Evaluation of shepherding algorithms based on flock properties

Ori Gonen, Jan Flajžík, and Marko Muc

Collective behaviour course research seminar report

November 16, 2025

Iztok Lebar Bajec | associate professor | mentor

The purpose of this project is to study, implement, and evaluate existing shepherding algorithms based on various flock properties. In order to do so, we investigate how characteristics such as social heterogeneity of flock influence the convergence and overall effectiveness of herding algorithms. To achieve this, we implement several existing algorithms and evaluate their performance across different flock configurations.

Sheep herding | Flock behavior | Collective behavior

Introduction

Herding of sheep is a classic example of the unwilling movement of a group of individuals with a common goal. As such, it is a unique and fascinating subject for studying. The knowledge obtained might be useful in various fields from security to crowd control.

The motivation for this project arose from the observation of the geometric compactness of flock behavior during the herding process. Many examples can be found online[1]. Although the process itself appears almost magical, our primary goal is to determine how the herding can be carried out as efficiently as possible and to identify the factors that most strongly influence its effectiveness.

Related work

There are numerous studies examining the behavior of sheep flocks. Our primary reference is the paper by Jadhav et al. [1], in which the authors developed a model based on spatiotemporal data from a flock of 14 sheep. However, this approach has several limitations, including the small flock size and the lack of consideration for flock heterogeneity. As Bennett et al. demonstrated in [2], heterogeneity within social groups can negatively affect herding performance. In this paper, the authors developed multiple non-overlapping multiple social groups, with increasing the number of groups the performance decreased. The subgroups are mainly defined by two weighted forces: the force to interact with other sheep and the force to be repulsed from a dog.

There are also many existing studies that modeled the herding algorithms. Back in 2014 Daniel Strömbom et al. [3] presented an algorithm based on real world data that worked with attraction–repulsion herding. The algorithm is based on dynamical switching between two modes: collecting the sheep when they are too dispersed, and driving them to the goal when they are cohesive.

In a newer paper by Cai et al. [4] the authors present four herding algorithms that are also based on repulsion and attraction, but for reaching the goals, they introduce far-end and pausing mechanism that dynamically adjust the target point of a herd and pauses to optimize control and prevent circling respectively.

Methods

To achieve the goal of this project, we are going to implement the original algorithm from [1], the Herding Algorithm With Dynamic Far-end and Pausing Mechanism introduced in [4] and the Strömboms algorithm [3] for reference. Additionally, we divide the sheep into social subgroups based on the repulsion and attraction forces inspired by ideas described in [2].

To perform the experiments, we will tweak the parameters and run differently set up flocks on different algorithms. The main idea is to find the optimal parameters for each algorithm to converge as fast as possible.

We study, compare and compare existing sheep herding algorithms to find the optimal flock configuration which could increase the effectiveness of herding procedure.

[1] https://www.youtube.com/watch?v=tDQw21ntR64

For the implementation, we use Python programming language with standard math libraries such as *numpy* and *PyGame* for graphical interface. The simulation we are using simple box without any obstacles.

In this iteration, we started by implementing the algorithms. In this case, we only implement the herding algorithm from [1]. The authors of the original paper created their model in Matlab, and have published the model in their github repository[2].
In order to properly interact and experiment with the model, we re-implemented it in Python in our github repository[3], and created a simple graphical interface that can be seen in figure 1.

The algorithm defines movements for the individual sheep and herding dog based on the measured spatiotemporal data. The movement of sheep is based on the following factors:

1. repulsion from the other sheep,
2. attraction to the center of local neighbors,
3. alignment of the velocity direction to the average of neighbors,
4. repulsion from the dog,
5. random noise.

On the other hand, the dog has two modes of operation:

1. **Collecting:** the dog moves to collect stray sheep.
   This mode takes place when the group is not cohesive.
2. **Driving:** the dog moves behind the herd to drive it forward.
   This mode occurs when the group is cohesive.

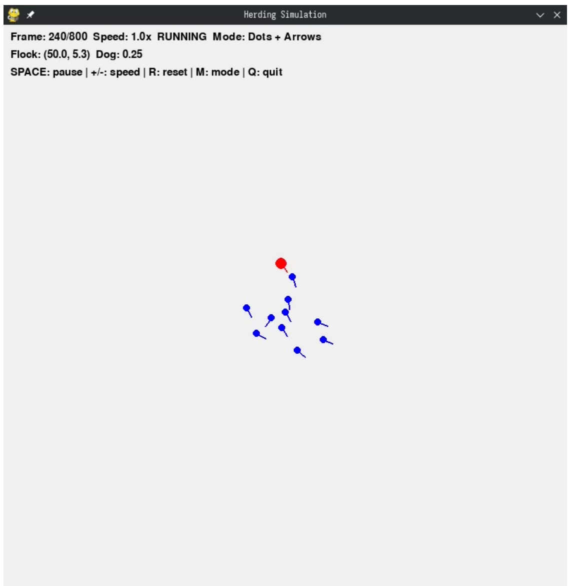

Figure 1. Graphical interface for displaying shepherding algorithms

Algorithm evaluation. To evaluate the existing herding algorithms, we will use the metrics of goal absement described as the distance between the goal location and the center of a sheep flock [2]. The goal is to minimize this score. The goal is also to measure the changing flock properties over time in different algorithm setting. Standard metrics such as elongation, cohesion, polarization, and others that are described in [1].

To evaluate multiple algorithm for any given flock configuration, we will use simple success rate metric that is defined as a ratio of all positive tests to all test.

[2] https://github.com/tee-lab/collective-responses-of-flocking-sheep-to-herding-dog
[3] https://github.com/OriGonen/CollectiveBehaviour_SheepHerd

Results

Since the project is at its beginning, we have not obtained any meaningful results except for re-implementing the original algorithm in Python and creating a basic visualization shown in Figure 1, but we expect several outcomes:

1. we expect that introducing the heterogeneous social groups will not only affect the convergence rate, but will also negatively affect the cohesion of the flock.
2. we expect that not every parameter set up will lead to convergence in reasonable time which requires us to set the maximal step count of an algorithm.
3. we expect the performance of all algorithms to vary significantly across different flock initial configurations, but for any given configuration, their performance should be roughly the same.

Discussion

The project is proceeding as planned. The only setback was the early departure of one team member, which resulted in minor communication issues that were quickly resolved. Also, the original project ideas that we described in our github readme file were reconsidered, therefore we decided not to implement terrain obstacles that were part of the original plan.

CONTRIBUTIONS. **JF** wrote the report, **OG** re-implemented the original algorithm and **MM** created the graphical interface.

Bibliography

1. Jadhav V, Pasqua R, Zanon C, et al. (2024) Collective responses of flocking sheep (ovis aries) to a herding dog (border collie). *Communications Biology* 7(1543).
2. Bennett C, Bullock S, Lawry J (2021) Demonstrating the differential impact of flock heterogeneity on multi-agent herding in *Towards Autonomous Robotic Systems*, eds. Fox C et al. (Springer International Publishing, Cham), pp. 147–157.
3. Strömbom D et al. (2014) Solving the shepherding problem: heuristics for herding autonomous, interacting agents. *Journal of the Royal Society Interface* 11:20140719.
4. Cai H, He Y, Wu J, Gao H (2023) Behavior-based herding algorithm for social force model based sheep herd. *Electronics* 12(2).