

# Noncooperative Herding With Control Barrier Functions: Theory and Experiments

Jaskaran Grover<sup>\*1</sup>, Nishant Mohanty<sup>\*1</sup>, Wenhao Luo<sup>2</sup>, Changliu Liu<sup>1</sup>, Katia Sycara<sup>1</sup>

**Abstract**—In this paper, we consider the problem of protecting a high-value unit from inadvertent attack by a group of agents using defending robots. Specifically, we develop a control strategy for the defending agents that we call “dog robots” to prevent a flock of “sheep agents” from breaching a protected zone. We take recourse to control barrier functions to pose this problem and exploit the interaction dynamics between the sheep and dogs to find dogs’ velocities that result in the sheep getting repelled from the zone. We solve a QP reactively that incorporates the defending constraints to compute the desired velocities for all dogs. Owing to this, our proposed framework is composable *i.e.* it allows for simultaneous inclusion of multiple protected zones in the constraints on dog robots’ velocities. We provide a theoretical proof of feasibility of our strategy for the one dog/one sheep case. Additionally, we provide empirical results of two dogs defending the protected zone from upto ten sheep averaged over a hundred simulations and report high success rates. We also demonstrate this algorithm experimentally on non-holonomic robots. Videos of these results are available at <https://tinyurl.com/4dj2kjwx>.

## I. INTRODUCTION

In the last decade, multi-robot systems (MRSs) have advanced from being researched in labs to being deployed in the real-world for solving practical problems [1], [2], [3]. The redundancy offered by an aggregated system provides resilience to faults and distributed acquisition of information. Several control algorithms have been developed that make multiple robots come together to solve team-level, global tasks using local interaction rules [4], [5]. These algorithms are (a) local (*i.e.* individual robots act on information locally available to them), (b) safe (*i.e.* result in collision-free motions amongst robots) and (c) emergent (*i.e.* global properties result from using local interaction rules) [6].

These characteristics can be treated as the insider’s perspective *i.e.* principles borne in mind by the control engineer when programming their *own* robots for a given task. Complementary to this is the outsider’s perspective *i.e.* the perspective of an external agent watching a group carry out a task by executing motions consistent with these characteristics [7]. Viewing the the motion of a group from the vantage point of an external observer is equally important. For example, if group is adversarial, potentially by posing a threat to a high-value unit, then the observer must predict the group’s motion and conscript robots (the

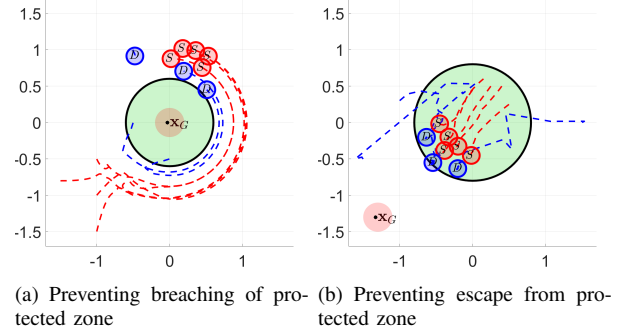


Fig. 1: Demonstration of our results showing (a) how to prevent sheep (red) from breaching a protect zone (green) and (b) preventing sheep (red) from escaping the protected zone using dog robots (blue).

defenders) to defend the unit [8], [9]. This requires the observer to orchestrate motions for their robots to prevent breach of the high-value unit. In this paper, we investigate how to develop provably correct control inputs for a group of defenders (“dog robots”) to prevent another group (the “sheep agents”) from breaching a protected zone. This is a challenging problem because the dog robots cannot directly command the actuators of the sheep agents, they must rely on their interaction dynamics (collision-avoidance behavior) with the sheep agents to influence the sheep’s behavior. This results in a non-allocated control problem. Additionally, this is also challenging because usually there are not as many defending robots as agents in the herd. Therefore, from the perspective of the dog robots, the control problem can become highly underactuated.

In this paper, we investigate how to solve this problem using ideas from control barrier functions. Specifically, we develop a centralized control technique that computes velocity inputs for the dog robots to ensure that the sheep agents do not breach a protected zone. We convert these requirements to constraints on the velocities of the dog robots. Our proposed framework is compositional in nature *i.e.* we can consider more sheep as well as more protected zones by just adding more constraints on the velocities. Furthermore, our approach relies on using automatic differentiation and symbolic computation tools, owing to which, we can easily change behavioral requirements from the sheep. For example, instead of preventing them from breaching a protected zone (Fig. 1(a)), we can prevent them from escaping a zone (Fig. 1(b)). We provide numerical results showing the success of

<sup>1</sup>J. Grover, N. Mohanty, C. Liu, K. Sycara are with the Robotics Institute at Carnegie Mellon University, 5000 Forbes Avenue, Pittsburgh, PA 15213, USA. {nishantm, jaskarag, cliu6, sycara}@andrew.cmu.edu

<sup>2</sup>W. Luo is with the Department of Computer Science, University of North Carolina at Charlotte. wenhao.luo@uncc.edu

\*Denotes Equal Contribution

our approach for multiple dogs v/s multiple sheep agents. Additionally, to test the repeatability of our algorithm, we conduct Monte Carlo simulations with increasing number of dogs and sheep averaged over 100 runs each and show high-success rates. Finally, we demonstrate our algorithm on real robots and demonstrate that we can prevent breaching of multiple zones from two sheep using one dog robot.

The outline of this paper is as follows: in section II we briefly review the prior work in this area. In section III, we give a mathematical formulation of the problem statement. In section IV, we show how to use control barrier functions to derive constraints on velocities of dog robots to pose the requirement of defense against the sheep. We consider additional collision avoidance constraints on dog robots' velocities. In section V, we provide both simulation as well as experimental results demonstrating our approach. Finally, we summarize in section VI with directions for future work.

## II. PRIOR WORK

Influencing group behavior has applications beyond just the adversarial context. For example, shepherding behaviors, specifically, are one class of flocking behaviors in which one or more external agents (called shepherds) attempt to control the motion of another group of agents (called a flock) by exerting repulsive forces on them [10], [11]. A successful practical demonstration of robotic herding was achieved in the Robot Sheepdog Project [12], [13]. Here an autonomous wheeled mobile robot (the external agent/shepherd) was used to gather a flock of ducks and manoeuvred them to a specified goal position.

Several prior works have considered the problem of non-cooperative shepherding using robots. Some of these include [11],[14],[15],[16],[17],[18]. They refer to the shepherding problem as noncooperative because the flock agents are not necessarily adversarial *i.e.* they do not work against the robots, but at the same time are not cooperative because the flock agents repel from the robots. These works exploit this repulsive interaction to develop feedback controllers for the robots to steer the flock agents to a designated region. While successful, one issue common among these works is that they fail to consider the self-motivated dynamics of the flock agents *i.e.* their nominal dynamics without any robots in the picture. As a result, the flock agents' motions are solely driven by repulsions from the robots. Additionally, these approaches tend to be handcrafted for generating a specific behavior in the sheep, for example: herding to a given location. Finally, many papers do not consider scalability with respect to the number of agents.

Differently from prior work, we do not omit the self-motivated dynamics in the sheep's motions. We synthesize the inputs for the dogs while considering cohesion, inter-sheep and dog/sheep repulsions and the sheep agents' attraction to their goal (the self-motivated part). Moreover, our proposed approach uses control barrier functions which only requires expected behaviors from sheep to be expressed as symbolic functions. Using automatic differentiation, we can generate constraints on dog velocities for any given

behavioral requirement from the sheep. Lastly, in our Monte Carlo study, we obtain high success rates even when there are many more sheep than the number of dog robots in the system. This provides an empirical evidence of scalability of our approach.

## III. PROBLEM FORMULATION

Suppose there are  $n$  "sheep" agents (the herd) and  $m$  dog robots (the defenders). We assume that the sheep are exhibiting flocking dynamics *i.e.* moving towards a common goal while staying close enough to each other and repelling from the dogs. Given this dynamics, it is possible that while en-route to their goal, they end up breaching a high-value unit *i.e.* the protected zone. From the perspective of the dogs, the sheep represent a non-cooperative group because they are not intentionally aiming towards the protected zone but may inadvertently end up breaching it. Therefore, the objective of the dog robots is to steer the sheep away from the protected zone. Let us pose this requirement mathematically.

Denote the position of the  $i^{th}$  sheep as  $\mathbf{x}_{S_i} \in \mathbb{R}^2$  and the collective positions of the herd as  $\mathbf{x}_S^{all} := (\mathbf{x}_{S_1}, \mathbf{x}_{S_2}, \dots, \mathbf{x}_{S_n})$ . Likewise, we denote the position of the  $k^{th}$  dog as  $\mathbf{x}_{D_k} \in \mathbb{R}^2$  and the collective positions of the defending robots' group as  $\mathbf{x}_D^{all} := (\mathbf{x}_{D_1}, \mathbf{x}_{D_2}, \dots, \mathbf{x}_{D_m})$ . We assume both sheep and dogs have single-integrator dynamics *i.e.* they are velocity controlled. For the  $i^{th}$  sheep, we have:

$$\begin{aligned} \dot{\mathbf{x}}_{S_i} &= \mathbf{u}_{S_i} \\ &= k_S \sum_{j \in \mathcal{S}} \left( 1 - \frac{R_S^3}{\|\mathbf{x}_{S_j} - \mathbf{x}_{S_i}\|^3} \right) (\mathbf{x}_{S_j} - \mathbf{x}_{S_i}) \\ &\quad + k_G (\mathbf{x}_G - \mathbf{x}_{S_i}) + k_D \sum_{k \in \mathcal{D}} \frac{\mathbf{x}_{S_i} - \mathbf{x}_{D_k}}{\|\mathbf{x}_{S_i} - \mathbf{x}_{D_k}\|^3} \\ &:= \mathbf{f}_i(\mathbf{x}_{S_1}, \dots, \mathbf{x}_{S_n}, \mathbf{x}_{D_1}, \dots, \mathbf{x}_{D_m}) \end{aligned} \quad (1)$$

Here the first term represents cohesion of the flock, the second represents attraction to goal and the third represents repulsion from dog robots. The attraction to the goal represents the self-motivated part of the dynamics of the sheep agents. This term is often neglected in prior work.  $R_S$  is the safety radius for sheep  $i$  to avoid collisions with the other sheep,  $\mathbf{x}_G$  is its desired goal position (common for all sheep) and  $k_S, k_G, k_D$  are proportional gains corresponding to forces in the dynamics. For each dog we have:

$$\dot{\mathbf{x}}_{D_k} = \mathbf{u}_{D_k} \quad \forall k \in \{1, 2, \dots, m\} \quad (2)$$

We denote the protected zone as  $\mathcal{P} \subset \mathbb{R}^2$  and for this paper, assume that it is a disc centered at  $\mathbf{x}_p$  and radius  $R_p$ :

$$\mathcal{P} := \{\mathbf{x} \in \mathbb{R}^2 \mid \|\mathbf{x} - \mathbf{x}_p\| \leq R_p\} \quad (3)$$

We denote the set excluding the protected zone as  $\mathcal{P}^c := \mathbb{R}^2 \setminus \mathcal{P}$ . The sheep are assumed to have no knowledge about the presence of  $\mathcal{P}$ . The dog robots need to ensure that the sheep remain in  $\mathcal{P}^c$  if they are initially in  $\mathcal{P}^c$  by finding suitable control inputs  $\{\mathbf{u}_{D_1}, \dots, \mathbf{u}_{D_m}\}$ . We make the following assumption on the dog's knowledge before posing the problem:

**Assumption 1.** *The dog robots have knowledge about the sheep's dynamics i.e. (1) and can measure the sheep's positions accurately.*

This is not a stringent assumption because if the dynamics are unknown, the dog robots can learn the dynamics online using multiagent system identification algorithms, some of which we have developed in our prior work [19], [20] and use certainty equivalence to design the controllers. We can pose the dog robots' problem as follows:

**Definition 1.** *Assuming that the initial positions of the sheep  $\mathbf{x}_S^{all}(0) \in \mathcal{P}^c$ , the dog robots' problem is to synthesize controls  $\{\mathbf{u}_{D_1}, \dots, \mathbf{u}_{D_m}\}$  such that  $\mathbf{x}_S^{all}(t) \in \mathcal{P}^c \forall t \geq 0$ . If  $\mathbf{x}_S^{all}(0) \notin \mathcal{P}^c$ , the dog robots' problem is to synthesize controls  $\{\mathbf{u}_{D_1}, \dots, \mathbf{u}_{D_m}\}$  such that  $\mathbf{x}_S^{all}(t) \rightsquigarrow \mathcal{P}^c$  in a finite time.*

Additionally, we require that the dog robots never collide with the sheep. In the next section, we show how to address this problem using control barrier functions.

#### IV. CONTROLLER DESIGN

In this section, we discuss our proposed approach to solve the problem of defending the protected zone as stated before. Given the protected zone as defined (4), we first pose the requirement for defending against one sheep, say sheep  $i$  located at  $\mathbf{x}_{S_i}$ . Subsequently, we will generalize this to the rest of the sheep in the herd. For this sheep, define a safety index  $h(\cdot) : \mathbb{R}^2 \rightarrow \mathbb{R}$  as follows:

$$h = \|\mathbf{x}_{S_i} - \mathbf{x}_P\|^2 - R_p^2 \quad (5)$$

By construction,  $h \geq 0 \forall \mathbf{x}_{S_i} \in \mathcal{P}^c$  i.e. non-negative whenever  $i$  is on the boundary or outside the protected zone. Thus, assuming that at  $t = 0$ ,  $h(\mathbf{x}_{S_i}(0)) \geq 0$ , we require  $h(\mathbf{x}_{S_i}(t)) \geq 0 \forall t \geq 0$ . Treating  $h(\cdot)$  as a control barrier function [21], this can be achieved if the derivative of  $h(\cdot)$  satisfies the following constraint:

$$\begin{aligned} \dot{h}(\mathbf{x}_{S_1}, \dots, \mathbf{x}_{S_n}, \mathbf{x}_{D_1}, \dots, \mathbf{x}_{D_m}) + p_1 h(\mathbf{x}_{S_i}) &\geq 0 \\ \implies 2(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \dot{\mathbf{x}}_{S_i} + p_1 h(\mathbf{x}_{S_i}) &\geq 0 \\ \implies 2(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \mathbf{f}_i + p_1 h(\mathbf{x}_{S_i}) &\geq 0 \end{aligned} \quad (6)$$

Define  $\mathbf{x} = (\mathbf{x}_S^{all}, \mathbf{x}_D^{all})$ , we rewrite this as

$$2(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \mathbf{f}_i(\mathbf{x}) + p_1 h(\mathbf{x}_{S_i}) \geq 0 \quad (7)$$

Here  $p_1$  is a design parameter that we choose to ensure that

$$p_1 > 0 \quad \text{and} \quad p_1 > -\frac{\dot{h}(\mathbf{x}(0))}{h(\mathbf{x}(0))} \quad (8)$$

The first condition on  $p_1$  requires that the pole is real and negative. The second depends on the initial positions  $\mathbf{x}(0)$  of all the sheep and dogs relative to the protected zone. Now while (6) depends on the positions of the sheep and dogs, it is the velocities of the dogs that are directly controllable not their positions (3). Since  $\mathbf{u}_D^{all}$  does not show up in (6), we define another function  $v(\cdot) : \mathbb{R}^{2(m+n)} \rightarrow \mathbb{R}$ :

$$v = \dot{h} + p_1 h \quad (9)$$

Like before, in order to ensure  $v \geq 0$  is always maintained, its derivative needs to satisfy

$$\dot{v}(\mathbf{x}) + p_2 v(\mathbf{x}) \geq 0. \quad (10)$$

Here  $p_2$  is another design parameter which we choose  $p_2$  to ensure that the following is satisfied at  $t = 0$

$$p_2 > 0 \quad \text{and} \quad p_2 > -\frac{\dot{h}(\mathbf{x}(0)) + p_1 \dot{h}(\mathbf{x}(0))}{\dot{h}(\mathbf{x}(0)) + p_1 h(\mathbf{x}(0))} \quad (11)$$

Using (9) in (10), we get:

$$\begin{aligned} \ddot{h}(\mathbf{x}) + (p_1 + p_2)\dot{h}(\mathbf{x}) + p_1 p_2 h(\mathbf{x}) &\geq 0 \\ \implies \ddot{h}(\mathbf{x}) + \alpha \dot{h}(\mathbf{x}) + \beta h(\mathbf{x}) &\geq 0 \end{aligned} \quad (12)$$

where we have defined  $\alpha := p_1 + p_2$  and  $\beta := p_1 p_2$ . The time derivatives of the control-barrier function  $h(\cdot)$  required in (12) are obtained as:

$$\begin{aligned} \dot{h}(\mathbf{x}) &= 2(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \dot{\mathbf{x}}_{S_i} \\ &= 2(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \mathbf{f}_i(\mathbf{x}_{S_1}, \dots, \mathbf{x}_{S_n}, \mathbf{x}_{D_1}, \dots, \mathbf{x}_{D_m}) \end{aligned} \quad (13)$$

$$\begin{aligned} \ddot{h}(\mathbf{x}) &= 2\dot{\mathbf{x}}_{S_i}^T \dot{\mathbf{x}}_{S_i} \\ &\quad + 2(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \left( \sum_{j=1}^n \mathbb{J}_{ji}^S \dot{\mathbf{x}}_{S_j} + \sum_{k=1}^m \mathbb{J}_{ki}^D \dot{\mathbf{u}}_{D_k} \right) \\ &= 2\mathbf{f}_i^T \mathbf{f}_i \\ &\quad + 2(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \left( \sum_{j=1}^n \mathbb{J}_{ji}^S \mathbf{f}_j + \sum_{k=1}^m \mathbb{J}_{ki}^D \mathbf{u}_{D_k} \right) \end{aligned} \quad (14)$$

where  $\mathbb{J}_{ji}^S$  and  $\mathbb{J}_{ki}^D$  are

$$\begin{aligned} \mathbb{J}_{ji}^S &:= \nabla_{\mathbf{x}_{S_j}} \mathbf{f}_i(\mathbf{x}_{S_1}, \dots, \mathbf{x}_{S_n}, \mathbf{x}_{D_1}, \dots, \mathbf{x}_{D_m}) \\ \mathbb{J}_{ki}^D &:= \nabla_{\mathbf{x}_{D_k}} \mathbf{f}_i(\mathbf{x}_{S_1}, \dots, \mathbf{x}_{S_n}, \mathbf{x}_{D_1}, \dots, \mathbf{x}_{D_m}) \end{aligned}$$

Note here that  $\ddot{h}(\mathbf{x})$  contains the velocities of dogs as we wanted. Using (13) and (14) in (12), we get the following linear constraints on dog velocities to ensure that the  $i^{th}$  sheep stays outside the protected zone  $\mathcal{P}$ :

$$\mathbf{A}_i^H \mathbf{u}_D^{all} \leq \mathbf{b}_i^H, \quad \text{where} \quad (15)$$

$$\begin{aligned} \mathbf{A}_i^H &:= (\mathbf{x}_P - \mathbf{x}_{S_i})^T [\mathbb{J}_{1i}^D \quad \mathbb{J}_{2i}^D \quad \dots \quad \mathbb{J}_{mi}^D] \\ \mathbf{b}_i^H &:= \mathbf{f}_i^T \mathbf{f}_i + (\mathbf{x}_{S_i} - \mathbf{x}_P)^T \sum_{j=1}^n \mathbb{J}_{ji}^S \mathbf{f}_j \\ &\quad + \alpha(\mathbf{x}_{S_i} - \mathbf{x}_P)^T \mathbf{f}_i + \beta \frac{h}{2} \end{aligned}$$

To ensure all  $n$  sheep stay away from  $\mathcal{P}$ , we compose constraints (15) for all the herd as follows:

$$\begin{bmatrix} \mathbf{A}_1^H \\ \vdots \\ \mathbf{A}_n^H \end{bmatrix} \mathbf{u}_D^{all} \leq \begin{bmatrix} \mathbf{b}_1^H \\ \vdots \\ \mathbf{b}_n^H \end{bmatrix} \implies \mathcal{A}^H \mathbf{u}_D^{all} \leq \mathbf{b}^H \quad (16)$$

Here  $\mathcal{A}^H \in \mathbb{R}^{n \times 2m}$  and  $\mathbf{b}^H \in \mathbb{R}^n$ . Given these constraints on the dogs' velocities, we can pose the following QP

that searches for the min-norm velocities that satisfies these constraints

$$\begin{aligned} \mathbf{u}_D^{*all} &= \arg \min_{\mathbf{u}_D^{all}} \|\mathbf{u}_D^{all}\|^2 \\ \text{subject to } \mathcal{A}^H \mathbf{u}_D^{all} &\leq \mathbf{b}^H \end{aligned} \quad (17)$$

Here  $\mathbf{u}_D^{*all}$  are the optimal velocities for all the dog robots to ensure both defending  $\mathcal{P}$  and collision avoidance simultaneously. By construction, our approach is centralized *i.e.* it computes velocities of all dog robots together. Future work will consider ways to decentralize this approach.

**Considering multiple protected zones:** While in the above derivation, we considered preventing the sheep from breaching only one protected zone, we can just as easily consider another protected zone by formulating similar constraints  $\mathcal{A}_2^H \mathbf{u}_D^{all} \leq \mathbf{b}_2^H$  on the dogs' velocities. By augmenting (17) with these constraints for the other zone, we will be able to defend both zones from all sheep simultaneously. This compositionality is a benefit offered by our constraint based framework. An experimental validation of this is shown in Fig. 5. In the following discussion, we prove that for the one dog v/s one sheep case, (17) is always feasible:

**Theorem 1.** *If there is one dog and one sheep, then (17) always has a solution.*

*Proof.* Let the position of the dog be  $\mathbf{x}_D$  and that of the sheep be  $\mathbf{x}_S$ . The sheep dynamics can be simplified to

$$\dot{\mathbf{x}}_S = \mathbf{f}(\mathbf{x}_S, \mathbf{x}_D) = k_G (\mathbf{x}_G - \mathbf{x}_S) + k_D \frac{\mathbf{x}_S - \mathbf{x}_D}{\|\mathbf{x}_S - \mathbf{x}_D\|^3} \quad (18)$$

The only case when (17) does not have a solution is when the defending constraint is infeasible *i.e.* when  $\mathcal{A}^H \mathbf{u}_D \leq \mathbf{b}^H$  is infeasible. This can occur when

- either when  $\mathcal{A}^H = \mathbf{0}$  and  $\mathbf{b}^H < \mathbf{0}$  (possibility 1)
- or when  $\mathbf{b}^H = -\infty$  (possibility 2).

For this case  $\mathcal{A}^H$  is:

$$\mathcal{A}^H = (\mathbf{x}_P - \mathbf{x}_S)^T \mathbb{J}_{11}^D \quad (19)$$

Thus, if  $\mathbb{J}_{11}^D$  is non-singular,  $(\mathbf{x}_P - \mathbf{x}_S)^T \mathbb{J}_{11}^D \neq \mathbf{0}$ . From our calculations, we find that the determinant of  $\mathbb{J}_{11}^D$  is

$$\det(\mathbb{J}_{11}^D) = \frac{-2k_D^2}{\|\mathbf{x}_D - \mathbf{x}_S\|^3} \quad (20)$$

As long as the distance between the dog and the sheep is finite,  $\det(\mathbb{J}_{11}^D)$  is always non zero. Thus, there exists no null space for the jacobian matrix  $\mathbb{J}_{11}^D$ . This implies  $\mathcal{A}^H \neq \mathbf{0} \forall \mathbf{x}_S \in \mathbb{R}^n, \mathbf{x}_D \in \mathbb{R}^2$ . This rules out possibility 1 for infeasibility. For possibility 2, we need to examine when does  $\mathbf{b}^H \rightarrow -\infty$ . The expression for  $\mathbf{b}^H$  is:

$$\mathbf{b}^H = \mathbf{f}^T \mathbf{f} + (\mathbf{x}_S - \mathbf{x}_P)^T \mathbb{J}_{11}^S \mathbf{f} + \alpha (\mathbf{x}_S - \mathbf{x}_P)^T \mathbf{f} + \beta \frac{h}{2}$$

We want to find the worst case lower bound of  $\mathbf{b}^H$ . Here  $\mathbf{f}^T \mathbf{f} \geq 0$  always. We assume that at the current time step, the sheep is outside the  $\mathcal{P}$ , this ensures  $\beta \frac{h}{2} \geq 0$ .

**Assumption 2.** *Assume that the following bounds hold  $\|\mathbf{x}_S - \mathbf{x}_G\| \leq M_1, \|\mathbf{x}_S - \mathbf{x}_P\| \leq M_2$  and  $\|\mathbf{x}_S - \mathbf{x}_D\| \geq M_3$ .*

With these assumptions, we can lower bound  $\mathbf{b}^H$  as follows:

$$\begin{aligned} \mathbf{b}^H &\geq (\mathbf{x}_S - \mathbf{x}_P)^T \mathbb{J}_{11}^S \mathbf{f} + \alpha (\mathbf{x}_S - \mathbf{x}_P)^T \mathbf{f} \\ &\geq -(\sigma_{max}(\mathbb{J}_{11}) + \alpha) \|\mathbf{f}\| \|\mathbf{x}_S - \mathbf{x}_P\| \\ &\geq -(\sigma_F(\mathbb{J}_{11}) + \alpha) \|\mathbf{f}\| \|\mathbf{x}_S - \mathbf{x}_P\| \end{aligned} \quad (21)$$

Here  $\|\mathbf{f}\| \leq k_G \|\mathbf{x}_S - \mathbf{x}_G\| + \frac{k_D}{\|\mathbf{x}_S - \mathbf{x}_D\|^2}$  using triangle inequality on (18). This gives  $\|\mathbf{x}_S - \mathbf{x}_P\| \|\mathbf{f}\| \leq k_G M_1 M_2 + \frac{k_D M_2}{M_3^2}$ . We can show that  $\sigma_F(\mathbb{J}_{11}) \leq \lambda_M := \sqrt{2k_G^2 + 5\frac{k_D^2}{M_3^6} + \frac{2k_G k_D}{M_3^3}}$ . Thus, using this, we obtain the following lower bound for  $\mathbf{b}^H$

$$\mathbf{b}^H \geq -(\lambda_M + \alpha) \left( k_G M_1 M_2 + \frac{k_D M_2}{M_3^2} \right) \quad (22)$$

This shows that  $\mathbf{b}^H$  is lower bounded and thus does not reach  $-\infty$ . Hence possibility 2 is also ruled out. Thus, (17) is always feasible. ■

#### A. Incorporating collision avoidance constraints

The defending constraints we posed above do not guarantee that the dog robots won't collide with the sheep. Even though the sheep dynamics have repulsions from the dogs, the velocities computed can result in aggressive behavior. Thus, we augment the defending constraints with additional constraints to ensure collision free behavior. Following the approach in [22], we define a pairwise safety index  $b^{ik}(\cdot) : \mathbb{R}^2 \times \mathbb{R}^{2m} \rightarrow \mathbb{R}$  as:

$$\begin{aligned} b^{ik}(\mathbf{x}_{S_i}, \mathbf{x}_{D_1}, \dots, \mathbf{x}_{D_m}) &= \|\mathbf{x}_{S_i} - \mathbf{x}_{D_k}\|^2 - R_S^2 \\ &= \|\mathbf{x}_{S_i} - C_k \mathbf{x}_D^{all}\|^2 - R_S^2 \end{aligned}$$

$b^{ik}(\cdot) \geq 0$  iff dog  $k$  is atleast  $R_S$  distance away from sheep  $i$ . Here  $C_k$  is a matrix defined appropriately to extract the position of the  $k^{th}$  dog from  $\mathbf{x}_D^{all}$ . If  $b^{ik}(\mathbf{x}_{S_i}(0), \mathbf{x}_D^{all}(0)) \geq 0 \forall k \in \{1, 2, \dots, m\}$ , we would like to ensure that  $b^{ik}(\mathbf{x}_{S_i}(t), \mathbf{x}_D^{all}(t)) \geq 0 \forall t \geq 0$  and  $\forall k \in \{1, 2, \dots, m\}$ . This can be achieved by requiring that

$$\dot{b}^{ik}(\mathbf{x}) + \gamma b^{ik}(\mathbf{x}) \geq 0 \quad \forall k \in \{1, 2, \dots, m\} \quad (23)$$

where  $\gamma > 0$ . This gives us a total of  $m$  linear constraints on the velocity of the dog robots for avoiding collisions with the  $i^{th}$  sheep:

$$A_i^C \mathbf{u}_D^{all} \leq \mathbf{b}_i^C \quad (24)$$

where,

$$\begin{aligned} A_i^C &= \begin{bmatrix} (\mathbf{x}_{S_i} - \mathbf{x}_{D_1})^T C_1 \\ \vdots \\ (\mathbf{x}_{S_i} - \mathbf{x}_{D_m})^T C_m \end{bmatrix} \\ b_i^C &= \begin{bmatrix} \frac{\gamma}{2} b^{i1} + (\mathbf{x}_{S_i} - \mathbf{x}_{D_1})^T \mathbf{f}_i \\ \vdots \\ \frac{\gamma}{2} b^{im} + (\mathbf{x}_{S_i} - \mathbf{x}_{D_m})^T \mathbf{f}_i \end{bmatrix} \end{aligned} \quad (25)$$



To ensure all collision avoidance with all  $n$  sheep<sup>1</sup>, we compose constraints (24) for all the herd as follows:

$$\begin{bmatrix} A_1^C \\ \vdots \\ A_n^C \end{bmatrix} \mathbf{u}_D^{all} \leq \begin{bmatrix} b_1^C \\ \vdots \\ b_n^C \end{bmatrix} \implies \mathcal{A}^C \mathbf{u}_D^{all} \leq \mathbf{b}^C \quad (26)$$

Given the defending (16) and collision avoidance (26) constraints on the dogs' velocities, we compose them together using the following QP:

$$\begin{aligned} \mathbf{u}_D^{*all} &= \arg \min_{\mathbf{u}_D^{all}} \|\mathbf{u}_D^{all}\|^2 \\ \text{subject to } & \mathcal{A}^H \mathbf{u}_D^{all} \leq \mathbf{b}^H \\ & \mathcal{A}^C \mathbf{u}_D^{all} \leq \mathbf{b}^C \end{aligned} \quad (27)$$

Here  $\mathbf{u}_D^{*all}$  are the optimal velocities for all the dog robots to ensure both defending and collision avoidance simultaneously. The cost function penalizes the total speed of the dog robots, thus encouraging them to minimize their movement.

## V. RESULTS

In this section, we show results of our approach by testing it on different scenarios consisting of varying numbers of sheep and dog and varying their initial positions. Additionally, we also run validate these results experimentally. We perform several experiments with nonholonomic Khepera robots and demonstrate how our algorithm find velocities for one dog to simultaneously defend multiple protected zones from multiple sheep.

### A. Numerical Simulation

We represent the protected zone using a circular disc with radius  $R_p$  and its center at the origin *i.e.*  $\mathbf{x}_P = \mathbf{0}$ . In our simulations, we purposefully choose the agent's goal  $\mathbf{x}_G = \mathbf{x}_P$  so that the sheep are motivated to breach the protected zone should the dog robots not interfere. Thus, this is an adversarial scenario. The initial position  $\mathbf{x}_{S_i}(0)$  of all the sheep are chosen such that they are all close to each other. This is done to ensure that the sheep have enough time to stabilize/cohere as a flock before interacting with the dog robots. The initial position  $\mathbf{x}_D^{all}(0)$  of the dog robots are chosen randomly within the area of operation. The sheep's velocities are calculated using (1). The values of the gains in the sheep dynamics were taken as  $k_G = 1$ ,  $k_S = 0.3$  and  $k_D = 0.08$ .

The velocities of the dog robot was obtained using eqn. (27). The hyperparameters  $\alpha, \beta, \gamma$  are tuned satisfy the conditions on the design parameters (8, 11). Figure 2 shows three simulation results for this behavior. In these simulations, we varied the initial position of the sheep (blue), the dog (red), number of sheep and the number of dogs. As can be noticed from the pictures, in all three scenarios, the dogs robots able to successfully intercept the sheep and prevent them from entering the protected zone while also avoiding collision with the sheep.

<sup>1</sup>inter-dog collision avoidance constraints can also be added following a similar procedure.

### B. Monte Carlo Simulations

We further study the performance of the proposed control strategy by using Monte Carlo simulations with varying initial configurations and varying number of sheep  $n$  and dog robots  $m$ . The values of the constants in sheep dynamics are  $k_G = 1$ ,  $k_S = 0.3$  and  $k_D = 0.08$ . We vary  $n$  and  $m$  from 1 to 10 and for a given pair of  $(n, m)$  we run the simulation for a hundred times with a random initialization of  $\mathbf{x}_0 = (\mathbf{x}_{S_1}(0), \dots, \mathbf{x}_{S_n}(0), \mathbf{x}_{D_1}(0), \dots, \mathbf{x}_{D_m}(0))$  in every run. Table I reports these results. Each entry of this table reports the percentage success rate *i.e.* in how many cases the sheep were diverted away from the protected zone. As can be seen, almost all entries are 100, which proves the success of our algorithm. Further, we considered the impact of including

TABLE I: Performance of the proposed strategy with varying number of sheep and dog robots. Here, we did not consider collision avoidance constraints *i.e.* the dogs were allowed to run into the sheep.

$N_S \backslash N_D$	2	4	6	8	10
2	100	100	100	100	100
4	100	100	100	100	100
6	100	98	100	100	100
8	100	98	100	100	98
10	100	98	98	100	96

collision avoidance constraints. These results are reported in Table II. Because of additional constraints, it is possible that collision avoidance conflicts with the defending constraint. As a result, we do not observe as good successes in this case compared to when there are no collision avoidance constraints.

TABLE II: Performance of the proposed strategy with varying number of sheep and dog robots. Here we considered collision avoidance constraints in the dynamics of the dogs.

$N_S \backslash N_D$	2	4	6	8	10
2	72	99	99	100	100
4	62	74	90	97	100
6	28	83	99	99	100
8	63	82	100	100	100
10	70	79	90	91	94

### C. Hardware Experiments

Finally, we tested our algorithm in robots in the multirobot test arena in our lab. It consists of a 14ft  $\times$  7ft platform, several Khepera IV robots and additionally eight Vicon cameras for motion tracking. All control inputs are computed on a desktop and conveyed to the robots over WiFi. While we developed our algorithms assuming that the dynamics of all agents are single-integrator based, the robots have unicycle dynamics given by

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} v \cos \theta \\ v \sin \theta \\ \omega \end{pmatrix} \quad (28)$$

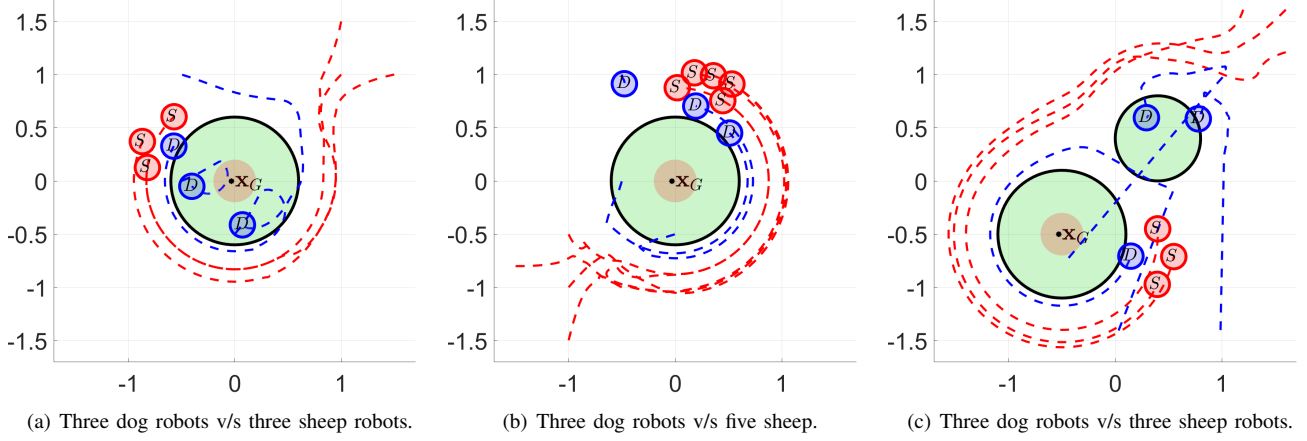


Fig. 2: Preventing the breaching of the protected zone. In these simulations, the dog is shown in blue and the sheep is shown in red. The green disc represents the protected zone. The nominal task of the red agent is to go straight towards its goal  $x_G$ . However, since this would result in infiltration of the protected zone, the dog intervenes using the control algorithm presented in (27). In 2(c), we defend two protected zones from three sheep.

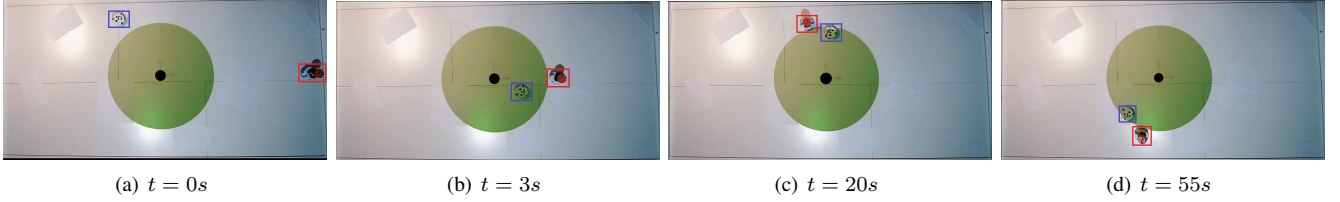


Fig. 3: Hardware experiment with one dog robot preventing one sheep from the breaching of the protected zone. The dog robot is highlighted in blue and the sheep in red. The goal position  $x_G$  is at the center of the protected zone and given as a black solid circle. The nominal task of the sheep is to go straight towards its goal  $x_G$ . However, since this would result in infiltration of the protected zone, the dog intervenes using the control algorithm presented in (17). Video at <https://tinyurl.com/2p9fjeft>.

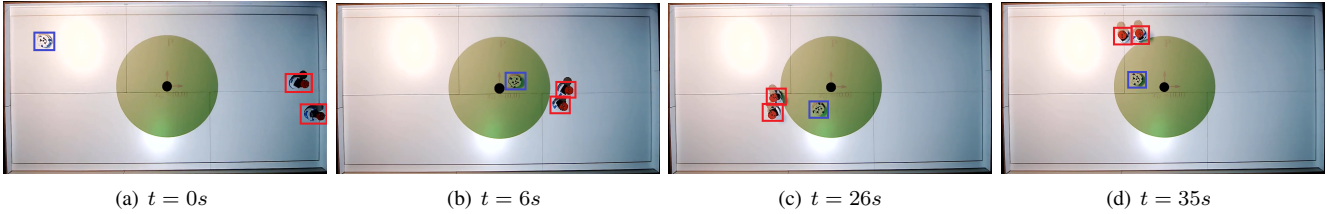


Fig. 4: Hardware experiment with one dog robot preventing two sheep from the breaching of the protected zone. The dog robot is highlighted with a blue box and sheep using a red box. The goal position  $x_G$  is at the center of the protected zone and shown as a solid black dot. Video at <https://tinyurl.com/37rduh43>.

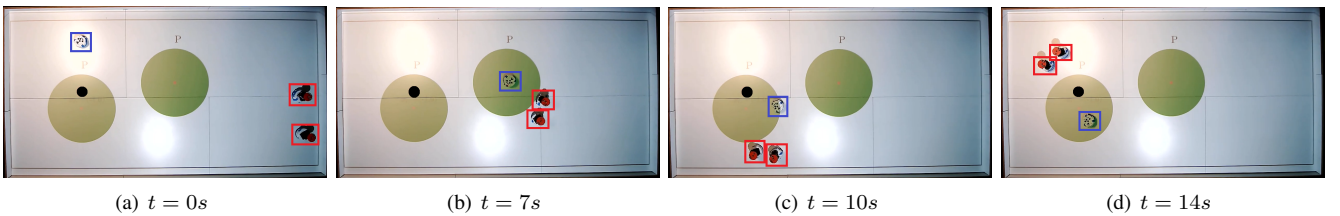


Fig. 5: Hardware experiment with one dog robot preventing two sheep from the breaching of two protected zones. The goal lies in the left most protected zone. Video at <https://tinyurl.com/ycuyhwe6>.

Thus, we do a minor adjustment to map the inputs computed from our algorithms to the angular speed and forward translational speed of these robots. This is done by considering a point at a distance  $d$  on the  $x_b$  axis of the body frame of the robot:

$$\begin{aligned} \mathbf{x} &= \begin{pmatrix} x + d \cos \theta \\ y + d \sin \theta \end{pmatrix} \\ \Rightarrow \dot{\mathbf{x}} &= \underbrace{\begin{pmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & d \end{pmatrix}}_M \begin{pmatrix} v \\ \omega \end{pmatrix} = \tilde{\mathbf{u}} \\ \Rightarrow \begin{pmatrix} v \\ \omega \end{pmatrix} &= M^{-1} \tilde{\mathbf{u}} \end{aligned} \quad (29)$$

For the robots representing the sheep,  $\tilde{\mathbf{u}}$  is obtained from (1) while for the robots representing the dog,  $\tilde{\mathbf{u}}$  is obtained from (17). In Fig. 3, we have one sheep (in red box) and one dog robot (in blue box). The protected zone is highlighted in green and the goal and center of the protected zone are the black dot. We use (17) to compute the velocity of the dog robot and convert it to angular speed and forward translational speed using (29). As can be noted from the snapshots, the dog robot is able successfully defend the zone from the sheep. Next we consider multiple sheep in Fig. 4. As can be seen from the snapshots, in this case, the dog is able to defend the zone from both sheep. Finally, in Fig. 5 we demonstrate that our approach is compositional *i.e.* we can have multiple protected zones. In this figure, we purposefully kept the goal of the sheep in the left most protected zone. This way, the sheep would be incentivized to breach both the protected zones. Yet still, our algorithm is able to find velocities for dogs to defend both the zones from both sheep.

## VI. CONCLUSIONS

In this paper, we developed a novel optimization-based control strategy for a group of dog robots to prevent a herd of sheep agents from breaching a protected zone. We have proven the feasibility of the algorithm for the single dog v/s single sheep case. Empirical results show that our designed controller can defend the protected zone from a flock of multiple sheep using multiple dogs as well. The results also show that the algorithm is composable and allows us to include multiple protected zones. Future work will focus on finding design parameters of the constraints such that the velocities computed by our controller does not exceed actuator limits. We also aim to perform hardware experiments with a higher number of sheep and dogs using Khepera robots. Further, in our current work, we assumed known dynamics of sheep. In future, we plan to extend this to the case where we learn their dynamics online while simultaneously performing defense of the protected zone.

## REFERENCES

- [1] R. D'Andrea, "Guest editorial: A revolution in the warehouse: A retrospective on kiva systems and the grand challenges ahead," *IEEE Transactions on Automation Science and Engineering*, vol. 9, no. 4, pp. 638–639, 2012.
- [2] R. D'Andrea and G. E. Dullerud, "Distributed control design for spatially interconnected systems," *IEEE Transactions on automatic control*, vol. 48, no. 9, pp. 1478–1495, 2003.

- [3] W. Kazmi, M. Bisgaard, F. Garcia-Ruiz, K. D. Hansen, and A. la Cour-Harbo, "Adaptive surveying and early treatment of crops with a team of autonomous vehicles," in *Proceedings of the 5th European Conference on Mobile Robots ECMR 2011*, 2011, pp. 253–258.
- [4] M. Ji and M. Egerstedt, "Distributed coordination control of multi-agent systems while preserving connectedness," *IEEE Transactions on Robotics*, vol. 23, no. 4, pp. 693–703, 2007.
- [5] J. Lin, A. S. Morse, and B. D. Anderson, "The multi-agent rendezvous problem-the asynchronous case," in *2004 43rd IEEE Conference on Decision and Control (CDC)(IEEE Cat. No. 04CH37601)*, vol. 2. IEEE, 2004, pp. 1926–1931.
- [6] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987, pp. 25–34.
- [7] Q. Gong, W. Kang, C. Walton, I. Kaminer, and H. Park, "Partial observability analysis of an adversarial swarm model," *Journal of Guidance, Control, and Dynamics*, vol. 43, no. 2, pp. 250–261, 2020.
- [8] C. Walton, I. Kaminer, Q. Gong, A. Clark, T. Tsatsanifos *et al.*, "Defense against adversarial swarms with parameter uncertainty," *arXiv preprint arXiv:2108.04205*, 2021.
- [9] T. Tsatsanifos, A. H. Clark, C. Walton, I. Kaminer, and Q. Gong, "Modeling and control of large-scale adversarial swarm engagements," *arXiv preprint arXiv:2108.02311*, 2021.
- [10] J.-M. Lien, O. B. Bayazit, R. T. Sowell, S. Rodriguez, and N. M. Amato, "Shepherding behaviors," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, vol. 4. IEEE, 2004, pp. 4159–4164.
- [11] A. Pierson and M. Schwager, "Controlling noncooperative herds with robotic herders," *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 517–525, 2017.
- [12] R. Vaughan, N. Sumpter, J. Henderson, A. Frost, and S. Cameron, "Robot control of animal flocks," in *Proceedings of the 1998 IEEE International Symposium on Intelligent Control (ISIC) held jointly with IEEE International Symposium on Computational Intelligence in Robotics and Automation (CIRA) Intell.* IEEE, 1998, pp. 277–282.
- [13] —, "Experiments in automatic flock control," *Robotics and autonomous systems*, vol. 31, no. 1-2, pp. 109–117, 2000.
- [14] A. Pierson and M. Schwager, "Bio-inspired non-cooperative multi-robot herding," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 1843–1849.
- [15] R. A. Licitra, Z. I. Bell, E. A. Doucette, and W. E. Dixon, "Single agent indirect herding of multiple targets: A switched adaptive control approach," *IEEE Control Systems Letters*, vol. 2, no. 1, pp. 127–132, 2017.
- [16] 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. 14 374–14 379, 2017.
- [17] E. Sebastián and E. Montijano, "Multi-robot implicit control of herds," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 1601–1607.
- [18] M. Bacon and N. Olgac, "Swarm herding using a region holding sliding mode controller," *Journal of Vibration and Control*, vol. 18, no. 7, pp. 1056–1066, 2012.
- [19] J. S. Grover, C. Liu, and K. Sycara, "Parameter identification for multirobot systems using optimization based controllers (extended version)," *arXiv preprint arXiv:2009.13817*, 2020.
- [20] J. Grover, C. Liu, and K. Sycara, "Feasible region-based identification using duality (extended version)," *arXiv preprint arXiv:2011.04904*, 2020.
- [21] A. D. Ames, S. Coogan, M. Egerstedt, G. Notomista, K. Sreenath, and P. Tabuada, "Control barrier functions: Theory and applications," in *2019 18th European Control Conference (ECC)*. IEEE, 2019, pp. 3420–3431.
- [22] L. Wang, A. D. Ames, and M. Egerstedt, "Safety barrier certificates for collisions-free multirobot systems," *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 661–674, 2017.