convergence and tracking of the desired trajectories, define the errors  $\tilde{\mathbf{r}}_{dj} = \mathbf{r}_{dj} - \xi_{a(j)}^s$  and  $\tilde{\mathbf{v}}_{dj} = \mathbf{v}_{dj} - \eta_{a(j)}^s$ . When the defenders are no longer in conflict with other defenders or obstacles (i.e.,  $\sigma_{dj}^{dj'} = \sigma_{dj}^{\delta_{j,k}} = \sigma_{dj}^{\delta_{j,c}} = \sigma_{dj}^{\delta_{j,p}} = 0 \quad \forall j, j' \in I_d; k \in I_o$ ), the errors satisfy

$$\begin{aligned} \dot{\tilde{\mathbf{r}}}_{dj} &= \tilde{\mathbf{v}}_{dj} \\ \dot{\tilde{\mathbf{v}}}_{dj} &= \Omega_{\bar{u}_d^{s1}}(\tilde{\mathbf{u}}_{dj}^{s1}) + \Omega_{\bar{u}_d^{s2}}(\tilde{\mathbf{u}}_{dj}^{s2}) - C_D \|\mathbf{v}_{dj}\| \mathbf{v}_{dj} \end{aligned} \quad (29)$$

where  $\tilde{\mathbf{u}}_{dj}^{s1} = -k_0 \text{sig}^{\alpha_1}(\tilde{\mathbf{r}}_{dj} + \frac{1}{k_0(2-\alpha_2)} \text{sig}^{2-\alpha_2}(\tilde{\mathbf{v}}_{dj}))$  and  $\tilde{\mathbf{u}}_{dj}^{s2} = -k_0 \text{sig}^{\alpha_2}(\tilde{\mathbf{v}}_{dj}) + C_D \|\mathbf{v}_{dj}\| \mathbf{v}_{dj}$ .

Since  $\|\mathbf{v}_{dj}\|$  and  $\|\eta_{a(j)}^s\|$  are bounded, we can choose  $k_0 = \frac{C_D \bar{v}_d^2}{(\bar{v}_d - \bar{\eta}^s)^{\alpha_2}}$  such that  $\|\mathbf{u}_{dj}^{s2}\| \leq \bar{u}_d^{s2}$ , i.e., the control term corresponding to the velocity feedback is unsaturated for all times. Let  $\tilde{\mathbf{x}}_{dj} = [\tilde{\mathbf{r}}_{dj}^T, \tilde{\mathbf{v}}_{dj}^T]^T$  and define  $\mathcal{H}_{dj}^s = \{\tilde{\mathbf{x}}_{dj} \in \mathbb{R}^4 \mid \|\mathbf{p}_{dj}\| \leq (\frac{\bar{u}_d^{s1}}{k_0})^{\frac{1}{\alpha_1}}\}$ . Since  $\|\mathbf{v}_{dj}\|$  is bounded, the set  $\mathcal{H}_{dj}^s$  is compact and nonempty. Let  $\tilde{\mathbf{x}}_{dj0} = [\tilde{\mathbf{r}}_{dj0}^T, \tilde{\mathbf{v}}_{dj0}^T]^T$  be the initial state. Consider the following two cases.

*Case (i),  $\tilde{\mathbf{x}}_{dj0} \in \mathcal{H}_{dj}^s$ :* Inside the set  $\mathcal{H}_{dj}^s$ , we have  $\|\tilde{\mathbf{u}}_{dj}^{s1}\| \leq \bar{u}_d^{s1}$ . The error dynamics inside  $\mathcal{H}_{dj}^s$  is

$$\dot{\tilde{\mathbf{r}}}_{dj} = \tilde{\mathbf{v}}_{dj}, \quad \dot{\tilde{\mathbf{v}}}_{dj} = -k_0 \text{sig}^{\alpha_1}(\mathbf{p}_{dj}) - k_0 \text{sig}^{\alpha_2}(\tilde{\mathbf{v}}_{dj}). \quad (30)$$

From the dynamics in (30), we have  $\dot{\mathbf{p}}_{dj} = -\mathbf{p}_{dj} \|\mathbf{p}_{dj}\|^{\alpha_1-1} \|\tilde{\mathbf{v}}_{dj}\|^{1-\alpha_2}$ , which implies that the set  $\mathcal{H}_{dj}^s$  is forward invariant. Define a candidate Lyapunov function  $V_d^s$  as

$$V_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj}) = \frac{1}{\alpha_2'} \|\mathbf{p}_{dj}\|^{\alpha_2'} + q_1 \tilde{\mathbf{v}}_{dj}^T \mathbf{p}_{dj} + \frac{q_2}{\alpha_2^{3-}} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2^{3-}} \quad (31)$$

where  $\alpha_2' = \frac{3-\alpha_2}{2-\alpha_2}$  and  $\alpha_2^{3-} = 3 - \alpha_2$ . The time derivative of  $V_d^s$  along the trajectories of (29) is

$$\begin{aligned} \dot{V}_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj}) &= -\|\mathbf{p}_{dj}\|^{\frac{1+\alpha_2}{2-\alpha_2}} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2^-} - q_1(k_0 \tilde{\mathbf{v}}_{dj}^T \mathbf{p}_{dj} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2^-} \\ &\quad + k_0 \|\mathbf{p}_{dj}\|^{\alpha_1'} + \tilde{\mathbf{v}}_{dj}^T \mathbf{p}_{dj} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2^-} \|\mathbf{p}_{dj}\|^{-\alpha_1'}) \\ &\quad - q_2 k_0 (\|\tilde{\mathbf{v}}_{dj}\|^2 + \tilde{\mathbf{v}}_{dj}^T \mathbf{p}_{dj} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2^-} \|\mathbf{p}_{dj}\|^{-\alpha_1'}) \end{aligned} \quad (32)$$

where  $\alpha_1^- = 1 - \alpha_1$ ,  $\alpha_2^- = 1 - \alpha_2$ , and  $\alpha_2^+ = 1 + \alpha_2$ . We can show that for any  $\lambda > 0$ , we have

$$\begin{aligned} V_d^s(\lambda^{2-\alpha_2} \tilde{\mathbf{r}}_{dj}, \lambda \tilde{\mathbf{v}}_{dj}) &= \lambda^{3-\alpha_2} V_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj}) \\ \dot{V}_d^s(\lambda^{2-\alpha_2} \tilde{\mathbf{r}}_{dj}, \lambda \tilde{\mathbf{v}}_{dj}) &= \lambda^2 \dot{V}_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj}). \end{aligned} \quad (33)$$

Similar to the analysis in [35], for  $0 < q_1 < 1$ ,  $q_2 > 1$ , and using the homogeneity property in (33), we can show that  $\dot{V}_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj}) \leq -c(V_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj}))^{\frac{2}{3-\alpha_2}}$  where  $c = -\max_{\tilde{\mathbf{x}}_{dj} \in \mathcal{H}_V} \dot{V}_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj})$  where  $\mathcal{H}_V = \{\tilde{\mathbf{x}}_{dj} \in \mathbb{R}^4 \mid V_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj}) = 1\}$ . From [37, Th. 4.2], we can conclude that the equilibrium  $\tilde{\mathbf{r}}_{dj} = \tilde{\mathbf{v}}_{dj} = \mathbf{0}$  is finite-time stable. The time of convergence  $T_{dj}^{s,tr}(\tilde{\mathbf{x}}_{dj0})$  satisfies  $T_{dj}^{s,tr}(\tilde{\mathbf{x}}_{dj0}) \leq \frac{1}{c(1-\frac{2}{3-\alpha_2})} (V_d^s(\tilde{\mathbf{r}}_{dj0}, \tilde{\mathbf{v}}_{dj0}))^{1-\frac{2}{3-\alpha_2}}$ .

*Case (ii),  $\tilde{\mathbf{x}}_{dj0} \notin \mathcal{H}_{dj}^s$ :* Outside the set  $\mathcal{H}_{dj}^s$ , we have

$$\dot{\tilde{\mathbf{r}}}_{dj} = \tilde{\mathbf{v}}_{dj}, \quad \dot{\tilde{\mathbf{v}}}_{dj} = \underbrace{-\bar{u}_d^{s1} \frac{\mathbf{p}_{dj}}{\|\mathbf{p}_{dj}\|}}_{\text{term 1}} \underbrace{-k_0 \text{sig}^{\alpha_2}(\tilde{\mathbf{v}}_{dj})}_{\text{term 2}}. \quad (34)$$



The time derivative  $\dot{V}_d^s$  along the trajectories of (34) is

$$\begin{aligned} \dot{V}_d^s &= -\frac{\bar{u}_d^{s1}}{k_0} \|\mathbf{p}_{dj}\|^{\alpha'_2} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2} - q_1 \bar{u}_d^{s1} \|\mathbf{p}_{dj}\| \\ &\quad - q_2 k_0 \|\tilde{\mathbf{v}}_{dj}\|^2 - q_2 \bar{u}_d^{s1} \frac{\tilde{\mathbf{v}}_{dj}^T \mathbf{p}_{dj}}{\|\mathbf{p}_{dj}\|} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2} \\ &\quad - q_1 \tilde{\mathbf{v}}_{dj}^T \mathbf{p}_{dj} \left( \frac{\bar{u}_d^{s1}}{k_0} \frac{\|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2}}{\|\mathbf{p}_{dj}\|} + k_0 \|\tilde{\mathbf{v}}_{dj}\|^{-\alpha_2} \right) \\ &\leq -\frac{\bar{u}_d^{s1}}{k_0} \|\mathbf{p}_{dj}\|^{\alpha'_2} \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2} - q_1 \bar{u}_d^{s1} \|\mathbf{p}_{dj}\| \\ &\quad - q_2 k_0 \|\tilde{\mathbf{v}}_{dj}\|^2 + q_2 \bar{u}_d^{s1} \|\tilde{\mathbf{v}}_{dj}\|^{2-\alpha_2} \\ &\quad + q_1 \left( \frac{\bar{u}_d^{s1}}{k_0} \|\tilde{\mathbf{v}}_{dj}\|^{2-\alpha_2} + k_0 \|\mathbf{p}_{dj}\| \|\tilde{\mathbf{v}}_{dj}\|^{\alpha_2} \right) \\ &< 0 \quad (\because \|\tilde{\mathbf{v}}_{dj}\| \leq \left( \frac{\bar{u}_d^{s1}}{k_0} \right)^{\frac{1}{\alpha_2}}). \end{aligned} \quad (35)$$

This implies that the origin  $\mathbf{r}_{dj} = \tilde{\mathbf{v}}_{dj} = \mathbf{0}$  is an asymptotically stable equilibrium. Asymptotic convergence of the trajectories of (34) implies that the error states  $\mathbf{p}_{dj}$  and  $\tilde{\mathbf{v}}_{dj}$  will reach inside the set  $\mathcal{H}_{dj}^s$  in finite time. Since the time derivative  $\dot{V}_d^s$  is only negative and no convergence rate is established, Lyapunov function based argument cannot be used to find upper bound on the time of convergence of the trajectories to the set  $\mathcal{H}_{dj}^s$ . We consider a switched system  $S_w$  to provide an upper bound on the time of convergence of the system trajectories to the set  $\mathcal{H}_{dj}^s$ . The switched system  $S_w$  has two vector fields:  $f_1$ : right-hand side (RHS) of (34) and  $f_2$ : RHS of (34) without term 1. Assume that the term 1 in (34) is inactive at  $t = 0$ , which yields

$$\dot{\tilde{\mathbf{r}}}_{dj} = \tilde{\mathbf{v}}_{dj}, \quad \dot{\tilde{\mathbf{v}}}_{dj} = -k_0 \text{sig}^{\alpha_2}(\tilde{\mathbf{v}}_{dj}). \quad (36)$$

The distance travelled  $\tilde{\gamma}_{dj}$  and the speed  $\tilde{v}_{dj} = \|\tilde{\mathbf{v}}_{dj}\|$  satisfy

$$\dot{\tilde{\gamma}}_{dj} = \tilde{v}_{dj}, \quad \dot{\tilde{v}}_{dj} = -k_0 (\tilde{v}_{dj})^{\alpha_2}. \quad (37)$$

We can integrate the system in (37) to find the time  $T_0$  and distance  $\tilde{\gamma}_0$  at which the speed  $\tilde{v}_{dj}$  becomes zero as

$$T_0(\tilde{v}_{dj0}) = \frac{(\tilde{v}_{dj0})^{1-\alpha_2}}{k_0(1-\alpha_2)}, \quad \tilde{\gamma}_0(\tilde{v}_{dj0}) = \frac{(\tilde{v}_{dj0})^{2-\alpha_2}}{k_0(2-\alpha_2)}. \quad (38)$$

At  $t = T_0(\tilde{v}_{dj0})$ , let the term 1 in (34) be turned ON, i.e., vector field  $f_1$  is active. Since  $\tilde{\mathbf{v}}_{dj} = \mathbf{0}$  at  $t = T_0(\tilde{v}_{dj0})$ , we observe that the position and velocity vectors are parallel and point along the radial direction  $\tilde{\mathbf{r}}_{dj}(T_0(\tilde{v}_{dj0})) = \tilde{\mathbf{r}}_{dj}(0) + \tilde{\gamma}_0(\tilde{v}_{dj0}) \frac{\tilde{\mathbf{v}}_{dj}(0)}{\|\tilde{\mathbf{v}}_{dj}(0)\|}$  for all  $t > T_0(\tilde{v}_{dj0})$ . The distance travelled  $\tilde{\gamma}_{dj}$  and the speed  $\tilde{v}_{dj}$  are then governed by

$$\dot{\tilde{\gamma}}_{dj} = \tilde{v}_{dj}, \quad \dot{\tilde{v}}_{dj} = \bar{u}_d^{s1} - k_0 (\tilde{v}_{dj})^{\alpha_2} \geq m_0 (\tilde{v}_{dj} - \bar{v}_{\alpha_2}) \quad (39)$$

where  $m_0 = -k_0 \alpha_2 \left( \frac{\bar{u}_d^{s1}}{k_0} \right)^{\frac{\alpha_2-1}{\alpha_2}}$  and  $\bar{v}_{\alpha_2} = \left( \frac{\bar{u}_d^{s1}}{k_0} \right)^{\frac{1}{\alpha_2}}$ . Using comparison lemma [36] on (39), we can show that the

$$\begin{aligned} \tilde{\gamma}_{dj}(t) &= \frac{\bar{v}_{\alpha_2}}{m_0} + \bar{v}_{\alpha_2} (\Delta t - \frac{e^{m_0 \Delta t}}{m_0}) \leq \tilde{\gamma}_{dj}(t) \\ \tilde{v}_{dj}(t) &= \bar{v}_{\alpha_2} (1 - e^{m_0 \Delta t}) \leq \tilde{v}_{dj}(t) \end{aligned} \quad (40)$$

where  $\Delta t = t - T_0(\tilde{v}_{dj0})$ . We know that after travelling distance  $\tilde{\gamma}_{dj}(t)$  along  $\tilde{\mathbf{r}}_{dj}(T_0(\tilde{v}_{dj0}))$  toward the origin, the position vector  $\tilde{\mathbf{r}}_{dj}(t)$  becomes  $\tilde{\mathbf{r}}_{dj}(t) = (\|\tilde{\mathbf{r}}_{dj}(T_0(\tilde{v}_{dj0}))\| - \tilde{\gamma}_{dj}(t)) \tilde{\mathbf{r}}_{dj}(T_0(\tilde{v}_{dj0}))$ . Since  $\tilde{\mathbf{v}}_{dj}(t)$  points in the direction of

$\tilde{\mathbf{r}}_{dj}(T_0(\tilde{v}_{dj0}))$ , we have

$$\begin{aligned} \|\mathbf{p}_{dj}(t)\| &= \|\tilde{\mathbf{r}}_{dj}(T_0(\tilde{v}_{dj0}))\| - \tilde{\gamma}_{dj}(t) - \frac{1}{k_0(2-\alpha_2)} (\tilde{v}_{dj}(t))^{1-\alpha_2} \\ &\leq \|\tilde{\mathbf{r}}_{dj}(T_0(\tilde{v}_{dj0}))\| - \tilde{\gamma}_{dj}(t) - \frac{1}{k_0(2-\alpha_2)} (\tilde{v}_{dj}(t))^{1-\alpha_2} \\ &= \bar{p}_{dj}(t). \end{aligned} \quad (41)$$

Let  $\bar{T}_{dj}^{\mathcal{H}}$  be the time such that:  $\left( \frac{\bar{u}_d^{s1}}{k_0} \right)^{\frac{1}{\alpha_1}} = \bar{p}_{dj}(\bar{T}_{dj}^{\mathcal{H}})$ , i.e., the time at which the trajectories of the switched system  $S_w$  would have already entered the set  $\mathcal{H}_{dj}^s$ . We claim that this time  $\bar{T}_{dj}^{\mathcal{H}}$  is an upper bound to the actual time of convergence,  $T_{dj}^{\mathcal{H}}$ , of the trajectories of (34) to the set  $\mathcal{H}_{dj}^s$ . This is true because turning off term 1 in (34) actually slows down the convergence of the trajectories toward the origin as term 1 always points toward the origin under Assumption 4, i.e., more acceleration toward the origin making convergence of the trajectories faster. Once inside the set  $\mathcal{H}_{dj}^s$  the trajectories converge to the origin in finite time, as described in the analysis of *Case (i)*. In conclusion, starting at  $\tilde{\mathbf{r}}_{dj0}, \tilde{\mathbf{v}}_{dj0}$ , the trajectories of the error dynamics (29) converge to origin in finite time  $T_{dj}^{s,tr}(\tilde{\mathbf{x}}_{dj0}) \leq \bar{T}_{dj}^{s,tr}(\tilde{\mathbf{x}}_{dj0})$  where  $\bar{T}_{dj}^{s,tr}(\tilde{\mathbf{x}}_{dj0})$

$$\bar{T}_{dj}^{s,tr}(\tilde{\mathbf{x}}_{dj0}) = \begin{cases} \frac{(V_d^s(\tilde{\mathbf{r}}_{dj0}, \tilde{\mathbf{v}}_{dj0}))^{\alpha_2''}}{c(\alpha_2'')}, & \text{if } \tilde{\mathbf{x}}_{dj0} \in \mathcal{H}_{dj}^s \\ \bar{T}_{dj}^{\mathcal{H}}(\tilde{\mathbf{x}}_{dj0}) + \frac{(\bar{V}_d^s)^{\alpha_2''}}{c(\alpha_2'')}, & \text{if } \tilde{\mathbf{x}}_{dj0} \notin \mathcal{H}_{dj}^s \end{cases} \quad (42)$$

where  $\bar{V}_d^s = \max_{\tilde{\mathbf{x}}_{dj} \in \mathcal{H}_{dj}^s} V_d^s(\tilde{\mathbf{r}}_{dj}, \tilde{\mathbf{v}}_{dj})$  and  $\alpha_2'' = \frac{1-\alpha_2}{3-\alpha_2}$ .

We also note that a similar upper bound on the time of convergence can be provided by considering the initial velocity to be pointing away from the origin in the radially outward direction with the same speed as the speed at the initial time. Let this time be  $\bar{T}_{dj}^{s,tr,\parallel}$ . In the interest of space, we do not provide derivation of  $\bar{T}_{dj}^{s,tr,\parallel}$ , but the idea is to integrate the system (34)

to find this time. Then, the minimum of  $\bar{T}_{dj}^{s,tr}$  and  $\bar{T}_{dj}^{s,tr,\parallel}$  is the upper bound on the time of convergence of the error  $\tilde{\mathbf{x}}_{dj0}$  to the origin ■

Next, we prove the convergence of the desired formation during the seeking phase to the desired neighborhood of the attackers' formation.

**Theorem 4:** If Assumption 3 holds, then under the dynamics in (13), (16) and control action (14), the center  $\mathbf{r}_{df}$  of the formation  $\mathcal{F}_d^s$  reaches within the distance of  $E_{trans}^s$  from  $\mathbf{r}_{ac}$ , i.e.,  $\|\mathbf{r}_{df} - \mathbf{r}_{ac}\| \leq E_{trans}^s$ , and the orientation  $\phi$  reaches within the distance of  $E_{rot}^s$  from  $\theta$  in finite time, i.e.,  $|\phi - \theta| < E_{rot}^s$ , while not colliding with the obstacles.

**Proof:** Under Assumption 3, safety of the formation  $\mathcal{F}_d^s$  can be established by doing analysis similar to that in Theorem 2. Define the errors  $\tilde{\mathbf{r}}_{dfs} = \mathbf{r}_{df} - \mathbf{r}_{ac}$  and  $\tilde{\mathbf{v}}_{dfs} = \mathbf{v}_{df} - \mathbf{v}_{ac}$ . When the formation  $\mathcal{F}_d^s$  is no longer in conflict with the obstacles, (i.e.,  $\sigma_{df}^{\delta f, k} = 0$ ), the errors satisfy

$$\begin{aligned} \dot{\tilde{\mathbf{r}}}_{dfs} &= \tilde{\mathbf{v}}_{dfs} \\ \dot{\tilde{\mathbf{v}}}_{dfs} &= \Omega_{\bar{u}_{df}^{s2}}(\tilde{\mathbf{u}}_{df}^{s2}) - \Omega_{\bar{u}_{df}^{s1}}(k_1 \tilde{\mathbf{r}}_{dfs}) - C_D \|\mathbf{v}_{df}\| \mathbf{v}_{df} - \dot{\mathbf{v}}_{ac} \end{aligned} \quad (43)$$

where  $\tilde{\mathbf{u}}_{df}^{s2} = -k_2 \tilde{\mathbf{v}}_{dfs} + C_D \|\mathbf{v}_{df}\| \mathbf{v}_{df}$ ,  $\dot{\mathbf{v}}_{ac}$  is the acceleration of ACoM that is unknown to the defenders and acts as a perturbation in (43), and  $\|\dot{\mathbf{v}}_{ac}\| \leq \bar{u}_a$ . We will use perturbation analysis to prove the result. The nominal error dynamics without the

perturbation  $\dot{\mathbf{v}}_{ac}$  reads

$$\begin{aligned}\dot{\mathbf{r}}_{df^s} &= \tilde{\mathbf{v}}_{df^s} \\ \dot{\tilde{\mathbf{v}}}_{df^s} &= \Omega_{\tilde{u}_{df}^{s2}} \left( \tilde{\mathbf{u}}_{df}^{s2} \right) - \Omega_{\tilde{u}_{df}^{s1}} (k_1 \tilde{\mathbf{r}}_{df^s}) - C_D \|\mathbf{v}_{df}\| \mathbf{v}_{df}.\end{aligned}\quad (44)$$

Under Assumption 2, following Theorem 3 with  $\alpha_1 = \alpha_2 = 1$ , we can show that the origin  $\tilde{\mathbf{r}}_{df^s} = \tilde{\mathbf{v}}_{df^s} = \mathbf{0}$  is an asymptotically stable equilibrium of (44).

Since  $\|\mathbf{v}_{df}\|$  and  $\|\mathbf{v}_{ac}\|$  are bounded, we can choose  $k_2 = \frac{C_D(\tilde{v}_{df}^s)^2}{(\tilde{v}_{df}^s - \tilde{v}_a)}$  such that  $\|\tilde{\mathbf{u}}_{df}^{s2}\| \leq \tilde{u}_{df}^{s2}$ , i.e., the control term corresponding to the velocity feedback is unsaturated for all times. Let us define the set  $\mathcal{H}_{df}^s = \{\tilde{\mathbf{r}}_{df^s}, \tilde{\mathbf{v}}_{df^s} \in \mathbb{R}^2 \mid \|\tilde{\mathbf{v}}_{df^s}\| < \tilde{v}_{df}^s + \tilde{u}_a \text{ \& } \|\tilde{\mathbf{r}}_{df^s}\| < \frac{\tilde{u}_{df}^{s1}}{k_1}\}$ . Consider the largest ball of radius  $\tilde{r}^s$ ,  $\mathcal{B}_{df}^s$ , that lies completely inside  $\mathcal{H}_{df}^s$  and is defined as  $\mathcal{B}_{df}^s = \{\tilde{\mathbf{r}}_{df^s}, \tilde{\mathbf{v}}_{df^s} \in \mathbb{R}^2 \mid \|\tilde{\mathbf{x}}_{df^s}\| < \tilde{r}^s\}$ , where  $\tilde{\mathbf{x}}_{df^s} = [(\tilde{\mathbf{r}}_{df^s})^T, (\tilde{\mathbf{v}}_{df^s})^T]^T$ . Inside  $\mathcal{B}_{df}^s$ , the errors follow:

$$\dot{\tilde{\mathbf{x}}}_{df^s} = \begin{bmatrix} \mathbf{0}_2 & \mathbf{I}_2 \\ -k_1 \mathbf{I}_2 & -k_2 \mathbf{I}_2 \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{r}}_{df^s} \\ \tilde{\mathbf{v}}_{df^s} \end{bmatrix} - \begin{bmatrix} 0 \\ \dot{\mathbf{v}}_{ac} \end{bmatrix} = \tilde{\mathbf{A}} \tilde{\mathbf{x}}_{df^s} + \tilde{\mathbf{g}} \quad (45)$$

where  $\mathbf{I}_2$  and  $\mathbf{0}_2$  are the two-by-two identity and zero matrices, respectively, the disturbance term  $\tilde{\mathbf{g}}$  is bounded as  $\|\tilde{\mathbf{g}}\| \leq \tilde{u}_a$ . The nominal system  $\dot{\tilde{\mathbf{x}}}_{df^s} = \tilde{\mathbf{A}} \tilde{\mathbf{x}}_{df^s}$  is exponentially stable for  $k_2, k_1 > 0$ . As per [36, Th. 4.6], exponential stability of the nominal linear time-invariant system guarantees existence of a positive definite matrix  $\mathbf{P}_d$  that satisfies the Lyapunov equation,  $\tilde{\mathbf{A}}^T \mathbf{P}_d + \mathbf{P}_d \tilde{\mathbf{A}} = -\mathbf{Q}_d$ , for any given positive definite matrix  $\mathbf{Q}_d$ . Then, the Lyapunov function defined as:  $V_{df} = (\tilde{\mathbf{x}}_{df^s})^T \mathbf{P}_d \tilde{\mathbf{x}}_{df^s}$  satisfies the conditions as required in [36, Lemma 9.2] with constants  $c_1, c_2, c_3, c_4$  given in terms of the eigenvalues of  $\mathbf{P}_d$  and  $\mathbf{Q}_d$  as:  $c_1 = \lambda_{\min}(\mathbf{P}_d)$ ,  $c_2 = \lambda_{\max}(\mathbf{P}_d)$ ,  $c_3 = \lambda_{\min}(\mathbf{Q}_d)$ , and  $c_4 = 2\lambda_{\max}(\mathbf{P}_d)$ . As per [36, Lemma 9.2], if  $\|\tilde{\mathbf{g}}\| \leq \tilde{u}_a < \frac{c_3}{c_4} \sqrt{\frac{c_1}{c_2}} c_0 \tilde{r}^s$  for all  $t > 0$ , all  $\tilde{\mathbf{x}}_{df^s} \in \mathcal{B}_{df}^s$  for some positive constant  $c_0 < 1$ , then for all  $\|\tilde{\mathbf{x}}_{df^s}(t_0)\| < \sqrt{\frac{c_1}{c_2}} \tilde{r}^s$ , the solution  $\tilde{\mathbf{x}}_{df^s}(t)$  of the perturbed system in (45) satisfies the following:

- 1)  $\|\tilde{\mathbf{x}}_{df^s}(t)\| \leq \sqrt{\frac{c_2}{c_1}} \exp(-\frac{(1-c_0)c_3}{2c_2}(t - t_0)) \|\tilde{\mathbf{x}}_{df^s}(t_0)\| \quad \forall t_0 \leq t < t_0 + T_{df}^s$ ;
- 2)  $\|\tilde{\mathbf{x}}_{df^s}(t)\| \leq b_{df}^s = \frac{c_4}{c_3} \sqrt{\frac{c_2}{c_1}} \frac{\tilde{u}_a}{c_0} \quad \forall t \geq t_0 + T_{df}^s$  for some finite time  $T_{df}^s \leq \bar{T}_{df}^s(\tilde{\mathbf{x}}_{df^s}(t_0))$ , where

$$\bar{T}_{df}^s(\tilde{\mathbf{x}}_{df^s}(t_0)) = t_0 - \frac{2c_2}{(1-c_0)c_3} \log\left(\frac{b_{df}^s}{\|\tilde{\mathbf{x}}_{df^s}(t_0)\|} \sqrt{\frac{c_1}{c_2}}\right). \quad (46)$$

We choose  $E_{\text{trans}}^s > b_{df}^s + \rho_{ac} + \rho_{sn}$ , which guarantees that the center  $\mathbf{r}_{df}$  reaches within a distance of  $E_{\text{trans}}^s$  from  $\mathbf{r}_{ac}$ .

Similarly, the orientation  $\phi$  satisfies  $\dot{\phi} = -k_d^\phi(\phi - \theta) - k_d^\phi(\dot{\phi} - \dot{\theta})$ . We have that  $\phi = \theta$  and  $\dot{\phi} = \dot{\theta}$  are exponentially stable equilibrium of (IV-B). Following the similar analysis as for  $\mathbf{r}_{df}$ , we can show that  $|\phi - \theta|$  becomes and stays smaller than  $E_{\text{rot}}^s$  in finite time. ■

### C. Enclosing Phase: StringNet Formation

Next, we prove the safe accomplishment of the StringNet formation during the enclosing phase in Theorem 5.

*Theorem 5:* The StringNet  $\mathcal{G}_{sn}$  centered at  $\mathbf{r}_{ac}$  is formed around the attackers in finite time under the state-feedback, bounded control actions given in (17) (seeking phase) and (20) (enclosing phase) while avoiding collisions, for all feasible initial conditions.<sup>4</sup>

*Proof:* 1) Seeking Phase: Assumption 3 ensures that the desired positions  $\xi_{a(j)}^s$  are such that when the defender  $\mathcal{D}_j$  is at  $\xi_{a(j)}^s$ , for all  $j \in I_d$ , the blending functions  $\sigma_{dj}^{\delta_{j,k}}, \sigma_{dj}^{\delta_{j,l}}$ , and  $\sigma_{dj}^{\delta_{j,k}}$  are zero. As shown in Theorem 3, the defenders track their desired trajectories forming the desired rigid formation  $\mathcal{F}_d^s$  in finite time  $\bar{T}_d^{s,tr} = \max_{j \in I_d} \bar{T}_{dj}^{s,tr}(\tilde{\mathbf{x}}_{dj0})$ . The dynamics (13) and (16), as shown in Theorem 4, ensure  $\|\mathbf{r}_{df} - \mathbf{r}_{ac}\| \leq E_{\text{trans}}^s$  and  $|\phi - \theta| < E_{\text{rot}}^s$  in finite time  $\bar{T}_{df}^s$  triggering the enclosing phase in finite time.

2) Enclosing phase: The safety of the defenders during the enclosing phase can be established using the similar arguments as in Theorem 2. Let  $\tilde{\mathbf{r}}_{dj^e} = \mathbf{r}_{dj} - \xi_{a(j)}^e, \tilde{\mathbf{v}}_{dj^e} = \mathbf{v}_{dj} - \eta_{a(j)}^e$ , and  $\tilde{\mathbf{x}}_{dj^e} = [(\tilde{\mathbf{r}}_{dj^e})^T, (\tilde{\mathbf{v}}_{dj^e})^T]^T$ . The dynamics governing the errors  $\tilde{\mathbf{r}}_{dj^e}$  and  $\tilde{\mathbf{v}}_{dj^e}$  are same as in (43). Following similar arguments as in the proof of Theorem 4, we can show that  $\|\tilde{\mathbf{x}}_{dj^e}(t)\| \leq b_{dj} = \frac{c_4}{c_3} \sqrt{\frac{c_2}{c_1}} \frac{\tilde{u}_a}{c_0}$ , for  $t > T_{dj}$ . This implies that  $\mathcal{D}_j$  reaches  $b_{dj}$  close to its desired trajectory in finite time and stays bounded within  $b_{dj}$  thereafter. Denote  $b_d = \max_{j \in I_d} b_{dj}$ . All the defenders reach within  $b_d$  distance of their desired locations in the StringNet in finite time  $T \geq \mathcal{T} + \bar{T}_{df}^s(\tilde{\mathbf{x}}_{df^s}(\mathcal{T})) + \bar{T}_d^{s,tr} + \max_{j \in I_d} T_{dj}$  and the StringNet is achieved in finite time. ■

*Remark 1:* Theorems 2–5 show that the StringNet formation by the defenders will be achieved around the attackers in finite time. We also provide upper bounds on the time of convergence of the proposed control laws when no conflicts with physical obstacles or other agents occur. However, finding the time that the defenders would take to avoid collisions with obstacles and with other defenders during seeking and enclosing phase is not trivial. Finding the initial conditions, under which the defenders will succeed in herding the attackers under the proposed strategy in an arbitrary obstacle environment, is ongoing work.

### D. Herding Phase

In Theorem 6, we prove the safety and convergence of the StringNet formation to  $\mathcal{S}$  during the herding phase.

*Theorem 6:* After StringNet  $\mathcal{G}_{sn}$  is formed, the defenders on  $\mathcal{G}_{sn}$  herd all the attackers enclosed inside  $\mathcal{G}_{sn}$  to the safe area  $\mathcal{S}$  ( $\rho_{sa} > \bar{\rho}_{sn}$ ), i.e.,  $\|\mathbf{r}_{ai} - \mathbf{r}_{sa}\| < \rho_{sa}$  for all  $\mathcal{A}_i$  inside  $\mathcal{G}_{sn}$  while avoiding the obstacles under the control action (25) starting at feasible initial conditions (see Definition 3).

*Proof:* Since the desired formation  $\mathcal{F}_d^h$  moves as a rigid formation, we only consider the virtual agent at  $\mathbf{r}_{df}$  with size

<sup>4</sup>Same as in Definition 3 but with the set  $\mathcal{M}_0$  defined as per (20) during the enclosing phase.

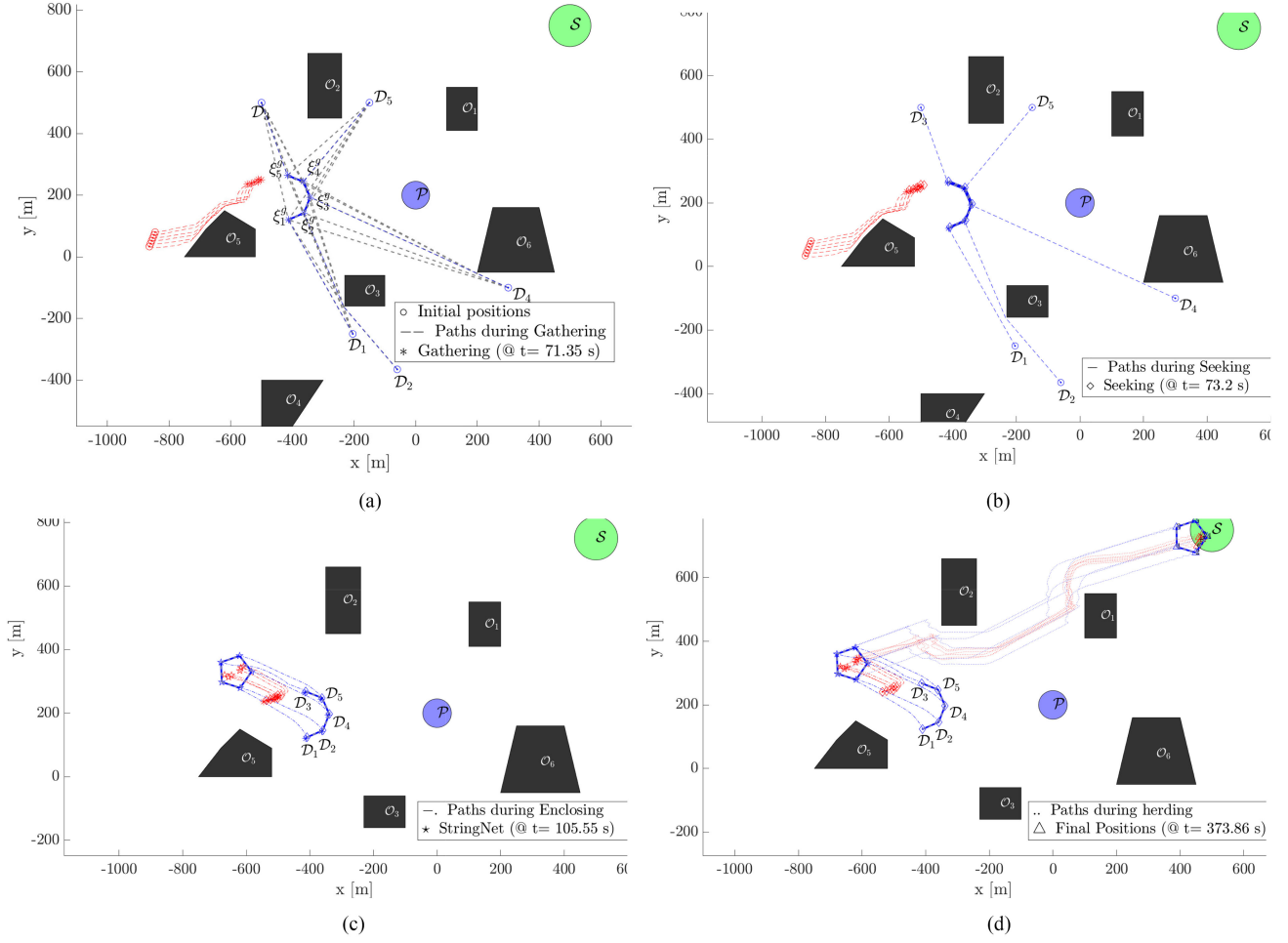


Fig. 6. Snapshots of the paths of the agents during StringNet Herding. (a) Gathering phase. (b) Seeking phase. (c) Enclosing phase. (d) Herding phase.

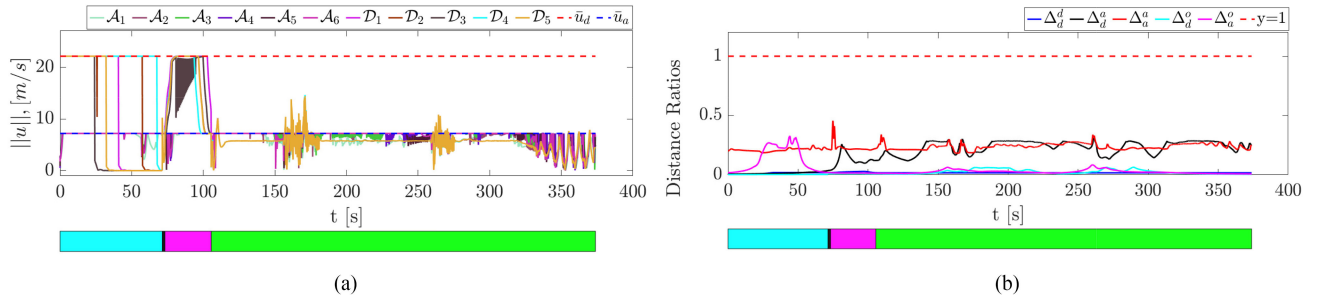


Fig. 7. Control inputs and critical distance ratios during StringNet Herding. (a) Inputs. (b) Critical distance ratios.

$\rho_{sn} + \rho_d$  whose dynamics are

$$\dot{\mathbf{r}}_{df^h} = \mathbf{v}_{df}, \quad \dot{\mathbf{v}}_{df} = \Omega_{\bar{u}_{df}^{h_1}}(\mathbf{u}_{df}^{h_1}) - C_D \|\mathbf{v}_{df}\| \mathbf{v}_{df} \quad (47)$$

where  $\tilde{\mathbf{r}}_{df^h} = \mathbf{r}_{df} - \mathbf{r}_{sa}$  and  $\mathbf{u}_{df}^{h_1} = -k_1(\mathbf{r}_{df} - \mathbf{r}_{sa}) + \sum_{k \in I_o} \sigma_{df}^{\delta_{f,k}} \mathbf{u}_p(\mathbf{x}_{df}^{\delta_{f,k}})$ . Using similar arguments as in Theorem 2, we can ensure the safety of  $\mathcal{F}_d^h$  if  $\bar{R}_{df}^{\delta_{f,k}, \infty} > \rho_{sn} + \rho_d + \bar{R}_d^{\delta_o} + \bar{S}$  during the herding phase. In the absence of any obstacle's local potential field ( $\sigma_{df}^{\delta_{f,k}} = 0$ ), we have

$\mathbf{u}_{df}^{h_1} = -k_1 \tilde{\mathbf{r}}_{df^h}$ . We define a candidate Lyapunov function

$$V_d^h = \begin{cases} \frac{k_1 \|\tilde{\mathbf{r}}_{df^h}\|^2}{2} + \frac{\|\mathbf{v}_{df}\|^2}{2}, & \text{if } \|\tilde{\mathbf{r}}_{df^h}\| < \frac{\bar{u}_{df}^{h_1}}{k_1} \\ \bar{u}_{df}^{h_1} \|\tilde{\mathbf{r}}_{df^h}\| + \frac{\|\mathbf{v}_{df}\|^2}{2} - \frac{(\bar{u}_{df}^{h_1})^2}{2k_1}, & \text{otherwise} \end{cases} \quad (48)$$

The function  $V_d^h$  is 0 at  $\tilde{\mathbf{r}}_{df^h} = \mathbf{v}_{df} = \mathbf{0}$ , is positive definite, continuous, and its time derivative along the trajectories of (47) is:  $\dot{V}_d^h = -C_D \|\mathbf{v}_{df}\|^3$ .  $\dot{V}_d^h$  is negative semidefinite and we have from the dynamics (47) that the largest invariant subset in



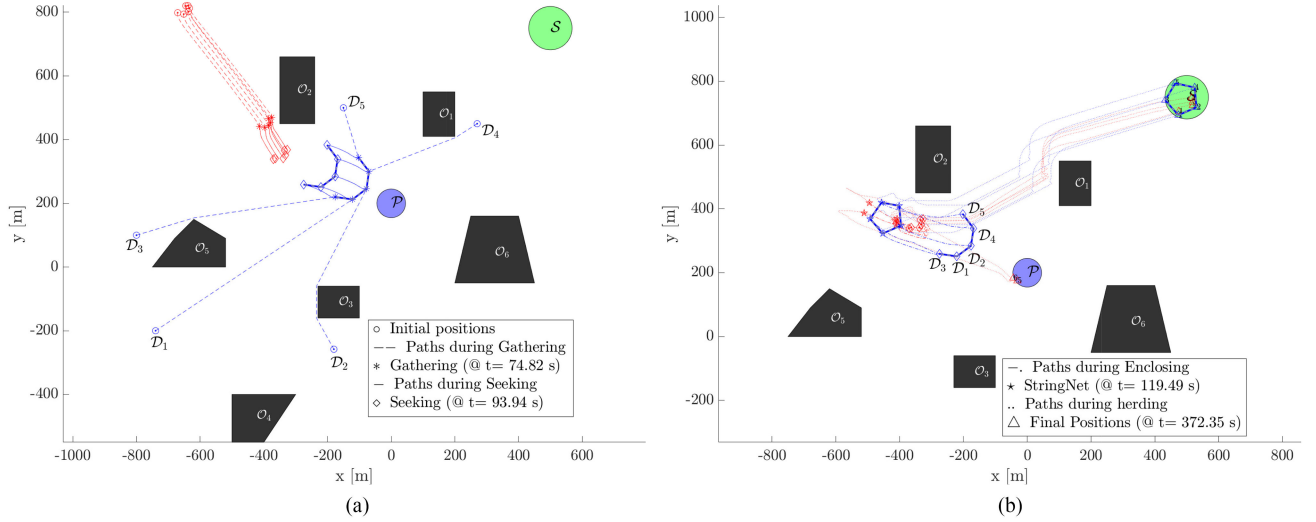


Fig. 8. Snapshots of the paths of the agents during StringNet Herding (attackers do not stay close). (a) Gathering and seeking phase. (b) Enclosing and herding phase.

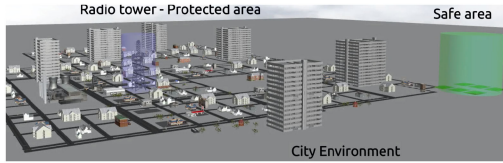


Fig. 9. City environment in Gazebo Simulator.

$\mathcal{Q} = \{\tilde{\mathbf{r}}_{df^h}, \mathbf{v}_{df} \in \mathbb{R}^2 | \dot{V}_h = 0\}$  is the origin  $\tilde{\mathbf{r}}_{df^h} = \mathbf{v}_{df} = \mathbf{0}$ . Using Lasalle's invariance principle ([36, Th. 4.4]), the trajectories of the system (47) converge to  $\tilde{\mathbf{r}}_{df^h} = \mathbf{v}_{df} = \mathbf{0}$ , i.e., the center  $\mathbf{r}_{df}$  converges to  $\mathbf{r}_{sa}$  and so does the desired formation  $\mathcal{F}_d^h$ . From Theorem 3, the defenders track these desired trajectories under (25) in finite time and, hence, herd the attackers to  $\mathcal{S}$ . ■

## V. SIMULATION RESULTS

In this section, we provide simulations of defenders herding an adversarial swarm of attackers to  $\mathcal{S}$ . The safety of the defenders is assessed by critical distance ratios:  $\Delta_d^o = \max_{j \in I_d, k \in I_o} \frac{R_{d,j,k}^{o,m}}{R_{d,j}^{a,m}}$ ,  $\Delta_d^a = \max_{i \in I_a, j \in I_d} \frac{R_{d,j}^{a,m}}{R_{d,i}^{o,m}}$ , and  $\Delta_d^d = \max_{j \neq j' \in I_d} \frac{R_{d,j,j'}^{d,m}}{R_{d,j}^{a,m}}$ , where the superscript  $m$  denotes the minimum safety distance between the corresponding agents. Critical distance ratios for the attackers  $\Delta_a^o$  and  $\Delta_a^a$  are defined similar to  $\Delta_d^o$  and  $\Delta_d^d$ , respectively. These critical ratios should be less than 1 for no collisions.

### A. MATLAB Simulations

1) *Herding*: We consider six attackers moving in a line-shaped formation, attacking the protected area. Based on the initial positions of the attackers, the defenders assume that the connectivity region of the attackers has radius  $\rho_{ac} = 36$  m.

We choose five defenders to herd the six attackers. Some key parameters used in the simulations are:  $\rho_a = \rho_d = 0.5$ ,  $C_D = 0.2$ ,  $\bar{v}_a = 6$  m/s ( $\bar{u}_a = 7.2$  m/s<sup>2</sup>),  $\bar{v}_d = 10.25$  m/s ( $\bar{u}_d = 21.02$  m/s<sup>2</sup>),  $\alpha_2 = 0.9$ , and  $\rho_{sn} = 48$  m.

For the given initial positions of the defenders and the attackers, we solve the MIQP and MILP iteratively, as described in Algorithm 1, to obtain the desired open formation  $\mathcal{F}_d^g$  and the corresponding defender-goal assignments. Fig. 6(a) shows paths for all possible defender-goal pairs (light gray). We compute the assignment as  $(\mathcal{D}_1, \xi_1^g), (\mathcal{D}_2, \xi_2^g), (\mathcal{D}_3, \xi_3^g), (\mathcal{D}_4, \xi_4^g), (\mathcal{D}_5, \xi_5^g)$  and the corresponding assignment cost to be 171.75 within the computation time of 2.6824 s using MATLAB on a computer with 16-GB RAM. The chosen paths of the defenders to gather at  $\mathcal{F}_d^g$  and the attackers' paths during this phase are shown in Fig. 6(a) (blue and red, respectively).

Fig. 6(b)–(d) shows the snapshots of the paths taken by the defenders and the attackers after the seeking, enclosing, and herding phases are completed. As we can observe from the plots in Fig. 6, the defenders are able to gather at the desired formation  $\mathcal{F}_d^g$  before the attackers. The defenders then are able to seek and enclose the attackers by forming the StringNet formation around them [see Fig. 6(c)] despite the attackers trying to move away. Fig. 6(d) shows that the defenders are able to herd all enclosed attackers to the safe area. The norms of the inputs of all the agents and critical distance ratios are shown in Fig. 7(a) and (b). The colored bars at the bottom in Fig. 7(a) and (b) show the time duration of each phase, where the cyan, black, magenta, and green color correspond to gathering, seeking, enclosing, and herding phase, respectively. As observed in Fig. 7(a), norms of all the inputs are bounded. From Fig. 7(b), we observe that all the distance ratios are smaller than 1 ensuring no collision happened during the entire duration.

2) *Herding When Some Attackers Leave the Connectivity Region*: In this section, we provide a simulation for the case when some of the attackers do not stay inside the connectivity region. The snapshots of paths traveled by the agent are shown in Fig. 8.

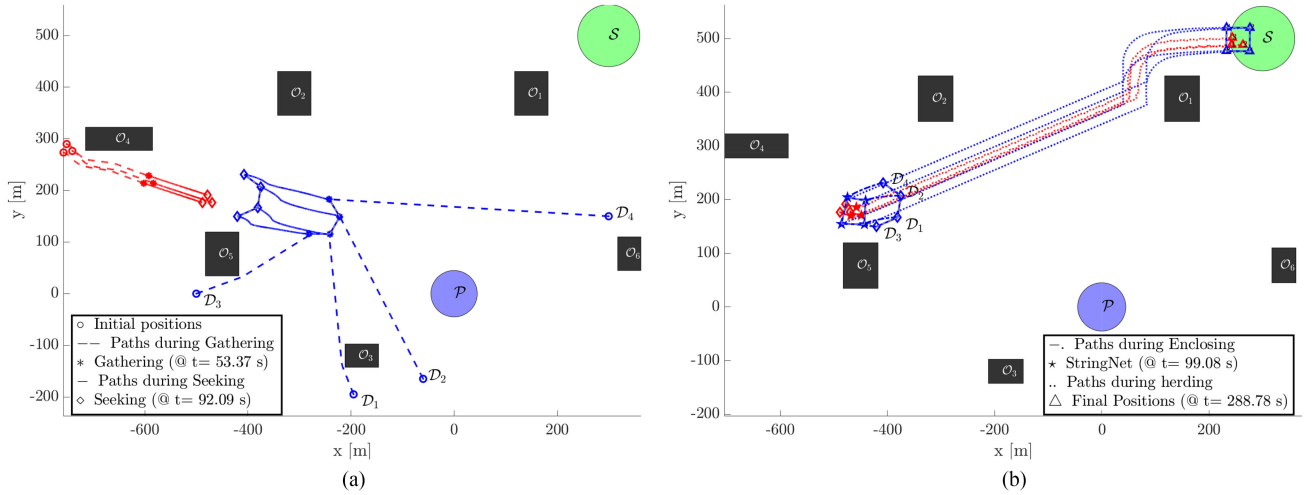


Fig. 10. Snapshots of the paths of the agents during StringNet Herding (Gazebo simulation). (a) Gathering and seeking phase. (b) Enclosing and herding phase.

As it is observed in Fig. 8(b), two of the attackers are outside the connectivity region and are not enclosed by the defenders inside the StringNet. The attackers outside the StringNet disconnect themselves from the rest of the attackers and are able to reach the protected area [see Fig. 8(b)]. Clearly, the current approach fails to herd all the attackers when the attackers do not stay together. To handle this kind of situations, the defenders can choose larger radius for the StringNet formation, but there is limit on how large this radius can be because the barrier can only be maintained when the defenders are within  $\bar{R}_{sb}$  from each other. Another solution is to split the defenders into smaller teams and task them to capture the smaller teams of the attackers whenever the attackers do not stay together and split into smaller teams. In our ongoing work, we are extending the proposed “StringNet Herding” approach to herd multiple teams of the attackers to safe area by dynamically assigning the defenders to the teams of attackers based on the diameter of the attackers’ teams. For visualization, the video of the simulations can be found at <https://tinyurl.com/y4xzumbj>.

### B. Gazebo Simulations

In this section, we provide simulation results for the proposed approach implemented on quadrotor vehicles simulated in the physics-based Gazebo simulator, RotorS [38]. We consider the city environment, as shown in Fig. 9, with several tall buildings as the primary obstacles. Since the quadrotors are flying at a certain altitude, the smaller houses are not considered as obstacles. Gazebo environment provides noisy measurements for each of the quadrotors. The quadrotors track the trajectories generated by our algorithm using the on-board controller [39]. Snapshots of the trajectories followed by the quadrotors in Gazebo simulation are shown in Fig. 10. As observed, the defender quadrotors are able to gather in the path of the attackers, enclose the attackers, and then herd them to the safe area located outside the city. The Gazebo simulation video can be found at <https://tinyurl.com/y4xzumbj>.

## VI. CONCLUSION

In this article, we proposed a herding method called “StringNet Herding” for defending a protected area from an adversarial swarm. A closed formation of strings (StringNet) was formed by the defenders around the attackers in 2-D space, restricting the attackers’ motion to the interior of the StringNet. The StringNet was then moved to a safe area while avoiding the convex polygonal obstacles in the space. Using a combination of near time-optimal, open-loop controllers for planning the formation of the defenders, along with state-feedback finite-time controllers for tracking the desired formation, the defenders were able to herd an attacking swarm that starts sufficiently far from the protected area.

Future work includes investigation of the proposed approach on hardware systems and its extension to the 3-D case. The current work would also be extended to include more intelligent behaviors by the attackers, for example, attacking as multiple smaller flocks or splitting in smaller teams instead of maintaining cohesiveness as a flock.

## ACKNOWLEDGMENT

The authors would like to thank K. Garg for several helpful discussions and A. Unnikrishnan for providing the basic-city block model in Gazebo Simulator.

## REFERENCES

- [1] R. D. Arnold, H. Yamaguchi, and T. Tanaka, “Search and rescue with autonomous flying robots through behavior-based cooperative intelligence,” *J. Int. Humanitarian Action*, vol. 3, no. 1, 2018, Art. no. 18.
- [2] Y. Liu and G. Nejat, “Robotic urban search and rescue: A survey from the control perspective,” *J. Intell. Robot. Syst.*, vol. 72, no. 2, pp. 147–165, 2013.
- [3] D. Albani, J. IJsselmuiden, R. Haken, and V. Trianni, “Monitoring and mapping with robot swarms for agricultural applications,” in *Proc. 14th IEEE Int. Conf. Adv. Video Signal Based Surveillance*, 2017, pp. 1–6.
- [4] M. Duarte *et al.*, “Application of swarm robotics systems to marine environmental monitoring,” in *Proc. OCEANS-Shanghai*, 2016, pp. 1–8.
- [5] S. Wilson, T. P. Pavlic, G. P. Kumar, A. Buffin, S. C. Pratt, and S. Berman, “Design of ant-inspired stochastic control policies for collective transport by robotic swarms,” *Swarm Intell.*, vol. 8, no. 4, pp. 303–327, 2014.

- [6] J. Chen, M. Gauci, W. Li, A. Kolling, and R. Groß, "Occlusion-based cooperative transport with a swarm of miniature mobile robots," *IEEE Trans. Robot.*, vol. 31, no. 2, pp. 307–321, Apr. 2015.
- [7] M. Brambilla, E. Ferrante, M. Birattari, and M. Dorigo, "Swarm robotics: A review from the swarm engineering perspective," *Swarm Intell.*, vol. 7, no. 1, pp. 1–41, 2013.
- [8] M. Chen, Z. Zhou, and C. J. Tomlin, "Multiplayer reach-avoid games via pairwise outcomes," *IEEE Trans. Autom. Control*, vol. 62, no. 3, pp. 1451–1457, Mar. 2017.
- [9] M. Coon and D. Panagou, "Control strategies for multiplayer target-attacker-defender differential games with double integrator dynamics," in *Proc. 56th IEEE Conf. Decis. Control*, 2017, pp. 1496–1502.
- [10] D. Shishika, J. Paulos, and V. Kumar, "Cooperative team strategies for multi-player perimeter-defense games," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 2738–2745, Apr. 2020.
- [11] S. Gade, A. A. Paranjape, and S.-J. Chung, "Herdin a flock of birds approaching an airport using an unmanned aerial vehicle," in *Proc. AIAA Guid., Navigation, Control Conf.*, 2015, pp. 1540–1556.
- [12] S. Gade, A. A. Paranjape, and S.-J. Chung, "Robotic herding using wavefront algorithm: Performance and stability," in *Proc. AIAA Guid., Navigation, Control Conf.*, 2016, pp. 1378–1393.
- [13] A. A. Paranjape, S.-J. Chung, K. Kim, and D. H. Shim, "Robotic herding of a flock of birds using an unmanned aerial vehicle," *IEEE Trans. Robot.*, vol. 34, no. 4, pp. 901–915, Aug. 2018.
- [14] A. Pierson and M. Schwager, "Controlling noncooperative herds with robotic herders," *IEEE Trans. Robot.*, vol. 34, no. 2, pp. 517–525, Apr. 2018.
- [15] M. A. Haque, A. R. Rahmani, and M. B. Egerstedt, "Biologically inspired confinement of multi-robot systems," *Int. J. Bio-Inspired Comput.*, vol. 3, no. 4, pp. 213–224, 2011.
- [16] A. Varava, K. Hang, D. Kragic, and F. T. Pokorny, "Herdin by caging: A topological approach towards guiding moving agents via mobile robots," in *Proc. Robot.: Sci. Syst.*, 2017.
- [17] R. A. Licitra, Z. D. Hutcheson, E. A. Doucette, and W. E. Dixon, "Single agent herding of n-agents: A switched systems approach," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 14374–14379, 2017.
- [18] R. A. Licitra, Z. I. Bell, and W. E. Dixon, "Single-agent indirect herding of multiple targets with uncertain dynamics," *IEEE Trans. Robot.*, vol. 35, no. 4, pp. 847–860, Aug. 2019.
- [19] P. Deptula, Z. I. Bell, F. M. Zegers, R. A. Licitra, and W. E. Dixon, "Single agent indirect herding via approximate dynamic programming," in *Proc. IEEE Conf. Decis. Control*, 2018, pp. 7136–7141.
- [20] S. Nardi, F. Mazzitelli, and L. Pallottino, "A game theoretic robotic team coordination protocol for intruder herding," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 4124–4131, Oct. 2018.
- [21] H. Huang, Z. Zhou, W. Zhang, J. Ding, D. M. Stipanovic, and C. J. Tomlin, "Safe-reachable area cooperative pursuit," *IEEE Trans. Robot.*, vol. 10, no. 5, pp. 10–5, Jul. 2012.
- [22] J.-M. Lien, S. Rodriguez, J.-P. Malric, and N. M. Amato, "Shepherding behaviors with multiple shepherds," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2005, pp. 3402–3407.
- [23] V. S. Chipade and D. Panagou, "Herdin an adversarial attacker to a safe area for defending safety-critical infrastructure," in *Proc. Amer. Control Conf.*, 2019, pp. 1035–1041.
- [24] V. S. Chipade and D. Panagou, "Herdin an adversarial swarm in an obstacle environment," *Proc. 58th IEEE Conf. Decis. Control*, 2019, pp. 3685–3690.
- [25] K. Kant and S. W. Zucker, "Toward efficient trajectory planning: The path-velocity decomposition," *Int. J. Robot. Res.*, vol. 5, no. 3, pp. 72–89, 1986.
- [26] V. S. Chipade and D. Panagou, "Approximate time-optimal trajectories for damped double integrator in 2D obstacle environments under bounded inputs," 2020, *arXiv:2007.05155*.
- [27] Y.-H. Liu and S. Arimoto, "Path planning using a tangent graph for mobile robots among polygonal and curved obstacles: Communication," *Int. J. Robot. Res.*, vol. 11, no. 4, pp. 376–382, 1992.
- [28] S. Akella and S. Hutchinson, "Coordinating the motions of multiple robots with specified trajectories," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2002, vol. 1, pp. 624–631.
- [29] R. Hegde and D. Panagou, "Multi-agent motion planning and coordination in polygonal environments using vector fields and model predictive control," in *Proc. Eur. Control Conf.*, 2016, pp. 1856–1861.
- [30] W. D. Esquivel and L. E. Chiang, "Nonholonomic path planning among obstacles subject to curvature restrictions," *Robotica*, vol. 20, no. 1, pp. 49–58, 2002.
- [31] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks," *IEEE Trans. Autom. Control*, vol. 52, no. 5, pp. 863–868, May 2007.
- [32] A. Mirjan, A. Federico, D. Raffaello, G. Fabio, and K. Matthias, "Building a bridge with flying robots," in *Robotic Fabrication in Architecture, Art and Design*. Cham, Switzerland: Springer, 2016, pp. 34–47.
- [33] L. Gurobi Optimization, "Gurobi optimizer reference manual," 2018. [Online]. Available: <http://www.gurobi.com>
- [34] K. G. Shin and Q. Zheng, "Minimum-time collision-free trajectory planning for dual-robot systems," *IEEE Trans. Robot. Autom.*, vol. 8, no. 5, pp. 641–644, Oct. 1992.
- [35] S. P. Bhat and D. S. Bernstein, "Continuous finite-time stabilization of the translational and rotational double integrators," *IEEE Trans. Autom. Control*, vol. 43, no. 5, pp. 678–682, May 1998.
- [36] H. K. Khalil, *Nonlinear Control*. New York, NY, USA: Pearson, 2015.
- [37] S. P. Bhat and D. S. Bernstein, "Finite-time stability of continuous autonomous systems," *SIAM J. Control Optim.*, vol. 38, no. 3, pp. 751–766, 2000.
- [38] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, "RotorS—A modular gazebo MAV simulator framework," in *Robot Operating System (ROS): The Complete Reference*, vol. 1. Berlin, Germany: Springer, 2016, pp. 595–625. [Online]. Available: [http://dx.doi.org/10.1007/978-3-319-26054-9\\_23](http://dx.doi.org/10.1007/978-3-319-26054-9_23)
- [39] T. Lee, M. Leok, and N. H. McClamroch, "Control of complex maneuvers for a quadrotor UAV using geometric methods on SE (3)," 2010, arXiv:1003.2005.



agent systems.



**Vishnu S. Chipade** (Student Member, IEEE) received the Bachelor of Technology and Master of Technology degrees in aerospace engineering from the Indian Institute of Technology Kanpur, Kanpur, India, in 2017. He is currently working toward the Ph.D. degree in aerospace engineering at University of Michigan with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, USA.

His research interests include motion planning, decision making, and collaborative control for multi-

**Dimitra Panagou** (Senior Member, IEEE) received the Diploma and Ph.D. degrees in mechanical engineering from the National Technical University of Athens, Athens, Greece, in 2006 and 2012, respectively.

She is currently an Associate Professor with the Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, USA. Prior to joining the University of Michigan, she was a Postdoctoral Research Associate with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Champaign, IL, USA, a Visiting Research Scholar with the GRASP Lab, University of Pennsylvania, Philadelphia, PA, USA, and a Visiting Research Scholar with the Mechanical Engineering Department, University of Delaware, Newark, DE, USA. Her research interests include the fields of multiagent planning, control and estimation, with applications in safe and resilient robotic networks, autonomous multivehicle systems (ground, marine, aerial, space), and human-robot interaction.

Dr. Panagou was a recipient of the NASA 2016 Early Career Faculty Award, the AFOSR 2017 Young Investigator Award, and the NSF CAREER Award in 2020. She is a senior member of the AIAA.