BIMAS Proposal: Simulated Shepherding

Zach Atkins and Gowri Shankar Raju Kurapati

April 11, 2023

1 Problem Description

The herding problem concerns algorithmic methods for a small number of shepherding agents to control a substantially larger group of autonomous agents which behave according to a flocking algorithm. This problem arises naturally in the context of sheep-dog herding, where one dog is able to corral an entire herd of sheep, either trapping them within an area or moving them to a target. While this behavior is easy to train to dogs, modelling these interactions requires a analysis of a complex multi-agent system. The applications for such models, however, are quite broad. While some seek to simply use robots to perform the existing task of herding animals [11], such models could also be used to perform crowd control or prevent entry into dangerous or protected areas [3].

Generally, the herding problem consists of n autonomous, interacting agents (sheep), whose behavior is determined by a flocking algorithm, and a controlled shepherding agent (dog), which applies a repulsive force on the sheep in order to herd them toward a particular goal region. We initially adopt the model presented in [9], wherein the behavior of sheep is generally modelled using an artificial potential-field approach, which views the sheep as particles which exhibit attractive forces over long distances (cohesion), repulsive forces over short distances (collision-avoidance), and which, absent the presence of the dog, will graze and thus be mostly stationary with small random movement. See Appendix A for a rigorous description of the flocking model. Other potential models are discussed in Section 2 below.

The primary concern of the herding problem is defining the behavior of the dog, or in some cases, dogs. Section 2 discusses a variety of approaches in the literature and Section 3 addresses our intended approaches.

2 Background and Related Work

Modeling flocking behavior in multi-agent biological systems began with the three-heuristic boids model of [8], which was designed for computer graphics, and has since evolved to a wide variety of so-called *flocking algorithms*. In [1], Beaver and Malikopoulos distinguish between different flocking algorithms by whether they result in a line or cluster formation, as well as by their optimality in terms of energy consumption. Regardless of the particular heuristics used, flocking is generally modeled using an artificial potential field [5], where the individual agents are modeled as particles which exhibit attractive forces over long distances, repulsive forces over short distances, and which tend to align their velocities.

Most prior work modeling herding behavior has utilized potential field-based flocking algorithms to determine the behavior of the sheep [3, 2, 11] augmented with a strong repulsive force from the dog(s). In [11, 2], the sheep (or ducks in [11]) are presumed to have no intended target and wander according to their flocking behavior, while the dog(s) intend to herd the sheep toward a target area. Conversely, in [3], Grover and Mohanty et al. assume that the sheep agents have some self-motivated target area, the path to which may cause them to enter *protected zones*. In this scenario, the dogs simply aim to prevent the sheep from entering protected zones.

The behavior model of the dog varies widely between authors, with the seminal work by Vaughan et al. utilizing only the centroid of the flock [10]. The dog has the first two boid heuristics, i.e. it is attracted to the centroid proportional to their mutual distance so that it moves toward the flock and repulsed proportional to the inverse square of their mutual distance to prevent collisions, and has an additional repulsive force from the goal area with constant magnitude, which ensures that the orbit around the flock has a minimum value with the flock directly between the dog and the goal. They found this model to be robust to a variety of flocking parameters, and implemented it successfully using a robot and ducks. Fujioka and Hayashi build on this approach, with the dog favoring alternating positions of directly behind and to the left and right relative to the goal along the orbit around the flock, leading to a V-shaped trajectory of the flock.

Pierson and Schwager simplify the dynamics by assuming that some number of dogs are equidistant from the center of the herd, which transforms the mathematics of the system into that of nonholonomic vehicles, or vehicles which can only translate in the direction they are currently heading and which must stop completely to pivot [6]. Under these simplified assumptions, their model is able to herd a moderately sized group of sheep to a target zone *and* keep the herd there, which is a more sophisticated result than most other models. Additionally, the model scales to an arbitrary number of dogs.

In [3], the optimal velocities of the dogs are determined by solving a quadratic programming (QP) constrained optimization problem with the current positions of the sheep and the specified control regions as input. The control regions are formulated as *control barrier functions* $h(\cdot)$ that are non-negative whenever a given sheep is outside or on the boundary of the control region. This design allows for extension to multiple control regions and *containment* tasks by reversing the sign of h. Unfortunately, this model requires centralized computation of dog velocities and full knowledge of agent positions, which are unrealistic assumptions for real-world implementation.

Also, the Boid algorithm has been utilized as a powerful tool for analyzing the flocking behavior of birds and sheep, enabling researchers to gain insights into the collective dynamics of these animals and their interactions with their environment. Reinforcement learning approaches are discussed in [4], where the authors note the difficulty of traditional reinforcement learning due to the long exploration period times. Mahdavimoghadam et al. address these issues by allowing for information sharing between herding agents, permitting knowledge fusion to accelerate learning, and by introducing heuristic methods to prime herding behavior.

3 Project Plan

We will implement our own simulation environment for the herding problem, then use our simulation to test approaches for the dog behavioral model. Specifically, we will implement the unicycle model presented by Pierson and Schwager in [7] since it is the most sophisticated model and allows for multiple dog agents. In their paper, they did only cursory experiments with multiple sheep. We will extend their experiments in order to determine optimal numbers of dogs for different herd sizes.

Specifically, we will measure

1. The average distance of the sheep from the center of the herd (herd cohesion) over time,
2. The distance of the center of the herd from the goal,
3. Time needed to converge to the goal.

Using each of these metrics, we will determine optimal number of dogs for a given number of sheep. Then, we will compare the results from each metric, including time-series and time-averaged results for metric (1), and attempt to explain any discrepancies.

If time permits, we will implement and train an evolutionary AI model, which we anticipate will outperform the unicycle model while requiring fewer dog agents. Our model will use a reward function which incentivizes herd cohesion, the heading of the herd relative to the goal region, and the velocity of the herd.

3.1 Weekly Milestones

- Week 1-2: Implement simulation environment flocking algorithm as described in Appendix A in Python.
  - Stretch goal: Add interactive front-end which allows for manual movement of dog.
- Week 2-3: Implement the unicycle-model dog controller as in [7].
- Week 3-5: Run experiments as described above.
- Week 4-6: Compile results in presentation.
- Week 5-end: Write final paper.

References

[1] Logan E. Beaver and Andreas A. Malikopoulos. "An overview on optimal flocking". en. In: *Annual Reviews in Control* 51 (Jan. 2021), pp. 88–99. ISSN: 1367-5788. DOI: 10.1016/j.arcontrol.2021.03.004.

[2] Kaoru Fujioka and Sakiko Hayashi. "Effective shepherding behaviours using multi-agent systems". In: *2016 IEEE Region 10 Conference (TENCON)*. 2016 IEEE Region 10 Conference (TENCON). ISSN: 2159-3450. Nov. 2016, pp. 3179–3182. DOI: 10.1109/TENCON.2016.7848636.

[3] Jaskaran Grover et al. "Noncooperative Herding With Control Barrier Functions: Theory and Experiments". In: *2022 IEEE 61st Conference on Decision and Control (CDC)*. 2022 IEEE 61st Conference on Decision and Control (CDC). ISSN: 2576-2370. Dec. 2022, pp. 80–86. DOI: 10.1109/CDC51059.2022.9992986.

[4] Mahnoosh Mahdavimoghadam, Amin Nikanjam, and Monireh Abdoos. "Improved reinforcement learning in cooperative multi-agent environments using knowledge transfer". In: *The Journal of Supercomputing* 78.8 (2022), pp. 10455–10479.

[5] R. Olfati-Saber. "Flocking for multi-agent dynamic systems: algorithms and theory". In: *IEEE Transactions on Automatic Control* 51.3 (Mar. 2006), pp. 401–420. ISSN: 1558-2523. DOI: 10.1109/TAC.2005.864190.

[6] Alyssa Pierson and Mac Schwager. "Bio-inspired non-cooperative multi-robot herding". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. 2015 IEEE International Conference on Robotics and Automation (ICRA). ISSN: 1050-4729. May 2015, pp. 1843–1849. DOI: 10.1109/ICRA.2015.7139438.

[7] Alyssa Pierson and Mac Schwager. "Controlling Noncooperative Herds with Robotic Herders". In: *IEEE Transactions on Robotics* 34.2 (Apr. 2018), pp. 517–525. ISSN: 1941-0468. DOI: 10.1109/TRO.2017.2776308.

[8] Craig 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.

[9] Daniel Strömbom et al. "Solving the shepherding problem: heuristics for herding autonomous, interacting agents". In: *Journal of the royal society interface* 11.100 (2014), p. 20140719.

[10] Richard Vaughan et al. "Robot sheepdog project achieves automatic flock control". In: *Proc. Fifth International Conference on the Simulation of Adaptive Behaviour*. Vol. 489. 1998, p. 493.

[11] Richard Vaughan et al. "Experiments in automatic flock control". In: *Robotics and autonomous systems* 31.1-2 (2000), pp. 109–117.

A Flocking Model

Denote by $\boldsymbol{D}$ the position of the dog and by $\boldsymbol{S}_i$ the position of the ith sheep for $i = 1, \ldots, n$. Regardless of the dog, the sheep are repelled from any other sheep within a distance of less than r_s in the direction. Specifically, if a sheep i has k sheep within r_s from its position and those k sheep have positions $\tilde{\boldsymbol{S}}_1, \ldots, \tilde{\boldsymbol{S}}_k$, then sheep i experiences a repulsive force in the direction of

$$\boldsymbol{R}_i^s = \sum_{j=1}^{k} \frac{\boldsymbol{S}_i - \tilde{\boldsymbol{S}}_j}{\left\| \boldsymbol{S}_i - \tilde{\boldsymbol{S}}_j \right\|},$$

scaled by a weighting parameter ρ_s. If a sheep i is further than r_s from the dog, i.e. $\|\boldsymbol{S}_i - \boldsymbol{D}\| > r_d$, then it will graze and only exhibit small random motion and collision-avoidance forces. Otherwise, if $\|\boldsymbol{S}_i - \boldsymbol{D}\| \le r_d$, then the rest of the flocking model will be applied to the sheep. First, the dog applies a repulsive force in the direction of $\boldsymbol{R}_i^d = \boldsymbol{D} - \boldsymbol{S}_i$ scaled by a weight ρ_d. Second, the sheep i experiences an attractive force to the local center of mass of its k nearest neighbors $\boldsymbol{C}_i = \boldsymbol{LCM}_i - \boldsymbol{S}_i$ scaled by a weight c. Third, the sheep i experiences some inertia in its current normalized direction $\hat{\boldsymbol{H}}_i$ scaled by a weight h. Finally, the model includes an error term, which can be used to introduce randomness to the new direction, with normalized direction $\hat{\boldsymbol{\epsilon}}$ and scaled by a weight e. In total, the new direction $\boldsymbol{H}_i'$ of the sheep i is given by a linear combination of the normalized above forces as

$$\boldsymbol{H}_i' = h\hat{\boldsymbol{H}}_i + c\hat{\boldsymbol{C}} + \rho_s\hat{\boldsymbol{R}}_i^s + \rho_d\hat{\boldsymbol{R}}_i^d,$$

where $\hat{\boldsymbol{V}}$ is the normalized vector $\boldsymbol{V}/\|\boldsymbol{V}\|$. At each timestep, the new position $\boldsymbol{S}_i'$ of each sheep i is then computed by moving a distance of δ in the new direction, that is,

$$\boldsymbol{S}_i' = \boldsymbol{S}_i + \delta\hat{\boldsymbol{H}}_i'.$$