Mediating Human-Swarm Interactions Through a Robot Helper: A Shepherding-Inspired Approach

Arihant Yadav[1], Sophie Jörg[2] and Eakta Jain[1]

***Abstract*— This work investigates human-swarm interaction (HSI) for steering large collectives through an intermediary robot helper. While prior robot-swarm control and HSI techniques commonly assume swarm members are addressable, these assumptions often fail for "ungovernable" swarms such as wildlife flocks and schools of fish. Motivated by pastoral shepherding, we compare low-level teleoperation to a higher-autonomy motion-primitives interface in a virtual reality (VR) shepherding task under varied swarm cohesion dynamics. Results indicate that motion primitives that operate on collective shape (e.g., boundary-based control) improve task performance and reduce perceived workload.**

I. INTRODUCTION

Large, naturally occurring collectives such as bird flocks and fish schools often exhibit the properties of *ungovernable swarms*: systems in which individual members cannot be pre-programmed or directly commanded, yet whose aggregate motion must still be influenced for safety or logistical reasons (e.g., airfield operations, aquaculture). At present, influencing of such swarms is largely achieved through passive or environmental interventions. In aviation, for example, wildlife hazard management commonly prioritizes habitat modification (e.g., reducing attractants such as food, water, and shelter) and exclusion methods to discourage birds from using the airfield environment [1], [2]. Only under particular contingencies do airports more frequently deploy reactive dispersal and direct-control tactics (e.g., hazing/harassment, pyrotechnics, or removal) [3], [4]. Although effective in many routine contexts, these approaches are difficult to apply with precision during time-critical disruptions. A canonical example where an "ungovernable" collective is influenced via a human-controlled helper agent is pastoral shepherding. Here, a handler issues compact, discrete commands (e.g., "come-bye", "walk-up") to a sheepdog. The sheepdog then transmits the human's intent by shaping flock-level motion. This paradigm has inspired a substantial body of algorithmic and robotic shepherding research, in which one or more agents guide swarm members [5], [6].

Translating shepherding into human-robot interaction (HRI) raises a central interface-design question: **how should the human's interaction with the helper agent be structured?** The designer must decide how much of the helper's behavior should be specified explicitly by the human versus generated autonomously by the robot, and at what level of abstraction (e.g., continuous low-level control versus discrete, task-level commands). This choice directly determines the robot's effective autonomy and, consequently, the operator's role in the control loop. In HRI, "levels of robot autonomy" (LoRA) are consequential because they shape operator workload, situation awareness, and performance. Beer et al. formalize this idea via a 10-level taxonomy spanning teleoperation through full autonomy, enabling systematic reasoning about autonomy allocation and its human-factors implications [7].

In our setting, this autonomy-allocation question manifests concretely as an interface design choice: whether the human should continuously control the helper agent at the motion level, or instead delegate execution details to the agent through higher-level commands. Guided by the shepherding metaphor, we compare (i) a teleoperation condition, which provides direct, continuous control of the helper agent, with (ii) a shepherding condition with a motion-primitives based interface. This comparison is motivated by prior work on shared autonomy and primitive-based control which argues that elevating control from low-level actuation to higher-level task/motion primitives can reduce operator burden and improve efficiency while preserving meaningful human agency [8], [9], [10], [11].



Fig. 1. The sheepdog moves on a field in response to commands by the shepherd. The sheepdog influences the movement of the sheep.

We designed a VR shepherding game in which participants control a swarm of agents using a helper agent. The swarm dynamics are instantiated using Reynolds' Boids local-interaction rules [12] augmented with a repulsion response to the helper agent. Based on LoRA theory and prior findings that primitive-based/shared-autonomy assistance can improve operator efficiency and workload outcomes relative to con-

*This work was supported in part by a research gift from Meta.

[1]Arihant Yadav & Eakta Jain are with Department of Computer & Information Science & Engineering , University of Florida, Gainesville, USA arihant.yadav@ufl.edu, ejain@ufl.edu

[2]Sophie Jörg is a Professor in the Information Systems and Applied Computer Sciences Department at University of Bamberg, Germany sophie.joerg@uni-bamberg.de

tinuous low-level teleoperation [7], [8], we hypothesize: **H1**: The shepherding (motion-primitives) condition will reduce mental workload associated with the task, at the cost of reduced direct control granularity. **H2**: Task performance will be higher in the shepherding condition, particularly under more challenging scenarios.

II. BACKGROUND

A. Shepherding as Human–Helper-Agent Control

Shepherding provides a long-standing example of influencing large collectives through a controllable intermediary. Herding dogs enable a single handler to steer mobile flocks that would be difficult to manage via direct human motion alone; this partnership has been shaped through training practices and selective breeding for responsiveness, attention, and "stock sense" [13], [14], [15], [16]. Border Collies, in particular, are widely cited as highly specialized working dogs selected primarily for herding ability [14]. Table I provides a list of commonly used commands in Anglophone countries, as reported by Long et al. [6], first formalized by Bennett et al. [17] and Coppinger et al. [14].

TABLE I
COMMON SHEEPDOG COMMANDS AND THEIR FUNCTIONAL MEANINGS IN LIVESTOCK CONTROL.

Command	Functional Meaning
Come-bye [6], [14]	Flank clockwise around stock to adjust pressure and balance relative to the handler.
Away [6], [14]	Flank counter-clockwise around stock to adjust pressure and balance relative to the handler.
Walk-up [14]	Approach directly toward the livestock to apply forward pressure and initiate movement.
Stand/Stop [6], [14]	Halt or hold position to remove or modulate pressure on the stock.
Look back [14]	Disengage and gather left-behind livestock.
That'll do / Call off [14]	Terminate work and return control focus to the handler.

Shepherding is hierarchical control: the human issues compact commands, the dog executes positioning behaviors, and the flock responds via local interaction dynamics (e.g., attraction/repulsion) [5]. Modeling and empirical work suggests that effective herding can often be decomposed into "collect" and "drive" behaviors that yield reliable group-level guidance [5], [6]. Operationally, the dog influences the flock by modulating pressure, i.e. approaching and positioning relative to the group so that livestock move away from the dog while remaining cohesive with neighbors. By shifting laterally around the flock, the dog changes the group's balance point enabling turns, containment, and directed motion without contacting or individually commanding animals. Because the dog performs fine-grained adjustments on-the-fly, the human interface remains high-level, using a small command vocabulary suited to dynamic environments [14], [16].

In swarm-control terms, pressure corresponds to a localized influence applied at the swarm boundary: the intermediary agent induces a repulsion field that propagates through local interaction rules (e.g., avoidance and cohesion), thereby steering the collective without physically acting on individuals. The balance point can be understood as the goal-relative location on the swarm perimeter where the intermediary's influence best aligns the swarm's net motion with the desired direction of travel; moving the intermediary around the perimeter shifts this effective control point and changes the group's heading [5].

B. Directing Robot Swarms: Typical Assumptions

Robotics has extensively studied the coordination and control of *robot swarms*–engineered collectives whose behavior emerges from distributed controllers and local interactions [18]. Common approaches include consensus/flocking controllers [19], leader–follower or reference-tracking schemes [20], and potential-field or artificial-physics formulations [21]. In parallel, human-swarm interaction (HSI) research examines how operators can task and steer such swarms via broadcast commands, subgroup selection, parameter tuning, or direct control of designated leaders [22]. A key limitation is that most of this literature assumes swarm members are *addressable*: individuals can be programmed, communicated with, or instrumented to execute specified policies. These assumptions fail for ungovernable swarms (e.g., wildlife, fish schools) where individuals cannot be programmed or reliably commanded.

C. Bridging the Gap via Helper-Agent Mediation

This motivates a different control pathway: rather than communicating with swarm members, the human influences collective dynamics through a controllable helper agent that interacts with the swarm in the environment. Shepherding is the canonical example of this mediation strategy.

III. METHODOLOGY

To test our hypotheses, a VR shepherding game is designed in which participants use a sheepdog-like helper agent to steer a flock of boids into a goal location. Two human–agent interaction schemes are compared: (i) a teleoperation baseline condition (continuous, low-level control of the helper agent's position and velocity), and (ii) a shepherding-based motion primitives condition (discrete commands inspired by sheepdog work). Both interfaces are evaluated under two swarm-dynamics scenarios (low vs. high cohesion) within subjects, with interface order, as well as swarm cohesion levels counterbalanced across participants.

A. Conditions

Our conditions differ in how the helper agent is controlled.

1) Condition 1: Teleoperation: For this condition, the participant directly pilots the agent's planar motion. The left controller's joystick maps to heading and forward speed of the agent in the ground plane. Neutral stick position corresponds to zero velocity; deflecting the stick away from the participant moves the agent the direction the stick is deflected towards. The agent responds continuously to joystick input, with simple first-order dynamics (bounded

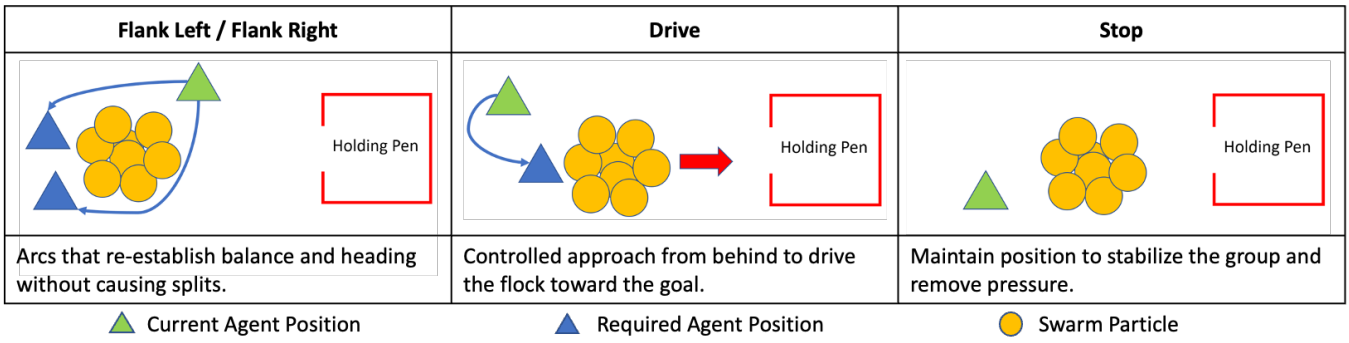


Fig. 2. Illustrations to describe shepherding inspired motion primitives implemented in this paper.

speed and turn rate). There is no automatic path planning or assistance; participants must manage approach angles, and path arcs themselves. No explicit "stop" command is required: releasing the joystick allows the agent to effectively come to a "stand" in place, removing pressure from the flock.

2) Condition 2: Motion Primitives: This condition has participants issue discrete button presses that activate pre-defined shepherding-inspired behaviors (Figure 2). Once selected, a primitive runs autonomously in closed loop until the participant switches to another command. Thus, participants specify *what* to do (e.g., flank), while the helper agent determines *how* to execute the motion in real time from the evolving flock geometry. All primitives are computed relative to the flock boundary, approximated by the convex hull at each timestep. The convex hull is the smallest convex polygon enclosing all agents; its construction and derived quantities are defined in Section III-B. **Flank-L / Flank-R:** The agent moves along a circular arc offset from the convex hull to re-establish balance without cutting through the flock. The system (i) computes the hull and centroid, (ii) defines the flock-pen axis, and (iii) selects a target on the left/right hull-offset arc corresponding to a balance position that steers the flock toward the pen. The target is continuously updated as the hull moves. These come from "come-bye" and "away" in Table I. **Drive:** This command is derived from the "walk-up" command mentioned in Table I. The agent applies sustained pressure from behind the flock by tracking a target along a rear hull segment opposite the pen. The target slides laterally along this rear arc to address hull bulges and emerging stragglers, distributing pressure across the back perimeter rather than pushing from a single point. **Stop:** The agent holds its current position, removing active pressure, until a new primitive is issued.

B. Swarm Dynamics

The swarm used in this experiment is based on Reynolds' Boids model [12], in which each particle ("boid") updates its velocity using local interaction rules. Let $\mathbf{x}_i(t)$ and $\mathbf{v}_i(t)$ denote the position and velocity of boid i at time t, and let $\mathcal{N}_i$ denote its neighborhood within a fixed perception radius. We compute the standard Boids steering terms: *separation* ($\mathbf{S}_i$), *alignment* ($\mathbf{A}_i$), *cohesion* ($\mathbf{C}_i$), and add an additional term modeling repulsion from the helper agent.

The boid velocity update used in our implementation is:

$$\mathbf{v}_i(t+1) = \mathbf{v}_i(t) + w_s\mathbf{S}_i + w_a\mathbf{A}_i + w_c\mathbf{C}_i + w_r\mathbf{R}_i, \quad (1)$$

where w_s, w_a, w_c, and w_r weight separation, alignment, cohesion, and agent repulsion, respectively.

a) Separation:

$$\mathbf{S}_i = -\sum_{j\in\mathcal{N}_i} \frac{\mathbf{x}_j - \mathbf{x}_i}{\|\mathbf{x}_j - \mathbf{x}_i\|}. \quad (2)$$

where $\mathcal{N}_i$ denotes the neighborhood of boid i within a fixed perception radius.

b) Alignment:

$$\mathbf{A}_i = \frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i} \mathbf{v}_j - \mathbf{v}_i. \quad (3)$$

c) Cohesion:

$$\mathbf{C}_i = \left(\frac{1}{|\mathcal{N}_i|}\sum_{j\in\mathcal{N}_i} \mathbf{x}_j\right) - \mathbf{x}_i. \quad (4)$$

d) Repulsion from the Helper Agent: To model the helper agent's influence, each boid experiences a distance-dependent repulsive steering term. Let $\mathbf{x}_a(t)$ denote the agent position and define $d_i = \|\mathbf{x}_i - \mathbf{x}_a\|$. Repulsion is applied only within an influence radius r_a:

$$\mathbf{R}_i = \begin{cases} \left(\frac{1}{d_i} - \frac{1}{r_a}\right)\frac{\mathbf{x}_i - \mathbf{x}_a}{d_i} & \text{if } d_i < r_a, \\ \mathbf{0} & \text{otherwise.} \end{cases} \quad (5)$$

This yields a smooth repulsive field that increases as the boid approaches the agent and vanishes at the influence boundary.

Position updates follow standard kinematic integration:

$$\mathbf{x}_i(t+1) = \mathbf{x}_i(t) + \mathbf{v}_i(t+1)\Delta t. \quad (6)$$

Such local interaction rules are sufficient to produce emergent flocking behavior consistent with biological collectives [23], [24]. For this experiment, participants interacted with two swarm configurations differing in internal cohesion strength, implemented by varying w_c. Higher values of w_c produce tightly aggregated formations resembling schooling fish [24], whereas lower values yield looser, more diffuse formations resembling avian flocks [25]. For the shepherding condition, it is necessary to estimate the spatial extent of

the swarm so that the helper agent can generate appropriate flanking arcs. To approximate the swarm boundary, we compute the convex hull of the set of boid positions at each timestep. Computationally, the hull is constructed using a planar convex hull algorithm (in this case, the QuickHull algorithm [26]). The algorithm returns an ordered set of boundary vertices $H = \{\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_m\}$, forming a convex polygon that encloses all agents.

From this boundary, we compute:

- The hull centroid:

$$\mathbf{c}_H = \frac{1}{m} \sum_{k=1}^{m} \mathbf{h}_k, \tag{7}$$

- The maximal radial extent:

$$R = \max_k \|\mathbf{h}_k - \mathbf{c}_H\|. \tag{8}$$

These geometric quantities allow the shepherding agent to compute tangential approach trajectories and maintain appropriate stand-off distance relative to the swarm envelope, enabling controlled flanking behaviors similar to biological shepherding strategies [5].

C. Task Flow

The VR game was developed using Unity 6.2 [27], and was deployed on a Meta Quest Pro headset for the experiment. The overall flow of the experiment is described in Figure 3. Each trial began with 25 boids distributed within a bounded workspace. Participants were instructed to guide a swarm of boids into a designated goal region (the holding pen). Boids entering the pen were removed from the scene, and participants received one point per successfully penned particle. A real-time scoreboard displayed the current number of penned boids, and a countdown timer indicated the remaining time.

1) Pre-Trial: Upon arrival, participants completed an informed consent form followed by a demographic questionnaire. To establish a baseline measure of potential cybersickness symptoms prior to VR exposure, participants completed the Cybersickness Questionnaire for Virtual Reality (CSQ-VR) [28]. Participants were then asked a brief set of interview questions regarding prior experience with swarm management, video games, robotics, or familiarity with shepherding practices. These responses were used to contextualize performance and subjective workload data. The complete set of interview questions is listed in Appendix A.

2) Experimental Trials: Each condition consisted of three trials, each lasting 90 seconds. Trials were initiated via a standardized trigger screen requiring participant input to begin.

1) **Training Trial:** Participants first completed a training scene designed to familiarize them with the VR environment and the assigned control interface. In this trial, the holding pen was twice the size of that used in subsequent trials to reduce initial difficulty. Participants were encouraged to explore how the flock responded to their control inputs and to refine their strategy before proceeding to scored trials.
2) **High Cohesion Trial:** This trial instantiated the high-cohesion swarm configuration. The swarm began dispersed within a bounded workspace, the holding pen was positioned near the participant's initial viewpoint, and the helper agent began at a fixed starting location. Participants were instructed to herd as many boids as possible into the pen within the time limit.
3) **Low Cohesion Trial:** This trial was structurally identical to the high-cohesion trial, with the only difference being the internal swarm cohesion parameter. Spatial layout, pen location, and initial agent positioning were held constant to isolate the effect of swarm dynamics. We expect this trial to be the more difficult trial as the swarm is less likely to stay together.

At the beginning of each trial, all dynamic entities (agent and the boids) are frozen. A start overlay displays a reminder of the control scheme, the current score (initialized to zero), and the trial duration. Participants initiate the trial by pressing the controller trigger, which simultaneously unfreezes all entities and starts the timer.

Once active, participants could use the assigned interaction scheme (teleoperation or motion primitives) to herd the flock toward the pen. Trials concluded automatically when the 90-second time limit elapsed. At trial end, the agent and swarm were frozen in their final positions to preserve the terminal configuration, and the score and timer ceased updating. The system then transitioned to the next trigger screen in the experimental sequence.

Following each trial, participants completed the NASA Task Load Index (NASA-TLX) to assess perceived workload [29]. Subsequently, a short semi-structured interview was conducted to capture qualitative impressions of strategy, perceived difficulty, and interaction usability. While a core set of questions was asked consistently, the researcher followed relevant lines of inquiry as they emerged.

3) Post-Trial: After the conclusion of all the trials, participants were engaged in a longer interview about their experiences. Following this interview, they were asked to fill out the CSQ-VR questionnaire again, upon which, they were free to leave the experiment space. The researcher also mentioned that participants were free to ask any questions about the experiment if they so wished.

D. Participant Background

A total of 24 participants were recruited for the study, primarily from a university population. The sample included individuals from diverse demographic backgrounds. In terms of gender identity, 18 participants self-identified as male, 4 as female, and 2 as non-binary. Participants also reported varied racial/ethnic backgrounds: 12 identified as White/Caucasian, 5 as East Asian, 3 as South Asian, 2 as Black, and 2 as mixed race. Participants fell within two primary age ranges: 20–29 and 30–39. The study protocol was approved by the University's Institutional Review Board (IRB) under an expedited review pathway, and was classified as minimal risk under the University IRB.

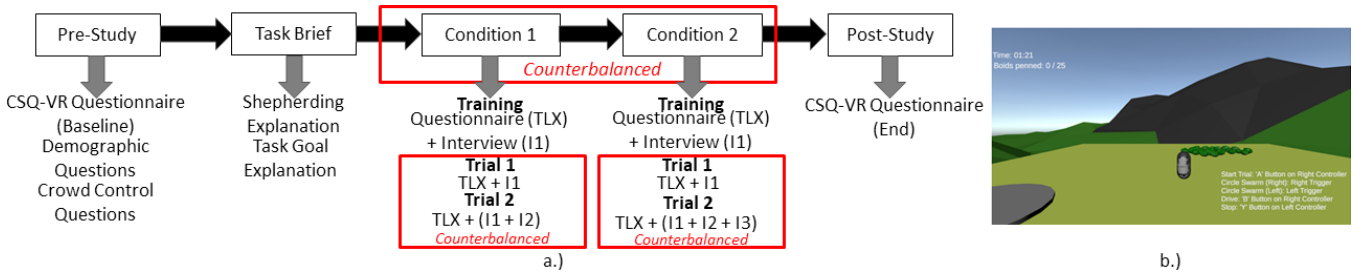


Fig. 3. (a) Flow of the experiment. (b) Experiment Setup: Grey capsule represents the agent controlled by user, green particles represent the swarm.

IV. RESULTS

We evaluated task performance, workload, and feedback from the semi-structured interview. Performance and workload were analyzed using 2×2 repeated-measures ANOVAs (Interface: Teleoperation vs. Shepherding; Cohesion: Low vs. High). Answers from the interviews were grouped by themes. For the CSQ-VR survey, no difference was observed in the pre and post-study scores for all of the participants.

A. Task Performance

The primary quantitative performance metric was the number of boids successfully guided into the holding pen by the end of each trial. As described in Section 3.3, each trial began with 25 boids. Participants were instructed to herd as many boids as possible into the pen within the allotted time; each particle that entered the pen contributed one point. The final score therefore served as the operational measure of task performance. Figure 4 presents aggregated performance across all conditions.

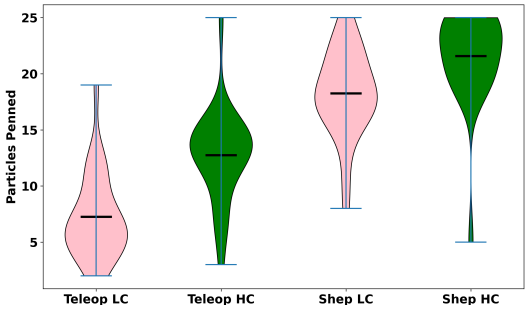


Fig. 4. Task Performance Results (LC → Low Cohesion; HC → High Cohesion; Teleop → Teleoperation; Shep → Shepherding)

The analysis showed a main effect of Interface, $F(1, 23) = 76.55$, $p < .001$. Participants achieved higher scores on average in the shepherding condition (20) than when using teleoperation (10). We also found a main effect of Cohesion, $F(1, 23) = 85.13$, $p < .001$ with more particles penned in the high cohesion scenario compared to the low cohesion scenario. There was no interaction effect ($p = 0.133$). Additionally, we found effect sizes η_p^2 of 0.769 and 0.787 for interface and cohesion, respectively using the partial eta squared metric [30].

B. Subjective Workload

Following each trial, participants completed the NASA Task Load Index (NASA-TLX) to assess subjective workload. NASA-TLX is a 21-point scale, that conceptualizes workload as a multidimensional construct spanning *Mental Demand*, *Physical Demand*, *Temporal Demand*, *Performance*, *Effort*, and *Frustration* [29], [31]. We computed an overall workload score as the mean of the six sub-scales.

Workload was analyzed using a 2×2 Repeated-Measures ANOVA (Interface: Teleoperation vs. Shepherding; Cohesion: Low vs. High). The analysis showed a significant main effect of Interface, $F(1, 23) = 33.12$, $p < .001$, effect size $\eta_p^2 = 0.590$. Cohesion and the Interface×Cohesion interaction were not significant.

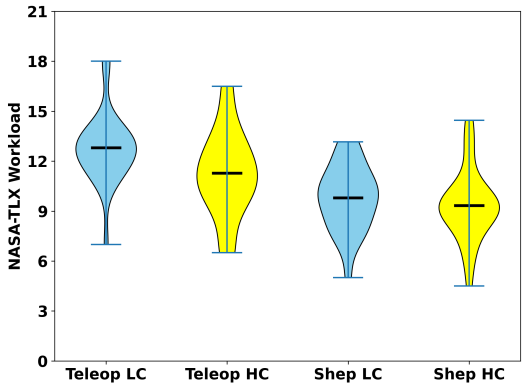


Fig. 5. NASA-TLX Results (LC → Low Cohesion; HC → High Cohesion)

Figure 5 illustrates how the shepherding interface reduces subjective workload (9.66 vs. 11.87). NASA-TLX aggregates multiple components of workload. Autonomy-related benefits may primarily reduce *Mental Demand* and *Effort* while other dimensions (e.g., *Temporal Demand*) remain similar because both conditions impose identical time limits and comparable task pacing [29]. In addition, even with motion primitives, participants must maintain situational awareness (e.g., monitoring flock shape and timing command switches) despite reduced low-level control requirements [31], [7]. Finally, short, game-like trials can compress subjective ratings, making moderate differences harder to detect even when objective performance diverges [31].

C. *Open-Ended Reflections*

Findings from the semi-structured interview questions shed light on how the two conditions were perceived. We extracted questions and themes and provide corresponding snippets from the answers.

1) Preference for Condition: Of the 24 participants, 14 preferred to use the shepherding commands, with the remaining 10 preferring to use the teleoperation commands. The primary differentiating factor for the two conditions is the amount of control users have over the agent, as well as the mental demand placed on the user during the task. Table II contains snippets from the answers to to this question.

TABLE II
INTERVIEW SNIPPETS ON PREFERENCE FOR CONTROL CONDITION

Participant ID	Snippet, When asked about what condition they prefer, and why:
p3	'the teleoperation condition, just because I like to have control over the agent at all times'
p10	'the shepherding condition, because it's easier on the shepherd and harder on the sheepdog'
p12	'the first (shepherding) condition; it seemed like the dog knew exactly what to do, and when I controlled the dog, I kept messing up'

2) Cognitive Load: Participants reported that the lower-autonomy teleoperation condition afforded finer-grained control over the helper agent, but required sustained attention to trajectory planning, spatial positioning, and continuous motion regulation. In contrast, the shepherding condition with its motion-primitives reduced the need for moment-to-moment control decisions. These subjective accounts align with the quantitative workload findings and are illustrated through interview excerpts presented in Table III.

TABLE III
INTERVIEW SNIPPETS ON COGNITIVE LOAD

Participant ID	Snippet
p6	**When asked to describe how the trial went**: '(for the shepherding condition) it went great! It was almost as if the agent knew the best way to move the swarm'
p15	**When asked about how demanding the two trials were compared to each other**: 'in the first condition (teleoperation), I felt like I was responsible for the dog's movement at all times and I couldn't take a break. the second condition was easier because i was more relaxed'

3) Participant Ideas for Control Schemes: As part of the post-trial assessment, participants were invited to propose an ideal control scheme after having experienced both interaction conditions. This open-ended prompt was intended to elicit reflections on interface preferences and perceived trade-offs. A majority of participants (14/24) suggested a hybrid approach that combined elements of continuous teleoperation with discrete motion primitives. A smaller subset indicated satisfaction with the shepherding-based motion-primitives condition as implemented. Additionally, a few participants proposed novel control strategies that were not represented in either experimental condition. Representative excerpts illustrating these perspectives are presented in Table IV.

TABLE IV
INTERVIEW SNIPPETS ON CONTROL SCHEMES

Participant ID	Snippet
p5	**When asked what their ideal control scheme for this task would be**: 'I like the sheepdog control scheme, but the arcs the dog made around the swarm were too tight sometimes. I would like for there to be an additional button that helps me fine-tune the arc radius'
p13	**When asked what changes they would make to the agent to make the task easier**: 'I think having multiple helpers instead of a single helper would be good, because I can then send helpers to round up split groups'

V. DISCUSSION

Results obtained from this experiment support our hypotheses or at least parts of them. We discuss these results below and provide actionable implications for HSI.

Autonomy Allocation and Task Performance The shepherding condition using motion-primitives yielded higher task performance than the teleoperation condition, supporting the first part of H2. We did not find this effect to be stronger in a more challenging condition involving low-cohesion swarm dynamics. In contrast, teleoperation performance varied significantly across swarm conditions, suggesting greater sensitivity to environmental instability. These findings indicate that structured autonomy can improve robustness. By encoding hull-based flanking and distributed rear pressure into motion primitives, the system embedded task-relevant geometric reasoning that participants otherwise had to compute manually. This aligns with shared autonomy literature demonstrating that abstraction at the task level can enhance performance while preserving user agency [8], [9].

Cognitive Load and Control Granularity NASA-TLX results indicate reduced workload in the shepherding (motion-primitives) condition, supporting H1. Qualitative data indicate that teleoperation required sustained spatial reasoning, trajectory planning, and continuous regulation of pressure on the swarm. This reflects a control granularity trade-off. Teleoperation provides maximal control authority but increases cognitive demand. Motion primitives reduce granularity by delegating micro-adjustments to the system. Consistent with LoRA theory [7], autonomy improved efficiency by reallocating control from low-level actuation to task-level intent specification. Notably, several participants described the primitive-based agent as a "helper," suggesting that autonomy allocation influences perceived collaboration as well as workload.

Hybrid Control Participants proposed control schemes that allow the user to override motion primitives with direct teleoperation as desired. Such interfaces would preserve the efficiency benefits of abstraction while giving users fine control in cases such as swarm splits or unexpected dispersal.

Swarm Fragmentation We found that our boids-based swarm split into smaller groups in the low cohesion trials.

Even if a single boid splits off, it can skew the centroid computation. We modified the base centroid computation algorithm so that sub-groups of less than three members were ignored in the centroid computation. In the field, shepherds often use multiple sheepdogs to prevent their herds from splitting into sub-groups.

Levels of Autonomy The two conditions designed for this experiment are chosen to cover the span of the levels of autonomy possible with currently available technology. Additional levels, such as teleoperation with low-level driving assistance (for example, velocity damping or collision avoidance) could have been added to bridge the gap between the two ends of the spectrum used here. We limited our experiment to two conditions to keep it concise and clean.

Implications for HSI Our results suggest three design implications for HSI:

Encode task structure, not just low-level automation: The primitives benefited users primarily because they embedded shepherding-relevant structure (e.g., flank geometry, behind-the-flock driving) that users otherwise had to continuously plan and maintain in the teleoperation condition. This was most consequential under low-cohesion dynamics, where instability increased the cost of manual trajectory synthesis. *Implication:* Autonomy should externalize invariants of effective swarm influence ("where to be" and "how to apply pressure") not merely reduce motor effort.

Operate on swarm geometry rather than individuals: By defining control actions relative to the swarm envelope (via convex-hull-based arcs and rear-boundary driving), the interface provided stable control handles despite deformation and splitting. *Implication:* HSI interfaces should expose geometric abstractions (boundary, centroid, heading, rear region) and support boundary-referenced actions that remain meaningful as the swarm changes shape.

Support adjustable autonomy with targeted override: Participants proposed hybrids to handle edge cases (e.g., splits, overly tight arcs). *Implication:* Primitives should be complemented by parameter tuning (e.g., arc radius) and/or temporary direct control, enabling users to intervene according to what works for them.

VI. LIMITATIONS AND FUTURE WORK

Our study was conducted in a controlled virtual environment using simulated Boids dynamics. While this type of local interaction models are widely used to study collective behavior [12], [23], [24], real-world swarms (e.g., animal groups, or adversarial drone swarms) may exhibit additional behavioral constraints, sensing limitations, and noise. Consequently, future work should validate the proposed motion-primitives framework on embodied robotic platforms.

Additionally, the experiment task involved a single helper agent interacting with a fixed swarm size (25 agents). Real-world scenarios often involve varying swarm sizes and multiple agents. Extending this framework to multi-helper settings and more variable swarms may reveal new coordination dynamics and interface challenges.

The motion-primitives condition implemented a fixed set of discrete behaviors. While this abstraction reduced workload and improved robustness, it may have constrained user expressiveness in edge cases, such as swarm fragmentation. Future research should investigate dynamic autonomy allocation models [7], where users can fluidly transition between primitive-based control and direct teleoperation. Such adaptive autonomy systems may balance efficiency with flexibility, particularly in ungovernable swarms.

Workload was assessed using NASA-TLX [29], a validated subjective instrument. However, subjective workload does not capture fine-grained cognitive processes such as attention allocation or strategy formation. Future studies may incorporate behavioral metrics (e.g., control smoothness) or physiological measures (e.g., eye tracking, heart-rate variability) to more precisely quantify cognitive effort.

Finally, this study focused on short-duration, laboratory-scale tasks. Real-world deployment would require consideration of long-term user adaptation, trust calibration, and failure recovery mechanisms. Investigating how users learn to collaborate with autonomy over time, and how trust evolves under varying swarm volatility, represents an important direction for future HRI research.

Swarm Dynamics The swarm designed for this experiment is based on the Boids principles, and it is assumed that the particles, as well as helpers, are purely kinematic bodies. Factors such as sliding friction, communication latency between controller and agent, etc. are not considered. This is a deviation from technology being used in the field currently, and thus, real-world testing would help validate the findings. Furthermore, parameters for the boids' as well as helper's movement were derived through a mix of testing, as well as taking inspiration from shepherding (by reflecting the movement speeds observed in shepherding). These worked well for the VR task, but real-world deployment may need further fine tuning depending on the swarm's characteristics.

Convex Hull Computation We assume that all swarm particles are perfectly visible for computing the convex hull. In real-life scenarios, occlusions will prevent the real-time location of all swarm members from being known at all times. Additionally, the hull-based primitives introduce an $O(n \log n)$ QuickHull computation per update for n boids. This cost is negligible for our 25-boid VR task, but larger swarms could use lower-rate updates, incremental hulls, and other techniques, instead of recomputing a hull every frame.

Workspace Our experiment workspace is flat and without obstructions. Introducing static obstacles inside the workspace could lead to more realistic trials. Although this idea was not tested as part of this study, it is an avenue worth exploring in future work. Previous work has shown that higher autonomy can reduce operator workload and our findings are consistent with those findings.

The significance of our study, however, is not simply that autonomy outperforms teleoperation, but that the trade-off between agent autonomy and user's control of the agent is examined for swarms where individual members cannot be directly controlled. In this context, the key question is

how autonomy should be structured to let a human influence the collective indirectly through a helper agent. Our results suggest that shepherding-inspired primitives provide an effective abstraction for manipulating the swarm, improving performance and reducing workload in human-guided control of ungovernable swarms.

APPENDIX

A. Semi-Structured Interview Questions

1) Have you ever had to manage a group of agents before (eg., animals, insects, etc.)?
2) Have you heard of shepherding before? If so, could you describe what the term means to you please?
3) In your own words, can you describe how this trial went?
4) How easy or difficult was it to learn the controls for this condition?
5) Did you notice any difference between the swarm behavior for all the trials?
6) For the task you just did, what condition did you prefer, and why?
7) What would be your ideal control scheme for this task?

Questions 1 and 2 were asked as part of the Pre-Study. Question 3 was asked at the conclusion of each trial (I1 in Figure 3). Questions 3 and 4 were asked after completion of each condition (I2 in Figure 3) Questions 6 and 7 were asked after completion of both the conditions (I3 in Figure 3).

ACKNOWLEDGMENT

Generative AI (ChatGPT) was used for organizing the Methodology section for conciseness and during the task design process for help with scripting in Unity. ChatGPT was also used for generating Figure 1.

REFERENCES

[1] Edward C. Cleary and Richard A. Dolbeer. Wildlife hazard management at airports. Technical report, Federal Aviation Administration (FAA), 2005.

[2] Bradley F. Blackwell, Travis L. DeVault, Esteban Fernández-Juricic, and Richard A. Dolbeer. Wildlife collisions with aircraft: A missing component of land-use planning for airports. *Landscape and Urban Planning*, 93(1):1–9, 2009.

[3] Jerrold L. Belant and Janet A. Martin. Bird harassment, repellent, and deterrent techniques for use on and near airports. Technical report, Transportation Research Board (ACRP Synthesis 23), 2011.

[4] ACRP Report 32: Guidebook for addressing aircraft/wildlife hazards at general aviation airports. Technical report, Airport Cooperative Research Program, Transportation Research Board, 2010.

[5] Daniel Strömbom, Richard P. Mann, Andrew M. Wilson, Stephen Hailes, A. James Morton, David J. T. Sumpter, and Andrew J. King. Solving the shepherding problem: Heuristics for herding autonomous interacting agents. *Journal of the Royal Society Interface*, 11(100):20140719, 2014.

[6] Nathan K. Long, Karl Sammut, Daniel Sgarioto, Matthew Garratt, and Hussein A. Abbass. A comprehensive review of shepherding as a bio-inspired swarm-robotics guidance approach. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(4):523–537, 2020.

[7] Jenay M. Beer, Arthur D. Fisk, and Wendy A. Rogers. Toward a framework for levels of robot autonomy in human–robot interaction. *Journal of Human-Robot Interaction*, 3(2):74–99, 2014.

[8] Shervin Javdani, Siddhartha S. Srinivasa, and J. Andrew Bagnell. Shared autonomy via hindsight optimization for teleoperation and human–robot interaction. In *Proceedings of Robotics: Science and Systems (RSS)*, 2018.

[9] Deepak Gopinath, Shikhar Jain, and Brenna D. Argall. Human-in-the-loop optimization of shared autonomy in assistive robotics. *IEEE Robotics and Automation Letters*, 2(1):247–254, 2016.

[10] Auke Jan Ijspeert, Jun Nakanishi, Heiko Hoffmann, Peter Pastor, and Stefan Schaal. Dynamical movement primitives: Learning attractor models for motor behaviors. *Neural Computation*, 25(2):328–373, 2013.

[11] C. E. Mower and S. Vijayakumar. Modulating human input for shared autonomy in dynamic environments. In *Proc. of RO-MAN*, 2019.

[12] Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. In *Proceedings of SIGGRAPH*, pages 25–34, 1987.

[13] James Serpell, editor. *The Domestic Dog: Its Evolution, Behaviour and Interactions with People*. Cambridge University Press, 1995.

[14] Raymond Coppinger and Lorna Coppinger. *Dogs: A New Understanding of Canine Origin, Behavior and Evolution*. University of Chicago Press, 2001.

[15] Evan L. MacLean, Brian Hare, Charles L. Nunn, et al. The evolution of self-control. *Proceedings of the National Academy of Sciences*, 111(20):E2140–E2148, 2014.

[16] Emily E. Bray, Evan L. MacLean, and Brian Hare. Early-emerging and highly heritable sensitivity to human communication in dogs. *Current Biology*, 31(14):3132–3136, 2021.

[17] Brandon Bennett and Matthew Trafankowski. A comparative investigation of herding algorithms. *AISB/IACAP World Congress 2012: Understanding and Modelling Collective Phenomena, Part of Alan Turing Year 2012*, 01 2012.

[18] Essam Debie, Kathryn Kasmarik, and Matt Garratt. Swarm robotics: A survey from a multi-tasking perspective. *ACM Comput. Surv.*, 56(2), September 2023.

[19] Reza Olfati-Saber. Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control*, 51(3):401–420, 2006.

[20] Weixu Zhu, Sinan Oğuz, Mary Katherine Heinrich, Michael Allwright, Mostafa Wahby, Anders Lyhne Christensen, Emanuele Garone, and Marco Dorigo. Self-organizing nervous systems for robot swarms. *Science Robotics*, 9(96):eadl5161, 2024.

[21] Veysel Gazi and Kevin M. Passino. *Swarm Stability and Optimization*. Springer, 2011.

[22] Joseph B. Lyons, August Capiola, Julie A. Adams, Janine D. Mator, Erin Cherry, and Kristen Barrera. Examining the human-centred challenges of human-swarm interaction. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 383(2289):20240140, 2025.

[23] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75(6):1226–1229, 1995.

[24] Iain D. Couzin, Jens Krause, Richard James, Graeme D. Ruxton, and Nigel R. Franks. Collective memory and spatial sorting in animal groups. *Journal of Theoretical Biology*, 218:1–11, 2002.

[25] Michele Ballerini et al. Interaction ruling animal collective behavior depends on topological rather than metric distance. *Proceedings of the National Academy of Sciences*, 105(4):1232–1237, 2008.

[26] C. Bradford Barber, David P. Dobkin, and Hannu Huhdanpaa. The quickhull algorithm for convex hulls. *ACM Transactions on Mathematical Software*, 22(4):469–483, 1996.

[27] Unity Technologies. Unity 6.2 [computer software]. `https://unity.com/`, 2025. Version: Unity 6.2. Accessed: 2026-03-01.

[28] Stéphane Bouchard, Geneviève Robillard, and Philippe Renaud. Revising the simulator sickness questionnaire for use in a virtual reality environment. *CyberPsychology & Behavior*, 10(4):549–555, 2007.

[29] Sandra G. Hart and Lowell E. Staveland. Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. In P. A. Hancock and N. Meshkati, editors, *Human Mental Workload*, pages 139–183. North-Holland, 1988.

[30] Daniël Lakens. Calculating and reporting effect sizes to facilitate cumulative science: A practical primer for t-tests and anovas. *Frontiers in Psychology*, 4:863, 2013.

[31] Sandra G. Hart. NASA-Task Load Index (NASA-TLX): 20 years later. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 50, pages 904–908, 2006.