

Multi-robot obstacle-aware shepherding of non-cohesive target agents

Cinzia Tomaselli¹, Stefano Covone¹, Andreagiiovanni Reina^{2,3}, Mario di Bernardo^{1,4,*}

Abstract—This paper presents a novel control strategy for multi-agent shepherding of non-cohesive targets in obstacle-rich environments. Unlike previous approaches that assume cohesive flocking behavior, our method handles targets that interact only with nearby herders through repulsive forces and exhibit no inter-target coordination. Each herder employs a hybrid control policy that combines direct goal-oriented steering with obstacle-tangent maneuvering, enabling targets to circumnavigate obstacles while being guided toward a goal region. The herder dynamics integrate three key behaviors: return-to-goal motion when idle, target steering with adaptive directional control, and obstacle avoidance using both normal and tangential force components. Numerical simulations demonstrate superior performance compared to existing shepherding methods, achieving higher target confinement rates in cluttered environments. Experimental validation using TurtleBot4 herders and Osoyoo target robots in an indoor arena confirms the practical effectiveness of the proposed approach.

I. INTRODUCTION

Multi-agent shepherding represents a challenging distributed control problem where herder agents must coordinate to guide independently moving targets to desired spatial configurations [1]. This bio-inspired paradigm, observed in nature through examples like sheepdogs herding sheep [2] or orcas coordinating to hunt fish [3], has attracted significant attention in robotics due to its broad application potential in crowd control, search-and-rescue operations, area defense, and autonomous vehicle coordination (see [1] for a comprehensive review of applications).

Unlike traditional formation control or consensus problems where all agents cooperate toward common objectives, shepherding involves non-cooperative targets that move independently according to their own dynamics [4]. This fundamental difference creates unique control challenges including heterogeneous agent populations, indirect influence mechanisms [5], and the need for real-time distributed decision-making without direct target communication [6].

The transition from abstract models to real-world applications introduces several additional challenges that must be addressed to ensure effective control. Among such challenges, the presence of obstacles in the environment plays a crucial role, as it can significantly affect the motion of agents, requiring control strategies to account for complex navigation scenarios.

Most existing shepherding strategies rely on heuristic coordination rules or model-based nonlinear or optimal control approaches, e.g., [7], [5], [8]. A critical limitation is the common assumption that targets exhibit cohesive collective behavior (e.g., flocking), enabling the group to be treated as a single controllable entity [9]. However, cohesive behavior frequently fails in practical scenarios, requiring herders to influence each target individually and dramatically increasing control complexity [10], [11], [12].

This is also commonly exploited to drive targets around environmental obstacles by treating them as a cohesive group and aiming at controlling their centre of mass [7].

In this work, we extend existing shepherding strategies to the case of non-cohesive targets moving in obstacle-rich environments. Specifically, we augment the basic repulsive herder-target interaction with a hybrid control strategy that – inspired by [13] – combines repulsive and tangential forces around obstacle boundaries, enabling herders to guide targets around obstacles through circumnavigation maneuvers. This extension addresses the practical challenge of deploying shepherding algorithms in realistic environments where navigation constraints significantly impact system performance without requiring any assumption of cohesive behaviour in the targets.

A. Related work

In previous work, obstacles have been represented in shepherding problems either through explicit geometric descriptions, such as polygons with smoothed vertices or continuous boundary curves [14], or implicitly through repulsive potential fields that influence agents' motion [15]. Given these representation choices, shepherding approaches must address obstacle avoidance through task-specific strategies that ensure both herder safety during navigation and effective target steering around environmental constraints.

Existing methods adopt different computational trade-offs to achieve these dual objectives, though most assume cohesive target behavior. Hybrid planning approaches prioritize herder safety through local reactive terms while using map-based planning to guide cohesive target groups toward the goal [16], achieving near-optimal global paths at significant computational expense. Similarly, other methods employ graph-based path planning for herder navigation while steering contained cohesive targets reactively through formation-level fields [17], reducing computational overhead but increasing susceptibility to local minima and deadlocks.

Purely reactive approaches also typically assume target cohesion while employing uniform strategies for both herder navigation and target steering. Pierson and Schwager [7]

¹Scuola Superiore Meridionale, Naples, Italy

²Universität Konstanz, Konstanz, Germany

³Max Planck Institute of Animal Behavior, Konstanz, Germany

⁴Department of Electrical Engineering and Information Technology, University of Naples Federico II, Naples, Italy

* Corresponding author (mario.dibernardo@unina.it)

generate local viewpoints around obstacles to guide cohesive groups via unicycle dynamics, while Zhang et al. [8] apply identical radial potential fields to both herders and cohesive targets. These uniform approaches offer computational simplicity but may lack the specialized control needed for the distinct behavioral requirements of herders versus targets.

Learning-based approaches such as that of Zhi and Lien [18] similarly focus on cohesive target groups, employing reinforcement learning where a single herder steers the collective while avoiding obstacles. Although effective in complex scenarios, learned policies lack interpretability and may not generalize beyond training conditions.

In contrast to these methods, our strategy addresses the more challenging case of non-cohesive targets while incorporating obstacle avoidance capabilities. It combines computational efficiency with enhanced robustness by employing differentiated, reactive control strategies for herders and targets. By avoiding map-based global planning while incorporating hybrid normal-tangential obstacle avoidance, we achieve low computational overhead while reducing the deadlock susceptibility that typically affects purely repulsive potential field methods.

II. PRELIMINARIES AND PROBLEM STATEMENT

a) Notation: For any two points $\mathbf{q}, \mathbf{p} \in \mathbb{R}^2$, we denote their relative position vector as $\mathbf{d}_{\mathbf{q}\mathbf{p}} = \mathbf{q} - \mathbf{p}$. Given a closed set $B \subset \mathbb{R}^2$ with boundary ∂B and any point $\mathbf{q} \in \mathbb{R}^2$, the orthogonal projection of \mathbf{q} onto ∂B is defined as $\text{proj}_B(\mathbf{q}) = \arg \min_{\mathbf{p} \in \partial B} \|\mathbf{q} - \mathbf{p}\|$. Given two closed sets $A, B \subset \mathbb{R}^2$ with boundaries ∂A and ∂B respectively, the distance between these two sets is defined as $d(\partial A, \partial B) = \inf_{\mathbf{x} \in \partial A, \mathbf{y} \in \partial B} \|\mathbf{x} - \mathbf{y}\|$. We use $\mathbf{R}(\theta)$ to denote the standard planar rotation matrix by angle θ . When the subscript \perp is used, $\mathbf{R}_\perp(\theta)$ denotes the fixed rotation matrix corresponding to a $\theta = \pm \frac{\pi}{2}$ rad rotation (perpendicular direction).

b) Problem statement: We consider a spatial domain $\Omega \equiv \mathbb{R}^2$, populated by L obstacles, N active herder agents and M target agents, denoted by the sets \mathcal{O} , \mathcal{H} and \mathcal{T} , respectively. We denote by \mathbf{T}_a , $a = 1, \dots, M$, the cartesian coordinates of the target agents, by \mathbf{H}_i , $i = 1, \dots, N$, those of the herders and by \mathbf{P}_j , $j = 1, \dots, L$, those of the obstacles' centroid. Obstacles are assumed to be convex, static in position, with boundaries such that $d(\partial O_i, \partial O_j) > \lambda_o + \varepsilon_o \forall i, j \in \mathcal{O}, i \neq j$, where λ_o and ε_o are two positive constants. This separation constraint ensures that the influence regions of different obstacles do not overlap, simplifying the navigation dynamics and preventing conflicting repulsive forces. The goal is for the herders to corral and confine the targets within a desired region assumed to be, without lack of generality, a disc centered at the origin of radius ρ_g [10].

The target agents exhibit autonomous stochastic behavior without inherent coordination or communication capabilities. Each target $T_a \in \mathcal{T}$ is influenced by any herder only within a distance λ , forming the neighbor set defined as:

$$\mathcal{N}_H(T_a) = \{H_i \in \mathcal{H} : \|\mathbf{H}_i - \mathbf{T}_a\| < \lambda\}. \quad (1)$$

The dynamics of each target follows the model in [11], with an additional term describing the local repulsion from nearby obstacles:

$$\dot{\mathbf{T}}_a = \sqrt{2D}\mathcal{W} + \beta \sum_{i \in \mathcal{N}_H(T_a)} (\lambda - \|\mathbf{d}_{\mathbf{H}_i \mathbf{T}_a}\|) \hat{\mathbf{d}}_{\mathbf{H}_i \mathbf{T}_a} + \mathbf{F}_{a,O}^T \quad (2)$$

where \mathcal{W} is a Brownian noise term with diffusion $D > 0$, $\beta > 0$ scales the repulsive drift induced by nearby herders, and $\mathbf{F}_{a,O}^T$ is the repulsive force felt by the targets near obstacles (see Fig. 1 for a schematic representation of the forces at play).

Specifically, we set

$$\mathbf{F}_{a,O}^T = - \sum_{j=1}^L \nabla U_j(\mathbf{T}_a) \quad (3)$$

where U_j is the repulsive potential field associated with obstacle $O_j \in \mathcal{O}$.

Following [19], the repulsive potential associated with each obstacle O_j is defined, for any point $\mathbf{q} \in \mathbb{R}^2$, as

$$U_j(\mathbf{q}) = \begin{cases} \frac{k_o}{2} \left(\frac{1}{\|\mathbf{s}_j(\mathbf{q})\|} - \frac{1}{\lambda_o} \right)^2, & \text{if } \|\mathbf{s}_j(\mathbf{q})\| \leq \lambda_o, \\ 0, & \text{if } \|\mathbf{s}_j(\mathbf{q})\| > \lambda_o, \end{cases} \quad (4)$$

where $\mathbf{s}_j(\mathbf{q}) = \mathbf{q} - \text{proj}_{O_j}(\mathbf{q})$ is the vector from \mathbf{q} to its orthogonal projection on obstacle O_j 's perimeter, and $\lambda_o > 0$ denotes the obstacle influence radius. This potential is continuously differentiable for $\|\mathbf{s}_j(\mathbf{q})\| > 0$, diverges near the boundary of O_j , and vanishes outside the influence radius λ_o , thus preventing collisions while keeping the effect local.

We assume that each herder $H_i \in \mathcal{H}$ is capable of sensing all targets as well as the other herders in the environment. Furthermore, when operating within the influence zone of an obstacle, a herder is assumed to have knowledge of the obstacle's centroid and to be able to associate each point on the obstacle boundary with its corresponding centroid.

We assume the herders have a maximum speed $v_H > 0$, with dynamics given by

$$\dot{\mathbf{H}}_i = \mathbf{u}_i \quad (5)$$

where the control input \mathbf{u}_i must be designed so as to successfully drive targets towards the goal region in the presence of obstacles.

Remark 1. Note that as often done in the literature [20], [8], [10], we assume targets follow first-order dynamics under the kinematic assumption that their motion is primarily governed by instantaneous behavioral responses rather than inertial effects, which is reasonable for lightweight agents or when response times are much slower than the mechanical time constants [21]. The same assumption is also made for herders under the hypothesis that higher-level motion planning generates velocity commands that are executed by lower-level dynamic controllers [22].

III. OBSTACLE-AWARE SHEPHERDING STRATEGY

When steering targets, herders must explicitly account for obstacle presence. We extend the target selection strategy

from [11] to handle obstacle-rich environments through a two-stage process: target selection followed by obstacle-aware steering. Each herder employs the same target selection rule as in [11]: it selects the farthest target from the goal, provided that target is not closer to any other herder in the environment. When no such target exists, the herder returns to the goal region to maintain containment while simply avoiding obstacles. The key extension lies in the steering phase, where herders must guide selected targets along feasible paths that circumnavigate obstacles when direct routes are blocked by positioning themselves in appropriate steering points.

To achieve this enhanced behavior, we choose \mathbf{u}_i in (5) as

$$\mathbf{u}_i = (1 - \eta_i) \mathbf{F}_i(\mathbf{H}_i) + \eta_i \mathbf{I}'_i(\mathbf{T}, \mathbf{H}, \mathcal{O}) + \mathbf{F}_{i,O}^H \quad (6)$$

where $\eta_i \in \{0, 1\}$ indicates whether H_i is chasing a target ($\eta_i = 1$) or not ($\eta_i = 0$). (See Fig. 1 for a schematic representation of the forces acting on the herders.)

If no target is selected ($\eta_i = 0$), the herder returns to the goal region at a constant speed v_H until reaching its boundary. This is achieved by setting:

$$\mathbf{F}_i(\mathbf{H}_i) = \begin{cases} -v_H \mathbf{H}_i, & \|\mathbf{H}_i\| > \rho_g, \\ 0, & \|\mathbf{H}_i\| \leq \rho_g. \end{cases} \quad (7)$$

Thus, idling herders do not wander but regroup near the goal region to keep the targets already driven therein confined.

At the same time while navigating the environment herders need to avoid obstacles. This is achieved by setting $\mathbf{F}_{i,O}^H$ in (6) as a hybrid force combining normal and tangential components via a tunable parameter $\gamma \in [0, 1]$:

$$\mathbf{F}_{i,O}^H = - \sum_{j=1}^L \left(\gamma \nabla U_j(\mathbf{H}_i) + (1 - \gamma) R_{\perp}(\theta_i) \nabla U_j(\mathbf{H}_i) \right) \quad (8)$$

The first term provides normal repulsion using the same potential field form adopted for the targets in (4), while the second term represents an additional tangential force obtained by rotating the normal component by an angle θ_i , chosen as

$$\theta_i = \begin{cases} \frac{\pi}{2}, & \mathbf{d}_{\mathbf{H}_i, \mathbf{P}_j} \times \mathbf{d}_{\mathbf{C}_i, \mathbf{P}_j} > 0, \\ -\frac{\pi}{2}, & \mathbf{d}_{\mathbf{H}_i, \mathbf{P}_j} \times \mathbf{d}_{\mathbf{C}_i, \mathbf{P}_j} \leq 0, \end{cases} \quad (9)$$

where \mathbf{C}_i denotes the herder's steering point relative to its selected target if it is chasing a target, and $\mathbf{C}_i = \mathbf{0}$ otherwise.

The tangential component prevents deadlock situations that can occur with purely repulsive normal forces, enabling herders to slide along obstacle boundaries toward their objectives. The rotation direction is determined by the relative geometry to avoid unnecessary detours and prevent oscillations.

If a target T_i^* is selected ($\eta_i = 1$), while still avoiding obstacles through the force $\mathbf{F}_{i,O}^H$, herders also need to drive the chosen target towards the goal region. This is achieved by activating the second term on the right-hand side of (6)

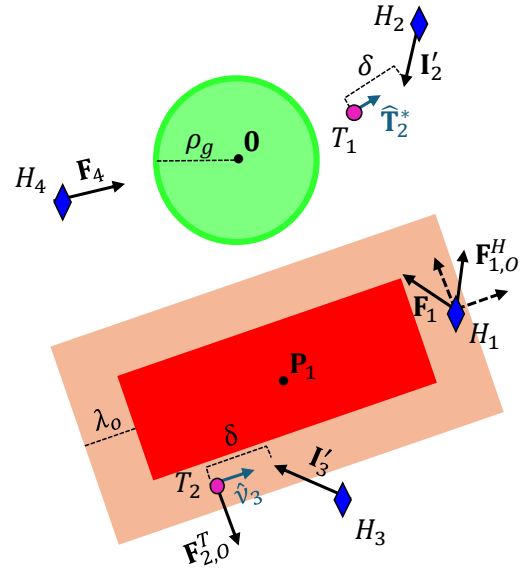


Fig. 1: Schematic representation of the forces acting on targets and herders according to (2) and (5)-(6) when moving in the environment, possibly interacting with each other and an obstacle. Herder H_1 is subject to the force \mathbf{F}_1 pulling it towards the goal region (in green) while being repelled by the obstacle (in red) through the force $\mathbf{F}_{1,O}^H$. Herder H_2 is moving towards the target T_1 in the absence of any nearby obstacle. Herder H_3 is moving towards target T_2 while avoiding the obstacle, placing itself at a tangential direction so as to make the target circumnavigate the obstacle. Herder H_4 is moving towards the goal region in the absence of nearby targets or obstacles.

which we set as:

$$\mathbf{I}'_i(\mathbf{T}, \mathbf{H}, \mathcal{O}) = -\alpha \left(\mathbf{H}_i - \left[\mathbf{T}_i^* + \delta (\mu_i \hat{\mathbf{T}}_i^* + (1 - \mu_i) \hat{\mathbf{v}}_i) \right] \right). \quad (10)$$

Here, $\alpha > 0$ denotes a control gain, $\delta < \lambda$ specifies the offset distance, $\hat{\mathbf{T}}_i^*$ is the unit vector pointing from the goal center toward the target, and $\hat{\mathbf{v}}_i$ represents the tangent direction along the obstacle boundary.

The binary variable μ_i is set to $\mu_i = 1$ for a purely straight push in the absence of nearby obstacles, and to $\mu_i = 0$ when the target is at a distance $\lambda_o + \epsilon_o$ from an obstacle blocking its path to the goal. When this happens, the herder need to push the target tangentially to the obstacle, allowing the target to avoid it while being driven towards the goal region.

The boundary-tangent direction is defined as a rotation of the vector $\mathbf{s}_j(\mathbf{T}_i^*) = \mathbf{T}_i^* - \text{proj}_{\mathcal{O}_j}(\mathbf{T}_i^*)$, obtained by setting:

$$\mathbf{v}_i = \begin{cases} R_{\perp}(-\frac{\pi}{2}) \mathbf{s}_j(\mathbf{T}_i^*), & \mathbf{d}_{\mathbf{T}_i^*, \mathbf{P}_j} \times \mathbf{d}_{\mathbf{O}_j, \mathbf{P}_j} > 0, \\ R_{\perp}(\frac{\pi}{2}) \mathbf{s}_j(\mathbf{T}_i^*), & \mathbf{d}_{\mathbf{T}_i^*, \mathbf{P}_j} \times \mathbf{d}_{\mathbf{O}_j, \mathbf{P}_j} \leq 0, \end{cases} \quad (11)$$

so that the target is consistently driven along the most favorable direction.

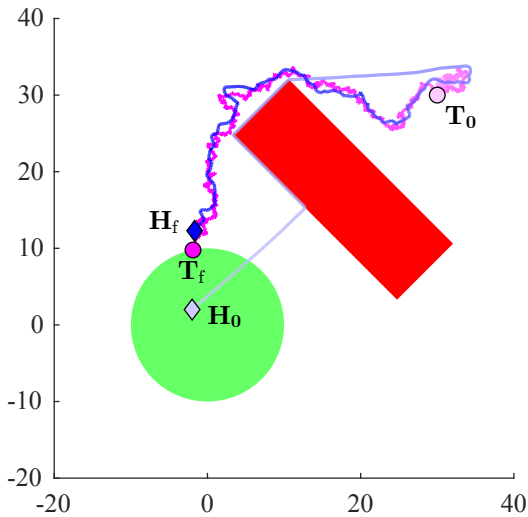


Fig. 2: Example of obstacle-avoidance strategy in a single herder and single target setup. In order to drive the target, the herder first circumnavigates the obstacle, then it pushes the target tangentially. Color gradients represent the progression of positions over time, going from $t = 0$ to $t = t_f$.

IV. NUMERICAL VALIDATION

Numerical experiments are conducted in MATLAB to evaluate the proposed obstacle-aware herding strategy.

The simulation environment consists of a circular domain $\Omega_0 \subset \Omega$ with radius $\rho_0 = 50$ m, from which initial positions of herders and targets are uniformly sampled. A single rectangular obstacle with dimensions $30\text{ m} \times 10\text{ m}$, centered at $(15\text{ m}, 15\text{ m})$, with its major axis rotated of an angle of $3\pi/4$ radians from the positive horizontal axis to create a challenging navigation scenario that blocks direct paths to the goal region.

Figure 2 illustrates a representative trajectory demonstrating the proposed obstacle circumnavigation strategy. The herder initially approaches the obstacle and utilizes the tangential force component to navigate around its boundary, maintaining forward progress toward the target. Once positioned appropriately, the herder applies repulsive forces to guide the target along a similar circumnavigation path, successfully steering it into the goal region while avoiding collision with the obstacle.

The proposed strategy scales naturally to large multi-agent scenarios through decentralized decision-making, where each herder independently selects targets using simple local rules. Fig. 3a shows the initial configuration of 10 herders and 100 targets in the presence of $L = 7$ obstacles whose width and height are independently sampled from uniform distributions over the intervals $[8, 20]$ m and $[2, 20]$ m, respectively. Under the proposed control law, all targets first enter the goal region at time 177.71 s, as illustrated in Fig. 3b. The temporal evolution of the agents' distances from the goal over the full simulation horizon is reported in Fig. 3c, confirming both convergence and subsequent containment within the goal region. The 10:100 herder-to-target ratio is based on

the herdability analysis in [11], ensuring sufficient herding capacity for the obstacle-free baseline case.

To investigate the impact of non-cohesive target behavior, we compare our method against the state-of-the-art approach by Pierson and Schwager [7]. Their method assumes cohesive targets and employs unicycle dynamics, positioning herders along an arc centered at the targets' center of mass to push the group toward the goal. For obstacle avoidance, their approach generates waypoints to guide the unicycle around obstacles. Since the original work does not specify the waypoint generation algorithm, we implement the A* path planning algorithm to ensure fair comparison through optimal path generation.

Over 50 simulations with randomly initialized conditions in an environment containing a single rectangular obstacle, our strategy consistently succeeded in herding all targets, whereas, as expected, the approach of [7] failed consistently, capturing only $\chi = 0.05 \pm 0.07$ of the targets on average. This stark performance difference demonstrates that obstacle avoidance capabilities alone are insufficient for non-cohesive targets. In this latter case, effective shepherding requires individual target consideration rather than center-of-mass control strategies designed for cohesive groups. Model parameters and simulation details are provided in the Appendix.

V. EXPERIMENTAL VALIDATION

To evaluate the effectiveness of the proposed control strategy, we conducted real-world experiments using an heterogeneous group of mobile robots comprising *TurtleBot4* as herders and *Osoyoo Robotic Car V4.0* as targets.

The *TurtleBot4*, equipped with wheel encoders and factory-calibrated odometry, provides precise motion control suitable for executing the deterministic herder control policy. The *Osoyoo Robotic Car V4.0* is a low-cost differential-drive platform based on a Raspberry Pi that lacks wheel encoders, resulting in less precise motion control. This inherent imprecision makes it an appropriate platform for representing the stochastic, non-cohesive target behavior modeled in our approach.

Experiments were carried out in a $2\text{ m} \times 2\text{ m}$ indoor arena containing a rectangular obstacle of size $0.7\text{ m} \times 0.25\text{ m}$ and a goal region of radius 0.35 m. The setup was monitored by a motion-tracking system providing ground-truth poses of all robots and obstacles. These data were processed by a central computer, resulting in a pseudo-distributed architecture: although computations were centralized, each robot acted solely on local information.

A. Model adaptation for practical implementation

The theoretical model assumes point agents governed by single-integrator dynamics. In practice, robots have finite dimensions requiring model adaptation to account for physical embodiment. Without this adaptation, agents would attempt to occupy the same spatial location, leading to unrealistic overlapping behaviors. To prevent inter-agent collisions, we introduce pairwise repulsive forces between agents of the same type. Following the obstacle avoidance potential field

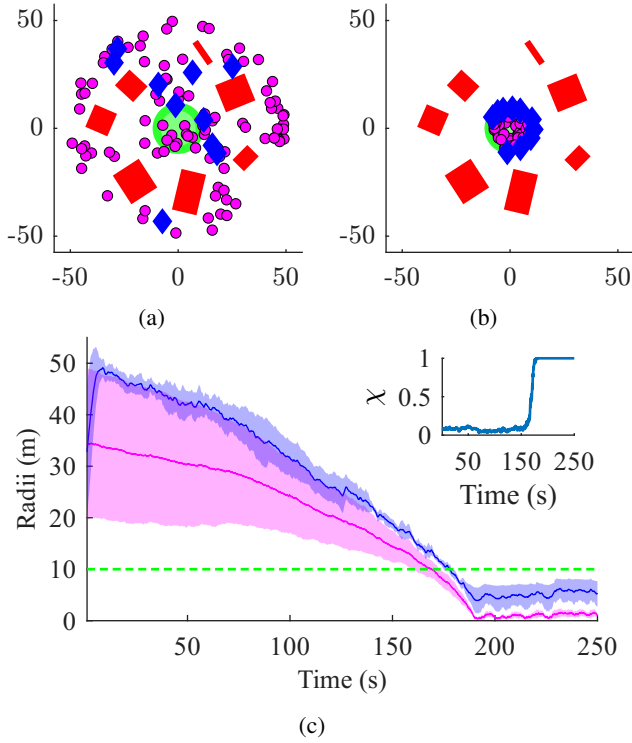


Fig. 3: Validation of the proposed shepherding strategy in a scenario with $N = 10$ herders, $M = 100$ targets and $L = 7$ rectangular obstacles with width and height uniformly sampled from $[8, 20]$ m and $[2, 20]$ m, respectively. (a) Initial positions of herders (blue diamonds) and targets (magenta points). (b) Configuration at time 177.71 s, corresponding to the first time instant at which all targets enter the goal region. (c) Temporal evolution of the mean distance from the goal region center for herders (blue) and targets (magenta), with the corresponding standard deviations (shaded areas), relative to the goal region radius (green dashed line). The inset shows the evolution of the fraction of captured targets χ , which successfully converges to unity.

design, this repulsive interaction vanishes when the inter-agent distance $\|\mathbf{d}_{\mathbf{q}_i, \mathbf{q}_j}\|$ exceeds threshold d_{th} and diverges as agents approach each other. For two agents positioned at \mathbf{q}_i and \mathbf{q}_j , the repulsive potential is:

$$V_{ij} = \begin{cases} \frac{k_d}{2} \left(\frac{1}{\|\mathbf{d}_{\mathbf{q}_i, \mathbf{q}_j}\|} - \frac{1}{d_{th}} \right)^2, & \text{if } \|\mathbf{d}_{\mathbf{q}_i, \mathbf{q}_j}\| \leq d_{th}, \\ 0, & \text{if } \|\mathbf{d}_{\mathbf{q}_i, \mathbf{q}_j}\| > d_{th}, \end{cases} \quad (12)$$

with the corresponding repulsive force $\mathbf{G}_{ij}^T = -\nabla V_{ij}$ if i and j are targets, \mathbf{G}_{ij}^H otherwise.

Consider herder pursuing its selected target T_i^* , with \mathbf{C}_i denoting the steering point relative to T_i^* . To prevent collision during approach while maintaining effective control, an orbiting term is introduced that enables the herder to circulate around the target when a direct approach is not feasible.

In the absence of nearby obstacles, the orbiting direction is determined by $\mathbf{p}_i = \alpha_o \mathbf{R}_\perp(\phi_i) \hat{\mathbf{d}}_{\mathbf{H}_i, T_i^*} +$

$\alpha_r \hat{\mathbf{d}}_{\mathbf{H}_i, T_i^*} (1 - \|\mathbf{d}_{\mathbf{H}_i, T_i^*}\|/r_{th})$, where r_{th} is a distance threshold, $\alpha_o > 0$ is the orbiting tangential gain, α_r is a radial gain regulating $\|\mathbf{d}_{\mathbf{H}_i, T_i^*}\|$ to r_{th} , and the rotation angle is given by:

$$\phi_i = \begin{cases} \frac{\pi}{2}, & \text{if } \mathbf{d}_{\mathbf{H}_i, T_i^*} \times \mathbf{d}_{\mathbf{C}_i, T_i^*} \geq 0 \\ -\frac{\pi}{2}, & \text{if } \mathbf{d}_{\mathbf{H}_i, T_i^*} \times \mathbf{d}_{\mathbf{C}_i, T_i^*} < 0 \end{cases} \quad (13)$$

The resulting orbiting direction is illustrated in Fig. 4a for the obstacle-free case. However, when the selected target T_i^* is in proximity of an obstacle, the nominal orbiting direction may induce configurations that lead to collisions during circulation. This situation is depicted in Fig. 4b. To address this issue, the rotation angle is modified as $\tilde{\phi}_i = \psi_i \phi_i$, where ψ_i is defined as follows:

$$\psi_i = \begin{cases} -1, & |\angle(\mathbf{d}_{\mathbf{H}_i, T_i^*}, \mathbf{d}_{\mathbf{C}_i, T_i^*})| > \frac{\pi}{2} \\ \wedge |\angle(-\mathbf{s}_j(T_i^*), \mathbf{d}_{\mathbf{H}_i, T_i^*})| < \frac{\pi}{2} \\ +1, & \text{otherwise.} \end{cases} \quad (14)$$

which allows the herder to circulate the target without colliding with obstacles.

The orbiting term is modulated by two scalar weights, denoted by $\sigma_i, \zeta_i \in [0, 1]$, that activate the orbiting action only when needed: the first depends on the distance between the herder and its selected target, and the second on the alignment between the vectors $\mathbf{d}_{\mathbf{H}_i, T_i^*}$ and $\mathbf{d}_{\mathbf{C}_i, T_i^*}$.

Specifically, let $r_{th} > 0$ be a distance threshold and $\varepsilon_h > 0$ a buffer distance, σ_i is given by:

$$\sigma_i = \begin{cases} 0, & \text{if } \|\mathbf{d}_{\mathbf{H}_i, T_i^*}\| \geq r_{th} + \varepsilon_h \\ 1, & \text{if } \|\mathbf{d}_{\mathbf{H}_i, T_i^*}\| \leq r_{th} \\ \frac{\|\mathbf{d}_{\mathbf{H}_i, T_i^*}\| - r_{th} - \varepsilon_h}{\varepsilon_h}, & \text{if } r_{th} < \|\mathbf{d}_{\mathbf{H}_i, T_i^*}\| < r_{th} + \varepsilon_h \end{cases} \quad (15)$$

Similarly, given $\beta_i = \angle(\mathbf{d}_{\mathbf{H}_i, T_i^*}, \mathbf{d}_{\mathbf{C}_i, T_i^*})$ and two angle thresholds β_{orb} and β_{th} such that $0 < \beta_{orb} < \beta_{th}$, ζ_i is defined as:

$$\zeta_i = \begin{cases} 0, & \text{if } |\beta_i| \leq \beta_{orb} \\ 1, & \text{if } |\beta_i| \geq \beta_{th} \\ \frac{|\beta_i| - \beta_{orb}}{\beta_{th} - \beta_{orb}}, & \text{if } \beta_{orb} < |\beta_i| < \beta_{th} \end{cases} \quad (16)$$

The region corresponding to the simultaneous saturation of both σ_i and ζ_i (i.e., $\sigma_i = \zeta_i = 1$) is highlighted in yellow in Fig. 4.

Given the additional orbiting term, $d(\partial O_i, \partial O_j) \forall i, j \in O, i \neq j$, has to be increased by d_{th} to enable target circulation.

Therefore, for experimental implementation, the control law in Eq. (6) is modified to incorporate the physical robot constraints and orbiting behavior described above:

$$\tilde{\mathbf{u}}_i = (1 - \eta_i) \mathbf{F}_i(\mathbf{H}_i) + (1 - \sigma_i) \eta_i \mathbf{I}'_i(\mathbf{T}, \mathbf{H}, \mathcal{O}) + \mathbf{F}_{i,O}^H + \sum_{j \in \mathcal{H}} \mathbf{G}_{ij}^H + \sigma_i \zeta_i \mathbf{p}_i \quad (17)$$

where the function \mathbf{I}'_i is also blended with σ_i to prevent overlapping.

In addition, given the inaccurate motion of the *Osoyoo* robots, D in Eq. (2) is set to 0 so that no additional noise is

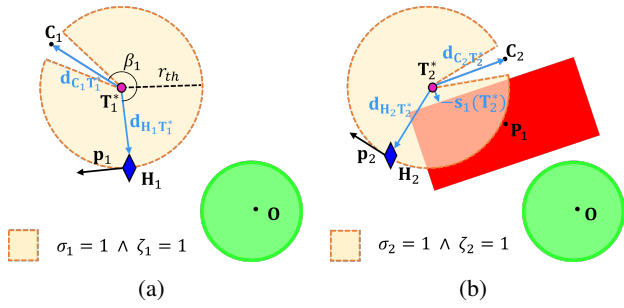


Fig. 4: Orbiting direction selection for herder H_i pursuing its assigned target T_i^* . (a) Obstacle-free case: the tangential orbiting direction is determined by the relative orientation between $\mathbf{d}_{H_i T_i^*}$ and $\mathbf{d}_{C_i T_i^*}$, yielding $\phi_i = \pm \frac{\pi}{2}$ according to Eq. (13) and enabling circulation toward the desired offset position C_i . (b) Target in proximity of obstacle O_j : the nominal orbiting direction may steer the herder toward collision-prone configurations due to the obstacle-induced constraint represented by $-\mathbf{s}_j(\mathbf{T}_i^*)$. In this case, the modulation term ψ_i , expressed in Eq. (14), reverses the orbiting direction to ensure safe circulation of T_i^* around the obstacle. The shaded regions denote configurations in which the tangential contribution in \mathbf{p}_i dominates the radial one, corresponding to the simultaneous saturation of σ_i in Eq. (15) and ζ_i in Eq. (16).

artificially introduced, and inter-agent repulsive forces are added for practical implementation. The target dynamics becomes:

$$\dot{\mathbf{T}}_a = \beta \sum_{i \in N_H(T_a)} (\lambda - \|\mathbf{d}_{H_i T_a}\|) \hat{\mathbf{d}}_{H_i T_a} + \mathbf{F}_{a,O}^T + \sum_{j \in \mathcal{F}} \mathbf{G}_{aj}^T \quad (18)$$

Once the single-integrator velocities are computed, a unicycle mapping has to be applied to obtain feasible linear and angular commands [23]. Specifically, let $\mathbf{u}_{si} = [u_x \ u_y]^T$ be the control input generated by the single-integrator model for a robot located at position $\mathbf{p} = [x \ y]^T$ with orientation φ . The nominal linear and angular velocities (v, ω) are obtained via

$$\begin{bmatrix} v_{nom} \\ \omega_{nom} \end{bmatrix} = \begin{bmatrix} \cos \varphi & \sin \varphi \\ -\frac{\sin \varphi}{d} & \frac{\cos \varphi}{d} \end{bmatrix} \mathbf{u}_{si}, \quad (19)$$

where $d > 0$ is a tuning parameter representing the look-ahead distance of the unicycle mapping. In addition, given the actuator limits of the robot wheels $|v_L|, |v_R| \leq v_{max}$ and the wheelbase l , a scaling factor s is applied:

$$s = \min \left\{ 1, \frac{v_{max}}{|v_{nom} + \frac{l}{2} \omega_{nom}| + \varepsilon}, \frac{v_{max}}{|v_{nom} - \frac{l}{2} \omega_{nom}| + \varepsilon} \right\}, \quad (20)$$

with $\varepsilon > 0$ arbitrarily small. This ensures that the commanded velocities respect the physical limits of both robot wheels while preserving the direction of the nominal control input.

B. Gazebo simulation

As an intermediate step between theoretical analysis and experimental implementation, the robotic system described

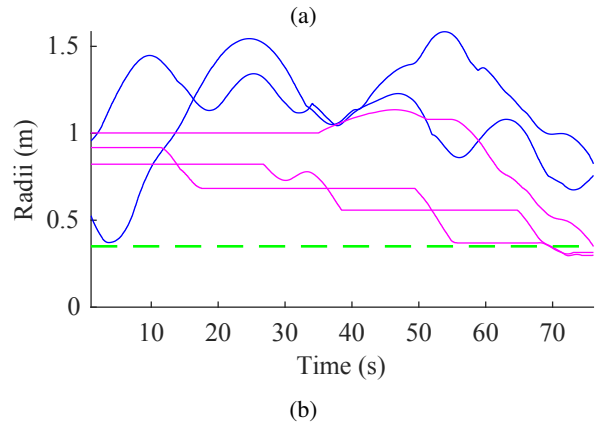
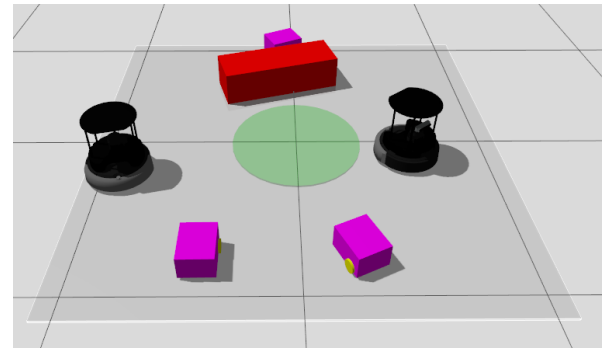


Fig. 5: Gazebo simulation of the obstacle-aware herding strategy with two *TurtleBot4* herders and three *Osoyoo* targets, controlled via a ROS 2 control architecture. (a) Snapshot from Gazebo depicting initial robot configuration. (b) Time evolution of the distances of herders (blue) and targets (magenta) from the goal center. The green dashed line denotes the goal radius ($\rho_G = 0.35$ m).

above is simulated in a Gazebo/ROS-2 environment (whose details are reported in Appendix). While the *TurtleBot4* model is publicly available, the *Osoyoo* implementation required a system identification procedure from the real robot, in order to determine its dynamical parameters.

The control strategy from Sec. III, developed on point-particles, is enhanced via the low-level controllers from the previous paragraph, to make it compliant with the differential-drive kinematics of the mobile robots.

Fig. 5a shows the initial robots' configuration in a trial with two *TurtleBot4* herders and three *Osoyoo* targets, while Fig. 5b depicts the agents' distances from the goal center over an example run, showing how herders effectively confine the targets (see supplementary videos).

C. Real robot experiments

Next, we validate the proposed strategy on the real robotic platform, using the set of parameters reported in the Appendix. In the first experiment, a single *TurtleBot4* herder and one *Osoyoo* target are deployed. The target is initially positioned behind the obstacle (Fig. 6a), and the herder successfully reaches it, guiding it toward the goal region while avoiding collisions (Fig. 6b). Figure 6c shows the

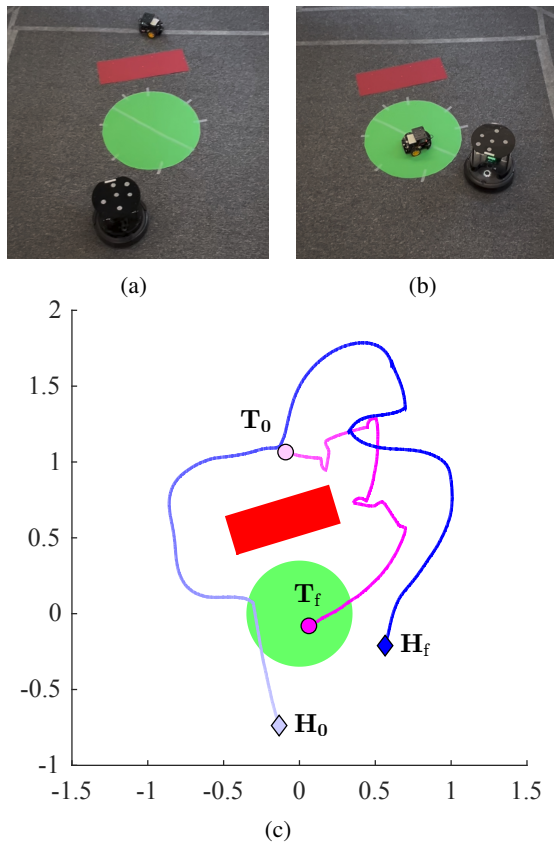


Fig. 6: Experimental validation of the obstacle-aware shepherding strategy with one *TurtleBot4* herder and one *Osoyoo* target. (a) Initial configuration with target positioned behind a rectangular obstacle. (b) Final configuration showing the target successfully contained within the goal region. (c) Robots' trajectories over time. Color gradients represent the progression of positions over time, going from $t = 0$ – agents indicated as $\mathbf{H}_0 = \mathbf{H}(0)$ and $\mathbf{T}_0 = \mathbf{T}(0)$ – to $t = t_f$ – agents indicated as $\mathbf{H}_f = \mathbf{H}(t_f)$ and $\mathbf{T}_f = \mathbf{T}(t_f)$.

trajectories of the robots: the herder's path clearly illustrates how it circumvents the obstacle by moving around it, then pushes the target tangentially along the most favorable side of the obstacle, before ultimately driving it into the goal region (see supplementary videos).

A second experiment considers a more challenging configuration with two herders and three targets, under the same obstacle placement. In this scenario, the herders autonomously select suitable targets and coordinate to effectively contain and steer the group toward the goal region.

Figures 7a–7b illustrate the initial and final configurations of the robots, while Fig. 7c highlights the success of our strategy by depicting the agents' distances from the goal center in time (see supplementary videos).

VI. CONCLUSIONS

This paper presents a control strategy for multi-agent shepherding of non-cohesive targets in obstacle-rich environments. The approach combines repulsive and tangential

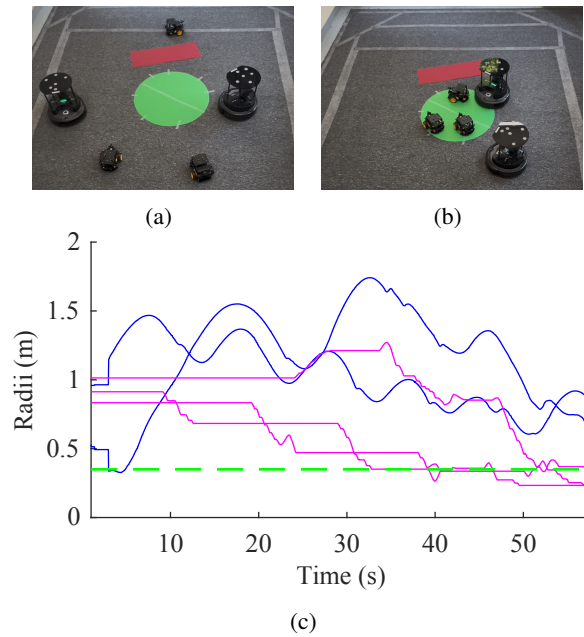


Fig. 7: Experimental validation of the obstacle-aware shepherding strategy with one *TurtleBot4* herder and one *Osoyoo* target. (a) Initial configuration with a target robot placed behind an obstacle. (b) Final configuration showing all targets successfully contained within the goal region. (c) Time evolution of the radial distances of herders (blue) and targets (magenta) from the goal center. The green dashed line denotes the goal radius ($\rho_G = 0.35$ m).

forces to enable herders to guide targets around obstacles through circumnavigation maneuvers, addressing limitations of existing methods that assume cohesive target behavior. Numerical simulations demonstrate superior performance compared to state-of-the-art approaches, achieving higher target confinement rates with improved efficiency. Experimental validation using heterogeneous robot platforms confirms practical viability under real-world constraints including actuator limitations and sensing uncertainties. Future work should address dynamic obstacles, theoretical convergence analysis, and extension to three-dimensional environments.

REFERENCES

- [1] N. K. Long, K. Sammut, D. Sgarioto, M. Garratt, and H. A. Abbass, "A comprehensive review of shepherding as a bio-inspired swarm-robotics guidance approach," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 4, no. 4, pp. 523–537, 2020.
- [2] D. Strömbom, R. Mann, A. Wilson, S. Hailes, A. Morton, D. Sumpter, and A. King, "Solving the shepherding problem: Heuristics for herding autonomous, interacting agents," *Journal of The Royal Society Interface*, vol. 11, 2014.
- [3] M. A. Haque, A. R. Rahmani, and M. B. Egerstedt, "Biologically inspired confinement of multi-robot systems," *International Journal of Bio-Inspired Computation*, vol. 3, no. 4, pp. 213–224, 2011.
- [4] C. Muro, R. Escobedo, L. Spector, and R. Coppinger, "Wolf-pack (*Canis lupus*) hunting strategies emerge from simple rules in computational simulations," *Behavioural Processes*, vol. 88, no. 3, pp. 192–197, 2011.

- [5] 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, 2018.
- [6] A. Li, M. Ogura, and N. Wakamiya, “Communication-free shepherding navigation with multiple steering agents,” *Frontiers in Control Engineering*, vol. 4, p. 989232, 2023.
- [7] A. Pierson and M. Schwager, “Controlling Noncooperative Herds with Robotic Herders,” *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 517–525, 2018.
- [8] S. Zhang, X. Lei, M. Duan, X. Peng, and J. Pan, “A distributed outmost push approach for multirobot herding,” *IEEE Transactions on Robotics*, vol. 40, pp. 1706–1723, 2024.
- [9] 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*, vol. 4, 2004, pp. 4159–4164.
- [10] F. Auletta, D. Fiore, M. J. Richardson, and M. di Bernardo, “Herding stochastic autonomous agents via local control rules and online target selection strategies,” *Autonomous Robots*, vol. 46, no. 3, pp. 469–481, 2022.
- [11] A. Lama and M. di Bernardo, “Shepherding and herdability in complex multiagent systems,” *Physical Review Research*, vol. 6, p. L032012, 2024.
- [12] D. Ko and E. Zuazua, “Asymptotic behavior and control of a “guidance by repulsion” model,” *Mathematical Models and Methods in Applied Sciences*, vol. 30, no. 04, pp. 765–804, 2020.
- [13] C. De Medio and G. Oriolo, “Robot obstacle avoidance using vortex fields,” in *Advances in Robot Kinematics*. Springer, 1991, pp. 227–235.
- [14] Y. Gong, R. Laha, and L. Figueredo, “Geopf: Infusing geometry into potential fields for reactive planning in non-trivial environments,” in *arXiv:2505.19688*, 2025.
- [15] S. Van Havermaet, Y. Khaluf, and P. Simoens, “Reactive shepherding along a dynamic path,” *Scientific Reports*, vol. 14, p. 14915, 2024.
- [16] M. Hamandi, F. Khorrami, and A. Tzes, “Robotic shepherding in cluttered and unknown environments using control barrier functions,” in *arXiv:2407.15701*, 2024.
- [17] V. S. Chipade and D. Panagou, “Multiagent planning and control for swarm herding in 2-d obstacle environments under bounded inputs,” *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1956–1972, 2021.
- [18] J. Zhi and J.-M. Lien, “Learning to herd agents amongst obstacles: Training robust shepherding behaviors using deep reinforcement learning,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 4163–4168, 2021.
- [19] O. Khatib, “Real-time obstacle avoidance for manipulators and mobile robots,” *The International Journal of Robotics Research*, vol. 5, no. 1, pp. 90–98, 1986.
- [20] E. Sebastián, E. Montijano, and C. Sagüés, “Adaptive Multirobot Implicit Control of Heterogeneous Herds,” *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3622–3635, 2022.
- [21] G. Albi, M. Bongini, E. Cristiani, and D. Kalise, “Invisible control of self-organizing agents leaving unknown environments,” *SIAM Journal on Applied Mathematics*, vol. 76, no. 4, pp. 1683–1710, 2016.
- [22] R. Olfati-Saber, “Flocking for multi-agent dynamic systems: algorithms and theory,” *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.
- [23] B. Siciliano, L. Sciacivico, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer, 2010.

APPENDIX

This appendix summarizes the key numerical settings used in the study.

a) Model parameters: Table A1 reports the physical constants defining herder and target dynamics (Section II).

TABLE A1: Model parameters for herders, targets, and environment used in the simulation study

Parameter	Notation	Value
Herder’s maximum velocity	v_h	7.5 m/s
Attraction strength to a selected target	α	5
Offset distance from selected target	δ	1.25 m
Balancing factor of obstacle avoidance for herders	γ	0.3
Influence radius of targets	λ	2.5 m
Strength of the repulsive drift from sensed herders	β	3
Obstacle safety margin	ϵ_o	1 m
Noise diffusion	D	0.5
Domain radius	ρ_0	50 m
Goal region radius	ρ_G	10 m
Obstacle influence radius	λ_o	2.5 m
Repulsion strength from obstacles	k_o	10
Sampling time	Δt	0.01 s

b) ROS 2 implementation details: To handle Vicon tracking data and control robot velocities, we employ the ROS 2 environment. A `ros_vicon_bridge` node on the central computer publishes Vicon measurements in real time to a dedicated topic for each subject, denoted as `\vicon\subject`, where `subject` corresponds to the namespace of each robot.

A `controller` node subscribes to these topics, extracts the poses of all entities, and uses them as inputs to the control law. Based on the estimated positions of robots and obstacles, this node applies the dynamics in Eq. (5) for *TurtleBot4* units and Eq. (2) for *Osoyoo* robots, accounting for their physical bodies and dynamics. The resulting linear and angular velocity commands are then published to the corresponding ROS 2 topics for execution. Table A2 reports the physical constants defining herder and target dynamics during experiments (Section V-C).

TABLE A2: Model parameters for herders, targets, and environment used in the experiments

Parameter	Notation	Value
Attraction strength to a selected target	α	3
Offset distance from selected target	δ	0.375 m
Balancing factor of obstacle avoidance for herders	γ	0.3
Orbiting tangential gain	α_o	4.5
Orbiting radial gain	α_r	3
Threshold distance for orbiting	r_{th}	0.375 m
Offset distance for orbiting blend	ϵ_h	0.1 m
Orbiting activation angle	β_{th}	$\frac{\pi}{4}$ rad
Orbiting deadband angle	β_{orb}	$\frac{\pi}{18}$ rad
Repulsion strength between robots of the same type	k_d	1
Distance threshold between robots of the same type	d_{ij}	0.45 m
Influence radius of targets	λ	0.5 m
Strength of the repulsive drift from sensed herders	β	3
Obstacle safety margin	ϵ_o	0.1 m
Noise diffusion	D	0
Obstacle influence radius	λ_o	0.2 m
Goal region radius	ρ_G	0.35 m
Repulsion strength from obstacles	k_o	10
Sampling time	Δt	0.002 s