Optimizing Herding Efficiency: A Multi-Agent Simulation Study on Finding the Optimal Number of Shepherds for Herding Large Flocks

Zach Atkins
University of Colorado Boulder
zach.atkins@colorado.edu

Gowri Shankar Raju Kurapati
University of Colorado Boulder
gowri.kurapati@colorado.edu

Abstract

This paper presents a study on the herding problem, which involves controlling a large group of autonomous agents using a small number of shepherding agents. We develop a simulation environment to test approaches for the dog behavioral model using the unicycle model. Our goal is to investigate the optimal number of dogs for different herd sizes using metrics such as herd cohesion, distance of the center of the herd from the goal, and time needed to converge to the goal. The results of our study will help to determine the most effective approach for controlling herds and inform applications such as sheep-dog herding, crowd control, and protecting certain areas. We present a detailed description of our simulation environment, the experimental setup, and the results obtained from our study. Our findings indicate that the optimal number of dogs depends on the size of the herd and the desired level of herd cohesion. We also discuss the limitations of our study and suggest future research directions for the herding problem.

1 Introduction

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 [12], 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 [10], 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, Section 3 addresses our approaches and Section 4 deals with our results and its detailed analysis.

2 Related Work

Modeling flocking behavior in multi-agent biological systems began with the three-heuristic boids model of [9], 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 [6], 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, 12] augmented with a strong repulsive force from the dog(s). In [12, 2], the sheep (or ducks in [12]) 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 [13]. 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 a unicycle-like vehicle, or a vehicles which can only translate in the direction they are currently heading and must pivot around the center to change their heading [7]. 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 [5], 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 Methodology

3.1 Modeling of Sheep and Shepherds

We will describe the modeling of sheep and shepherds used in our simulation environment for the herding problem. We will begin by providing an overview of the artificial potential field approach used to model the behavior of sheep and then discuss the unicycle model used to represent the behavior of shepherds.

3.1.1 Sheep Modeling

In prior work, Pierson and Schwager [8] used Vaughan's model [13, 12] to model the behavior of sheep in the herding problem. However, Vaughan's model has limitations as it requires global sensing and assumes that sheep want to form one large cluster, without accounting for sheep behavior in the absence of a shepherd. To overcome these limitations, Strömbom's model [10] provides a more sophisticated approach to modeling sheep behavior by incorporating two separate modes of grazing and flocking, depending on whether a shepherd is within a radius r_s of a given sheep (see Appendix A for more details). Therefore, we adopt Strömbom's model to represent the behavior of sheep in our simulation environment for the herding problem. This model captures the behavior of sheep in the absence of a shepherd, making it a more realistic approach for our study.

3.1.2 Shepherd Modeling

After careful consideration of available models, we have determined that the unicycle model, as presented in Pierson and Schwager [8], represents the most sophisticated approach to modeling the dynamics of the shepherd (dog) agents in the herding problem. This model allows for the inclusion of multiple shepherding agents and has provable success in single-sheep scenarios, as well as empirical success with larger numbers of sheep. Therefore, we have chosen to implement this model in our simulation environment for the herding problem. For completeness, we summarize the model as presented in [8] below.

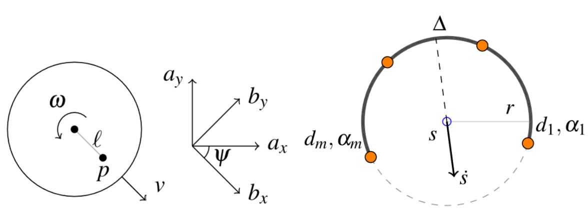
The unicycle model constrains the kinematics of the shepherds by reducing their motion to that of a unicycle-like vehicle using a local reference frame B relative to the global coordinate system A, with the transformation matrix defined as the rotation matrix from A to B with angle ψ, denoted in [8] as

$$^{A}R^{B} = \begin{bmatrix} \cos\psi & -\sin\psi \\ \sin\psi & \cos\psi \end{bmatrix}.$$

The coordinate system for the reference frame B is written as

$$\boldsymbol{b}_x = {}^{A}R^{B} \begin{bmatrix} 1 \\ 0 \end{bmatrix}, \quad \boldsymbol{b}_y = {}^{A}R^{B} \begin{bmatrix} 0 \\ 1 \end{bmatrix},$$

where the unicycle moves with speed $v = \|\dot{\boldsymbol{s}}\|$ in the $\boldsymbol{b}_x$ direction and has angular velocity $\omega = \dot{\psi}$. Figure 1a shows a schematic of the reference frames and kinematic reduction.



(a) (b)

Figure 1: Diagrams from Pierson and Schwager in [8] detailing the (a) reference frames and nonholonomic vehicle, and (b) the dog placement around the sheep center of mass $\boldsymbol{s}$.

The unicycle model reduces a herd of sheep to their global center of mass (GCM), or average position, denoted $\boldsymbol{s}$. Since the unicycle-like vehicle is nonholonomic, it is difficult to design an effective controller for it. However, Pierson and Schwager note that by defining a *point offset* some distance $\ell > 0$ from $\boldsymbol{s}$, denoted $\boldsymbol{p} = \boldsymbol{s} + \ell\boldsymbol{b}_x$, the dynamics of $\boldsymbol{p}$ are actually holonomic [8]. This fact allows the model to design a controller for $\boldsymbol{p}$, then map the dynamics back to the unicycle model. This mapping is given as $v = \boldsymbol{b}_x^T\dot{\boldsymbol{p}}$ and $\omega = \frac{1}{\ell}\boldsymbol{b}_y^T\dot{\boldsymbol{p}}$.

To conform to the unicycle model, further kinematic restraints are applied on the dog agents. Specifically, each of the $m \geq 2$ shepherding agents position themselves on a circle centered on the GCM of the herd $\boldsymbol{s}$, spaced along an arc of the circle spanning Δ radians. Figure 1b shows the placement of the dogs around the center of the herd. Each dog j has a target position in the global coordinate system $\boldsymbol{d}_j$ and a target angular position relative to ψ in the local reference frame α_j. The positions are given in terms of the angular positions as

$$\boldsymbol{d}_j = \boldsymbol{s} + r \begin{bmatrix} \cos\alpha_j \\ \sin\alpha_j \end{bmatrix},$$

where the angular positions with respect to ψ are written in terms of ψ and Δ as

$$\alpha_j = \psi + \pi + \Delta_j, \quad \Delta_j = \Delta\frac{2j - m - 1}{2m - 2}.$$

Since the unicycle model moves only with forward velocity, the dynamics can be written in terms of the For the shepherding problem without obstacles, as in the following, ψ is set to be the direction from the center of the herd to the goal. Determining the optimal value of Δ is substantially more difficult. Using the Vaughan model and substituting the positions of the dogs $\boldsymbol{d}_j$, the magnitude of the velocity of the sheep $\|\dot{\boldsymbol{s}}\|$, which is equal to the forward velocity v in the unicycle model, is given as

$$v = \|\dot{\boldsymbol{s}}\| = \frac{\sin\left(\frac{m\Delta}{2-2m}\right)}{r^2\sin\left(\frac{\Delta}{2-2m}\right)}.$$

Solving this equation for $\Delta \in [0, 2\pi]$ yields the arc angle for a given velocity. Then, by controlling the point offset with a proportional feedback controller $\dot{\boldsymbol{p}} = -k\boldsymbol{p}$, the desired forward velocity controller for the unicycle model is given by

$$v = -k\boldsymbol{b}_x \cdot \boldsymbol{p}.$$

Additionally, we determined the target radius r of the dogs using proportional feedback on a function of the current radius, the spread of the herd, and the repulsion radius r_s of the sheep model.

3.1.3 Our Model

One of the limitations of the shepherding dynamics presented in [8] is the use of the Vaughan model for modeling sheep, which has been shown to have certain drawbacks as detailed in Section 3.1.1. Moreover, the shepherd (dog) dynamics in [8] highly correlated with the fact that the Vaughan model is used for modeling sheep, which may not be practical in a real-world scenario. To overcome these limitations, we propose a novel shepherding dynamics that incorporates the most sophisticated algorithms, by modeling the sheep using Strömbom's model, as presented in [10], and using the unicycle model equations to describe the dog dynamics, as presented in [8].

4 Experiments & Results

4.1 Experimental Setup

We utilized the NetLogo platform [11] to simulate the multi-agent system of shepherding. To enable parallel processing, we integrated NetLogo with Python using the pyNetlogo module [4]. We set all the parameters of the Strömbom model to the values specified in [10] except for shepherd repulsion radius r_s increased to 75 from 65. To ensure the correctness of the Strömbom model used in our experiments, we first modeled the dynamics from [10] and replicated Figure 3 from the paper (see Figures 2a and 2b). This figure shows the proportion of successful shepherding events within 8000 time steps as a function of the number of sheep and nearest neighbors. We deemed a simulation successful if the shepherding agents were able to drive the sheep to the target location, which was fixed at $(-90, -90)$. The discrepancies between the figures are due to the fact that [10] used an unbounded domain, while NetLogo requires that the domain be either bounded or periodic; thus, it is possible for sheep to get stuck in the corners.

After implementing the unicycle model in NetLogo, we performed a parameter sweep on the number of sheep, number of shepherds, and number of nearest neighbors. For each combination of parameters, we ran the simulation with twelve different initial random seeds, which was used to place all agents' initial positions randomly within the center 300×300 unit region of the total 600×600 unit simulation region for a maximum of 6000 time steps. Figures 2c and 2d show the success rate heatmaps from this parameter sweep for our implementation of the unicycle model with different numbers of shepherds. For each simulation, we recorded whether the simulation was successful and the number of time steps required for completion.

Additionally, we ran a smaller parameter sweep, wherein we collected metrics at each timestep. These experiments are described in detail in Section 4.3.

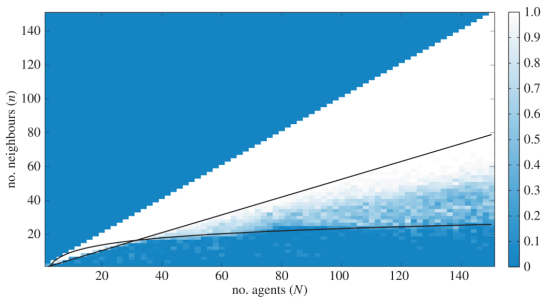
4.2 Analysis

In this section, we present an analysis of the shepherding simulations conducted in the previous section. Specifically, we investigate the optimal number of shepherds required to effectively herd a given number of sheep. To do so, we use the success rate and time taken as metrics to evaluate the performance of the shepherding simulations with varying numbers of shepherds and sheep. Furthermore, we will examine the changes in herd cohesion and sheep positions over time.

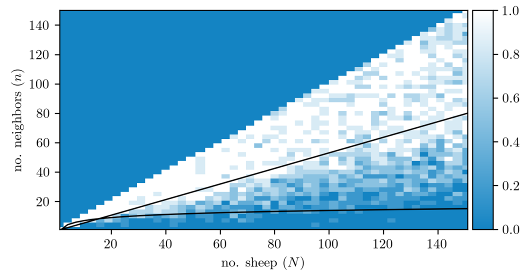
4.2.1 Success Rate

The success rate is defined as the average number of successful simulations for a given configuration of sheep and shepherds, where a successful simulation is one where the shepherds are able to successfully guide the sheep to the target location within the allotted time frame (see Figure 3). Our new model for shepherd dynamics is strengthened by the similarity observed in the pattern of the proportion of successful shepherding events plotted as a function of sheep for a specific number of shepherds, as compared to the results from Strömbom's paper (refer to Figures 2c and 2d).

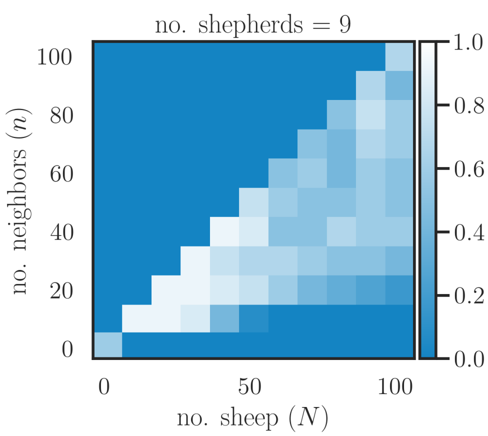
Table 1 represents the optimal number of shepherds needed to drive a particular number of sheep to the target location for the maximum possible success rate. The highest success rate of 100% is achieved for optimal configurations of sheep, nearest neighbors, and shepherds, indicating that the proposed shepherding model is highly effective in achieving the goal of gathering all the sheep to a central location. When there is only one sheep, two shepherds are optimal for achieving a 100% success rate which is unusual. But as the number of sheep increases, more shepherds are required to achieve a 100% success rate. As the number of sheep increases, the optimal number of nearest neighbors for



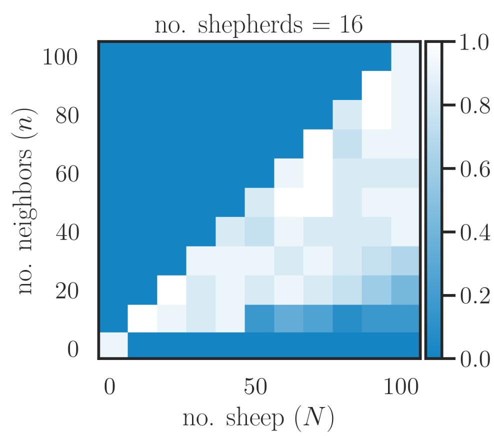
(a) Results from [10]



(b) Our implementation of Strömbom's model



(c) Our model, 9 Shepherds



(d) Our model, 16 Shepherds

Figure 2: Heatmaps for the proportion of successful shepherding events within 8000-time steps as a function of the number of sheep for (a) Original Figure from [10] (b) Our implementation of [10] (c) Our Model for 9 shepherds (d) Our Model for 16 shepherds

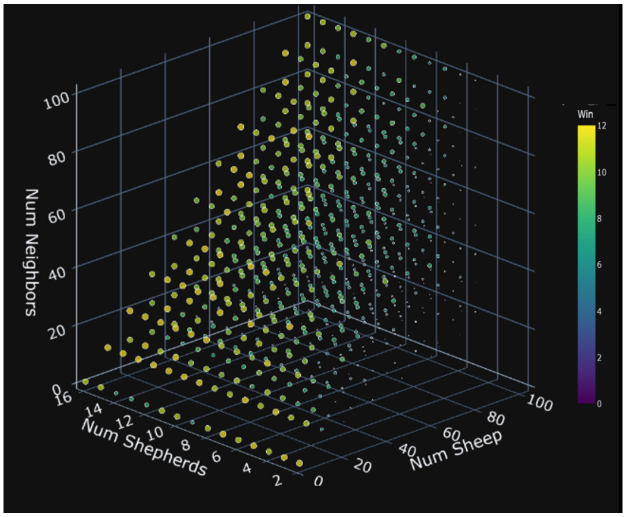


Figure 3: 3D scatter plot of the success rate of shepherding events based on the number of sheep, shepherds, and neighbors. Marker size indicates the proportion of wins, with larger markers representing higher success rates. The color bar represents the proportion of wins and is located parallel to the x-axis.

achieving the maximum success rate also increases, whereas the optimal number of shepherds does not show a consistent increase. Interestingly, there is no clear trend in the optimal number of shepherds as the number of nearest neighbors increases. For example, the optimal number of shepherds for 70 sheep with 70 nearest neighbors is 14, the same as for 60 and 80 nearest neighbors.

Sheep	Nearest Neighbors	Shepherds	Success Rate
1	1	2	100.00
11	10	5	100.00
21	20	7	100.00
31	30	7	100.00
41	40	10	100.00
51	50	14	100.00
61	60	14	100.00
71	70	14	100.00
81	70	13	100.00
91	80	16	100.00
101	90	13	100.00

Table 1: The optimal number of shepherds and nearest neighbors for a given number of sheep based on the maximum success rate possible.

For every value of sheep, the success rate increases as the number of neighbors increases. (see Figure 5).

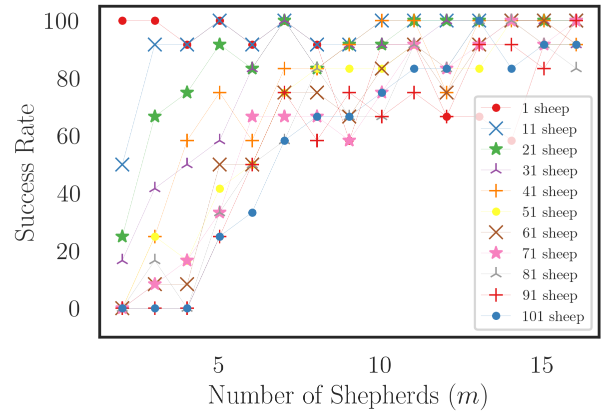


Figure 4: The figure shows the relationship between the success rate and the number of shepherds used, for a fixed number of sheep. The plot displays the maximum success rate achievable for the least number of neighbors possible, along with the corresponding number of shepherds used. Each line on the plot corresponds to a different value of sheep, which is identified using distinct markers. The plotted pairs of success rate and number of shepherds indicate the optimal settings for achieving the highest success rate with the least number of neighbors.

We can also observe that from Figure 4, the success rate increases as we increase the number of shepherds. Both of these inferences are expected and validate our new model for shepherding. When examining the data at a granular level, where the number of sheep is fixed and the variation of the number of shepherds vs success rate is plotted for available percentages of the number of neighbors, it can be observed that the success rate increases as the number of shepherds increases for a fixed number of nearest neighbors. However, it is also apparent that for a larger number of shepherds, the odd number of shepherds exhibit higher success rates compared to their corresponding even numbers. These observations are depicted in Figure 6 for the sheep count of 51 and 101.

4.2.2 Time Taken

In our experiments, the success of a simulation is determined by the time taken for the shepherds to guide the sheep to the target location within the allotted time frame. Table 1 shows the optimal number of shepherds required to achieve the minimum time taken for driving a particular number of sheep to the target location.

As the number of neighbors increases while keeping the number of sheep fixed, the time taken for shepherding decreases (See Figure 9).

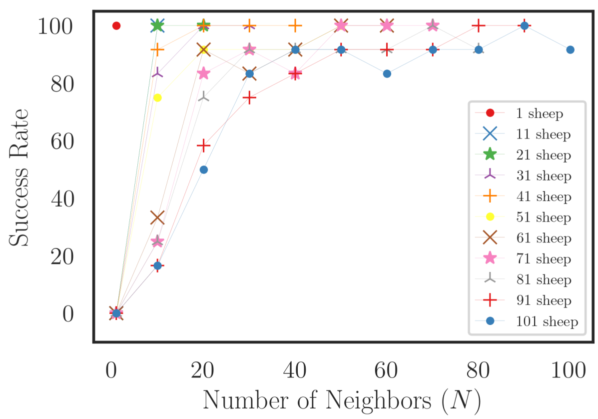


Figure 5: The figure shows the relationship between the success rate and the number of neighbors used, for a fixed number of sheep. The plot displays the maximum success rate achievable for the least number of shepherds possible, along with the corresponding number of neighbors used. Each line on the plot corresponds to a different value of sheep, which is identified using distinct markers. The plotted pairs of success rate and number of neighbors indicate the optimal settings for achieving the highest success rate with the least number of shepherds.

Furthermore, as the number of shepherds increases while keeping the number of sheep fixed, the time taken for shepherding also decreases, as demonstrated in Figure 8. Both of these observations align with our expectations and support the efficacy of our novel shepherding model in real-world scenarios.

Upon analyzing the data at a more detailed level, where the number of sheep is constant and the impact of the number of shepherds on time taken is plotted for various percentages of the number of neighbors, we found that time taken decreases with an increase in the number of shepherds for a fixed number of nearest neighbors. However, we also observed that odd numbers of shepherds lead to higher times than equivalent even numbers, particularly for larger numbers of shepherds. This phenomenon is illustrated in Figure 10.

As stated in 2, evaluating the optimal configuration based on the time taken is not recommended. This is due to the fact that, for sheep populations below sixty, achieving a high success rate requires a large number of shepherds despite the number of sheep being low. In most of these cases, the optimal number of shepherds being the maximum value that was used in the experiments is also concerning. On the other hand, for sheep populations above sixty, even though the number of shepherds and neighbors falls within the expected range, the success rates remain very low.

Sheep	Nearest Neighbors	Shepherds	Time Taken	Success Rate
1	1	9	294.57	58.33
11	10	14	403.67	100.00
21	20	16	419.75	100.00
31	30	16	379.90	91.667
41	40	16	412.30	83.34
51	50	16	412.90	83.34
61	10	5	461.00	8.33
71	10	5	460.00	8.33
81	80	5	427.00	8.33
91	20	5	439.00	8.33
101	30	5	428.00	8.33

Table 2: The optimal number of shepherds and nearest neighbors for a given number of sheep based on the minimum time taken.

4.3 Herd Cohesion

We measure the cohesion of the herd using its average and maximum spreads. The average spread is a measure of how dispersed the sheep are from the center of the herd, while the maximum spread is a measure of the maximum distance of any sheep from the center of the herd. Both these measures provide insight into the dynamics of the flock and the effectiveness of the shepherds in keeping the sheep together. Both the average spread and maximum spread exhibit similar trends as they measure the distance of sheep from the mean sheep position, albeit under different norms.

For simulations with a higher number of sheep, as time passes, we observe an overall decrease in both the average and maximum spread, but this decrease occurs usually in four distinct phases. Initially, there is a transitory or temporary phase followed by a very slow decrease in spread until a critical point is reached. Then, a sharp reduction in spread occurs, which is succeeded by a near-constant spread for a considerable period before the simulation concludes successfully (see Figures 7c and 7d).

This pattern can be explained by the Pierson dog model, which we used to model shepherd dynamics for our model. In these dynamics, dogs form a circle around the sheep and drive them collectively toward the goal. Initially, the dogs try to surround the sheep in a circle, causing a slow decrease in the spread. Once the dogs surround the sheep, they

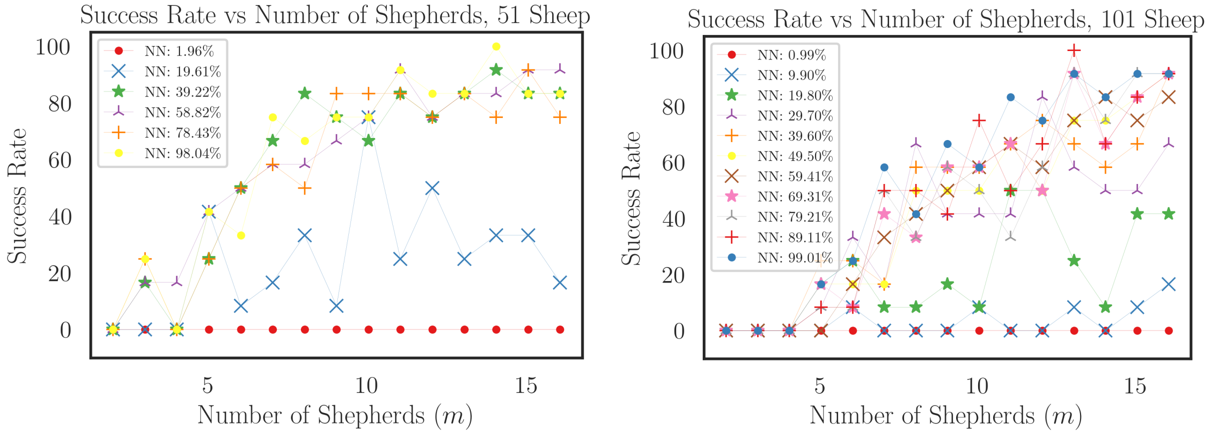
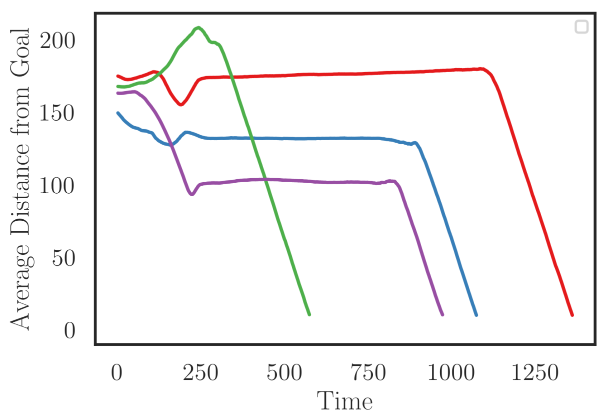
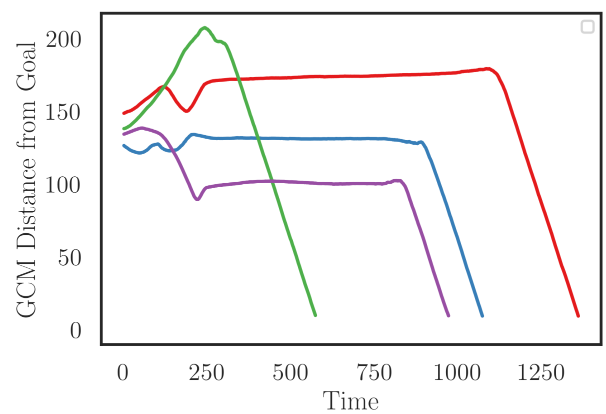


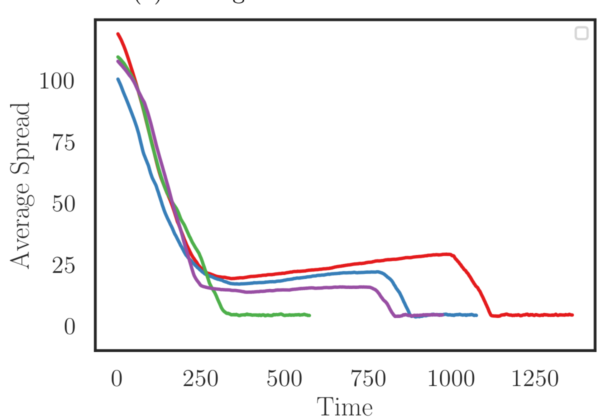
Figure 6: This plot illustrates the relationship between the success rate and the number of shepherds for different numbers of nearest neighbors (N), shown as a percentage of the number of sheep, for two numbers of sheep.



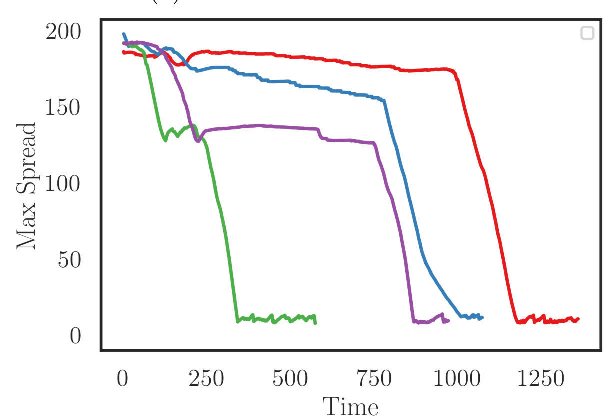
(a) Average Distance from Goal



(b) GCM Distance from Goal



(c) Average Spread



(d) Maximum Spread

Figure 7: This figure comprises four subplots, each displaying a different metric over time for the optimal configuration for 61 sheep to achieve maximum success rate. The colors in each subplot correspond to the results obtained from four separate simulations with different random initial positions of sheep and shepherds. The first subplot (a) shows the average distance from the goal, while the second subplot (b) displays the GCM distance from the goal. The third subplot (c) exhibits the average spread, and the fourth subplot (d) demonstrates the maximum spread, over time.

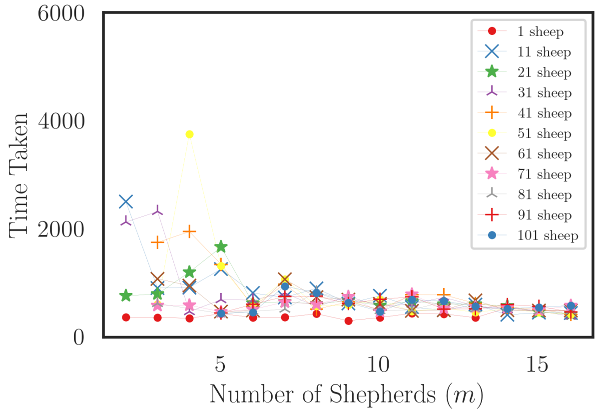


Figure 8: The plot illustrates the relationship between the time taken and the number of shepherds, for a fixed number of sheep. The plot displays the minimum time taken for the least number of neighbors possible, along with the corresponding number of shepherds used. Each line on the plot corresponds to a different value of sheep, which is identified using distinct markers. The plotted pairs of time taken and number of shepherds indicate the optimal settings for achieving the lowest time with the least number of neighbors.

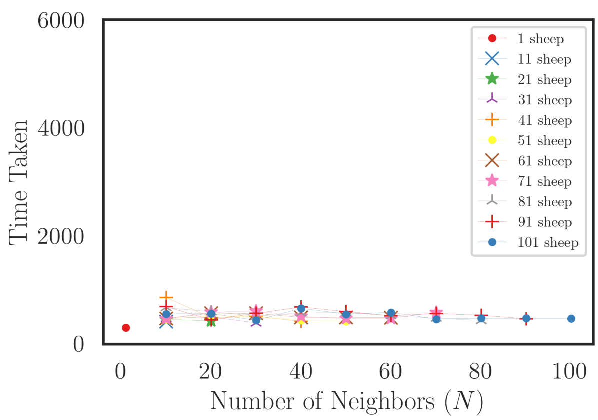


Figure 9: The plot illustrates the relationship between the time taken and the number of neighbors, for a fixed number of sheep. The plot displays the minimum time taken for the least number of shepherds possible, along with the corresponding number of neighbors used. Each line on the plot corresponds to a different value of sheep, which is identified using distinct markers. The plotted pairs of time taken and number of neighbors indicate the optimal settings for achieving the lowest time with the least number of shepherds.

close in on them radially and simultaneously push them toward the center of the circle, causing a steep decrease in spread. Then, once the sheep reach a minimal spread, the inter-sheep repulsion prevents any further compression, explaining the constant region as the sheep are driven to the goal.

For simulations with fewer sheep, the three phases occur simultaneously, making it challenging to visualize the distinct phases. As a result, we observe only a steady decrease in spread (see Figures 11c and 11d).

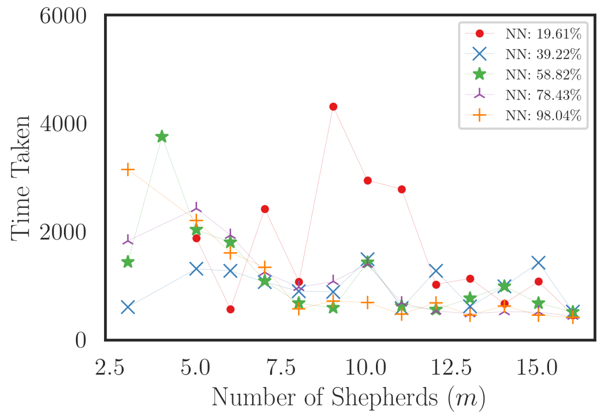
4.4 Global Center of Mass and Average distance

The average distance from the goal is a metric that calculates the average distance of all the sheep from the goal position. Similar to the average and maximum spread metrics, the GCM and average distance from the goal also display the same patterns.

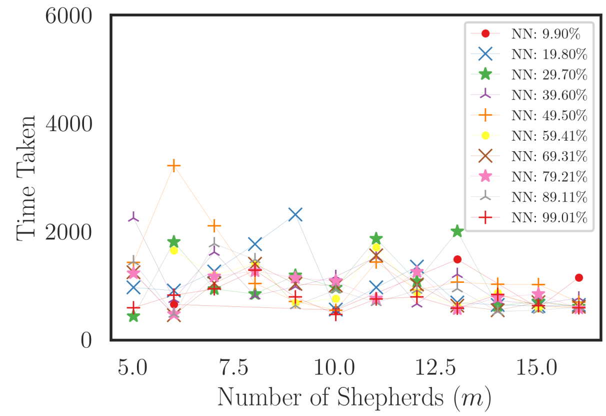
As the number of sheep in simulations increases, we observe a gradual decrease in both the GCM and average distance to the goal over time. This decrease happens in two distinct phases, which is two less than the four phases observed for sheep spread metrics. First, there is a transitory period marked by a slow decrease in distance as dogs take time to form a circle around the sheep. Second, a sharp decrease occurs until the distance to the goal reaches zero, as the dogs collectively drive the sheep (see Figures 7a and 7b). However, for simulations with a smaller number of sheep, the two phases occur concurrently, making them challenging to differentiate. Therefore, in such cases, only a steady decrease in distance to the goal is observed (see Figures 11a and 11b).

5 Conclusion

This research paper presents a novel approach to shepherding by proposing a model that combines sophisticated algorithms for modeling sheep using Strömbom's model and dog dynamics using the unicycle model equations. We have demonstrated the scalability of our model to handle large herds of sheep and large numbers of shepherds and investigated the relationships between the number of sheep, neighbors, and shepherds using success rate and time as metrics. Our findings suggest that success rate is a reliable metric for identifying the optimal configuration for a given number of sheep. We have also analyzed the impact of the number

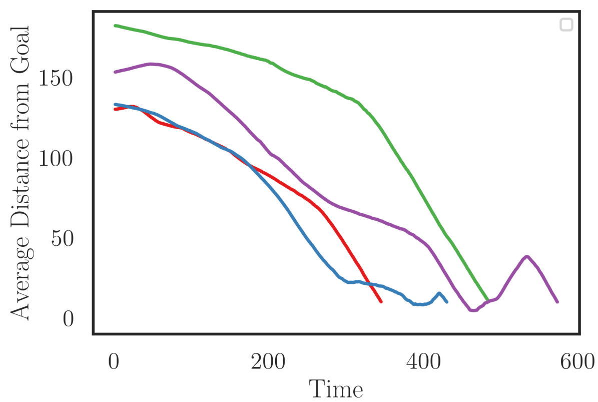


(a) $n = 51$ Sheep

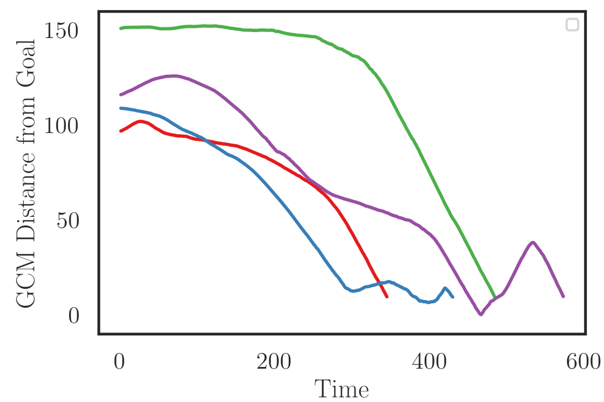


(b) $n = 101$ Sheep

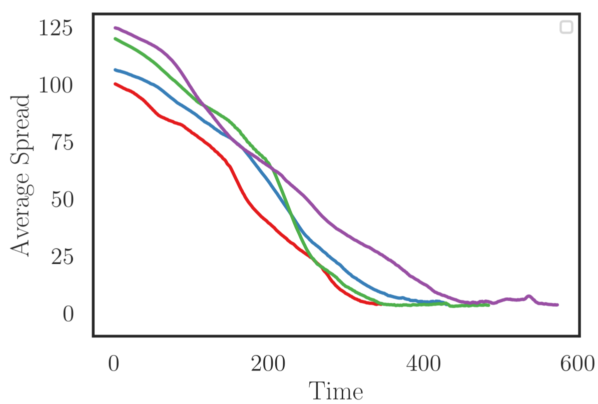
Figure 10: This plot illustrates the relationship between the time taken and the number of shepherds for different numbers of nearest neighbors (N), shown as a percentage of the number of sheep, for two numbers of sheep.



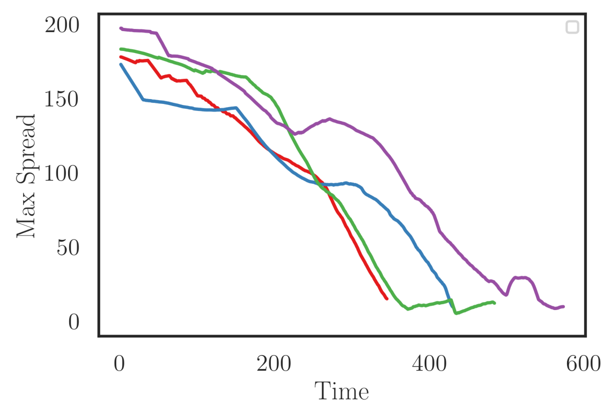
(a) Average Distance



(b) GCM Distance



(c) Average Spread



(d) Max Spread

Figure 11: This figure comprises four subplots, each displaying a different metric over time for the optimal configuration for 21 sheep to achieve maximum success rate. The colors in each subplot correspond to the results obtained from four separate simulations with different random initial positions of sheep and shepherds. The first subplot (a) shows the average distance from the goal, while the second subplot (b) displays the GCM distance from the goal. The third subplot (c) exhibits the average spread, and the fourth subplot (d) demonstrates the maximum spread, over time

of sheep on herd cohesion and positions over time. Although our model has shown success with the Strömbom and Pierson models, it still does not fully capture real-world sheep-shepherd dynamics. Future work should focus on incorporating more realistic models of shepherd dynamics. Overall, our study has contributed to the development of a more efficient and effective approach to shepherding.

Project Contribution

The project was a collaborative effort in which both Zach Atkins and Gowri Shankar Raju Kurapati made equal contributions towards the intellectual merit of the work. All decisions and ideas were discussed jointly, with active participation from both team members.

Zach wrote the final draft of the project proposal, led the creation of the NetLogo model for herding and shepherding dynamics, and was responsible for running the simulations and obtaining the necessary data for analysis. He oversaw the analysis process, made suggestions for improvement, and contributed to finalizing the project's conclusions. Additionally, Zach wrote the introduction and related work sections of the final paper and made final edits to the remaining sections.

Gowri Shankar Raju Kurapati conducted all data analysis, including creating necessary plots and drawing relevant inferences. He also wrote the methods, experimental results, and conclusion sections of the paper and made final edits to the remaining sections. Gowri contributed to the project proposal by determining the required metrics and analysis techniques, which informed Zach's simulation design.

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] Marc Jaxa-Rozen and Jan H Kwakkel. "Pynetlogo: Linking netlogo with python". In: *Journal of Artificial Societies and Social Simulation* 21.2 (2018).

[5] 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.

[6] 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.

[7] 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.

[8] 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.

[9] 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.

[10] 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.

[11] Seth Tisue and Uri Wilensky. "Netlogo: A simple environment for modeling complexity". In: *International conference on complex systems*. Vol. 21. Citeseer. 2004, pp. 16–21.

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

[13] 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.

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 $\|\boldsymbol{S}_i - \boldsymbol{D}\| \leq r_d$, 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 N 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,

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