To appear in *Advanced Robotics*
Vol. 00, No. 00, January 2013, 1–15

FULL PAPER

Analysis of Local-camera-based Shepherding Navigation

Y. Tsunoda[a*], Y. Sueoka[a], Y. Sato[a], and K. Osuka[a]

[a] *Dept. of Mechanical Engineering, Graduate School of Engineering, Osaka University, JAPAN*

(*v1.0 released January 2013*)

This study investigates group navigation with the aid of strong interaction between two kinds of agents: A shepherd drives a sheep group with a large population to a given goal position. Even though numerous studies have been performed on the realization of shepherd-like navigation, they are based on the condition that all sheep positions are given. This study examines the navigation of a sheep group using a local-camera-based approach, i.e., a shepherd perceives sheep using the shepherd's vision. Before testing local-camera-based navigation, we design a shepherd controller referred to as a farthest-agent targeting controller, in which the shepherd selects the sheep farthest from the goal. We demonstrate the validity of the proposed controller using statistical analysis and comparison with previous conventional controllers. After examining the effectiveness of this controller, we show that the controller works appropriately even if the shepherd cannot know all sheep positions. In addition, we show the robustness of the proposed controller for the positional errors of the sheep flock or for agent-lost cases to apply it to real-world situations.

Keywords: (Large group navigation, Local-camera based approach, Shepherd-like strategy)

1. Introduction

Multi-robot navigation, including the collective motions learned from living creatures such as fish schools [1–3] or bird flocks [4–6], has been widely studied to perform effective searching tasks or cooperative transport tasks [7, 8]. Such a remarkable group performance observed in nature has attracted not only biologists but also robotic researchers to examine swarm robotics systems [9–11]. Here, we present an insightful learning from living creatures. In a farm, only a small number of sheepdogs drive a sheep flock to a predefined target zone. Based on this, a robot sheepdog project has been demonstrated as an intelligent robot system, in which a small number of sheepdogs gather a flock of ducks and drive them to a given goal position [12].

In previous studies [12, 13], a navigation method was used to herd the center of a sheep group in numerical simulations, and this was applied successfully in the real world [14]. In addition, another shepherd controller, which was referred to as online target switching, was designed and numerically tested [13]. This controller switches its target depending on the states of a sheep flock [13] using two modes, i.e., the collecting mode and driving mode. Another related work focuses on the learning methods of a shepherd or the cooperation of multiple shepherds [15–17]. However, these works are based on the condition of the *global camera* –namely, i.e., the positions of all sheep are given.

This study examines shepherd-vision-based group navigation, in which a shepherd does not always know the positions of all sheep, i.e., the shepherd attempts to manipulate a sheep group based on the perceptive information obtained from the shepherd's vision. We refer to this ap-

*Corresponding author. Email: y.tsunoda@dsc.mech.eng.osaka-u.ac.jp

proach as *local-camera* navigation. Before investigating local-camera navigation, the authors propose a new kind of navigation method named farthest-agent targeting, in which the shepherd herds the sheep group from the side. This method enables the shepherd to realize stable navigation for various sheep characteristics.

First, we derive a sheep model based on previous studies and the Boid model [18, 19]. We show the effectiveness of the proposed controller by comparing the navigation performance for the following three shepherd controllers: center-of-group targeting, target-switching controller, and farthest-agent targeting. Then, we examine whether our controller works appropriately if the shepherd cannot know all sheep positions.

In addition, we demonstrate the robustness of the proposed controller by examining three situations, i.e., agent-lost cases, in which the target sheep is selected from the farthest sheep group, the case of including positional errors for the sheep, and the case of various initial position.

The rest of this paper is organized as follows: Section 2 describes the shepherd controllers and the modeling of a sheep group based on the Boid model. Section 3 compares three shepherd controllers mentioned above. Section 4 investigates the navigation of the sheep group using the local-camera approach, i.e., the shepherd determines its action based on vision. Section 5 examines the robustness of the proposed controller by examining two situations, i.e., selecting the target sheep from the farthest sheep group and including positional errors for the sheep, to apply the proposed controller to real-world situations. In addition, we examine navigation performance by changing related position between the flock and the shepherd. Section 6 summarizes the study and discusses the direction of future work.

2. Modeling of shepherding navigation

First, we create a sheep model, and then, we introduce two conventional controllers and the proposed controller.

2.1 *Design of sheep group*

We derive the model of a sheep group based on the Boid model proposed by Reynolds [18], which performs stable group formation using individual-based decision. The Boid model is the most typical model, and it demonstrates collective motions by considering three kinds of forces, i.e., repulsive force, cohesion force, and alignment force.

- Collision Avoidance: Avoid collisions with neighboring flock mates
- Velocity Matching: Attempt to match velocity with neighboring flock mates
- Flock Centering: Attempt to remain close to neighboring flock mates

Based on the model, we propose a sheep model as follows: First, we define a circular vision region, S_i (i is the identification number of a sheep), whose radius is R. The radius determines the interaction zone with others. An individual, i, responds by moving away from its neighbors, if neighbors are present in circular zone S_i. It follows the orientation of neighbors and is attracted by other individuals within the zone. Furthermore, every sheep attempts to avoid the shepherd as shown in Figure 1.

The moving vector, $v_i^s(k)$, of individual i at step k is integrated as follows:

$$v_i^s(k) = K_{s_1} v_i^1(k) + K_{s_2} v_i^2(k) + K_{s_3} v_i^3(k) + K_{s_4} v_i^4(k) \tag{1}$$

,where $K_{s_1}, K_{s_2}, K_{s_3}$, and K_{s_4} denote the gains of each vector. $v_i^1(k), v_i^2(k)$, and $v_i^3(k)$ represent repulsive, alignment, and attractive forces, respectively, and they are written as follows:

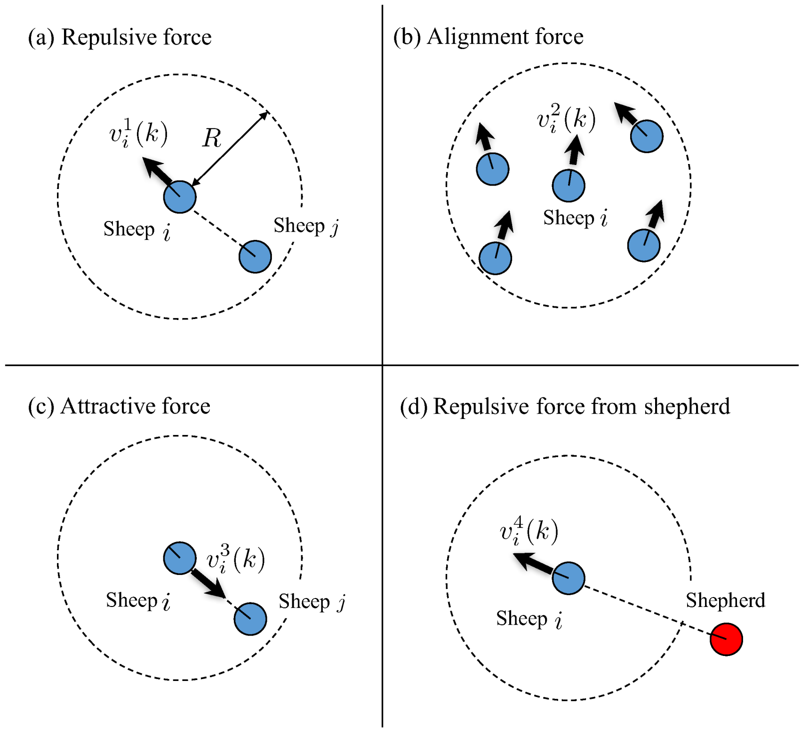


Figure 1. Schematic explanation of the sheep flock model. A sheep at the center is influenced by the neighboring agents within its vision area.

$$v_i^1(k) = -\frac{1}{N_{S_i}} \sum_{j \in S_i} \frac{x_j^s(k) - x_i^s(k)}{|x_j^s(k) - x_i^s(k)|^3} \tag{2}$$

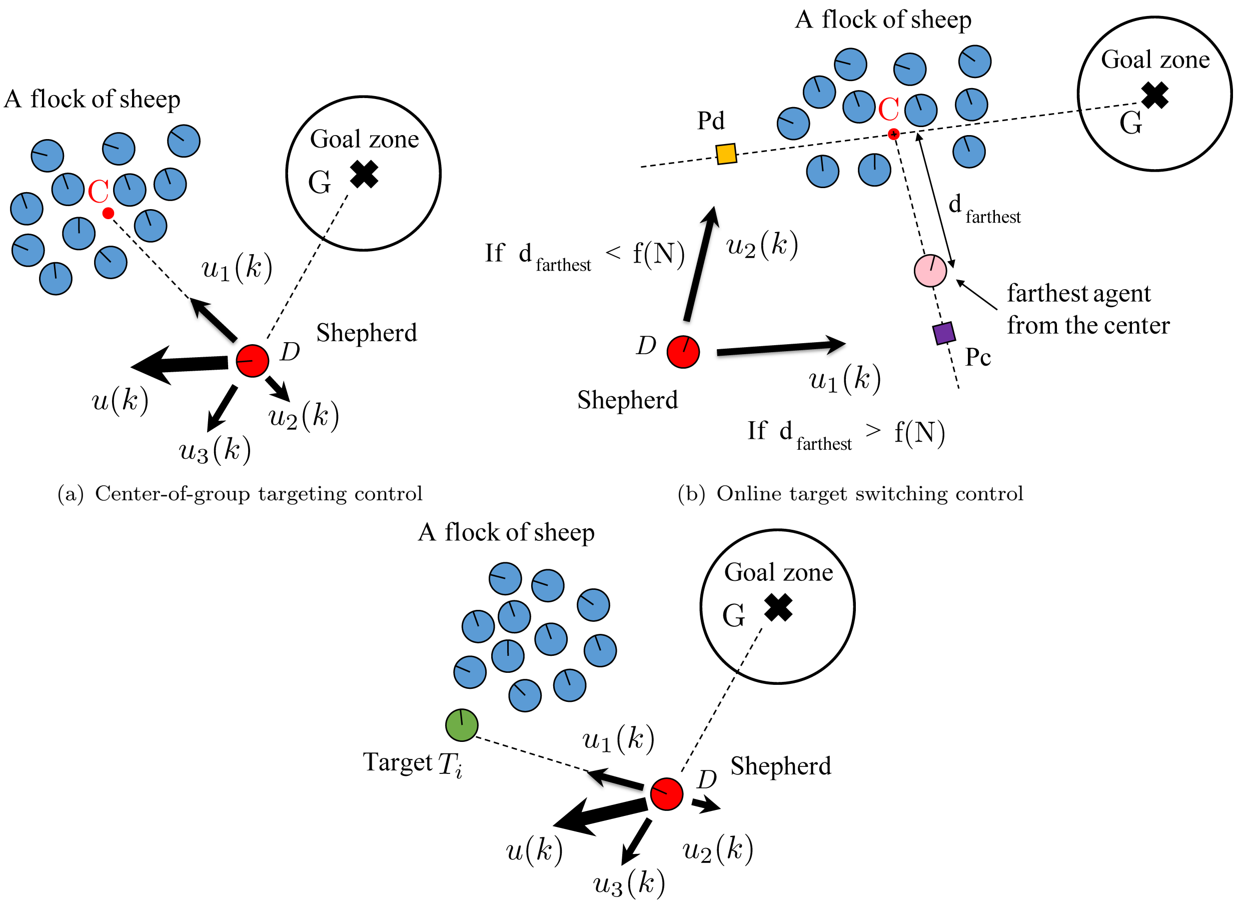
$$v_i^2(k) = \frac{1}{N_{S_i}} \sum_{j \in S_i} \frac{v_j^s(k-1)}{|v_j^s(k-1)|} \tag{3}$$

$$v_i^3(k) = \frac{1}{N_{S_i}} \sum_{j \in S_i} \frac{x_j^s(k) - x_i^s(k)}{|x_j^s(k) - x_i^s(k)|} \tag{4}$$

The repulsive force from the shepherd, $v_i^4(k)$, is given by

$$v_i^4(k) = -\frac{x_d(k) - x_i^s(k)}{|x_d(k) - x_i^s(k)|^3}, \tag{5}$$

where $x_i^s(k), x_j^s(k)$, and $x_d(k)$ denote the positions of individuals i and j and the shepherd, respectively, and S_i and N_{S_i} denote the neighbors of agent i within radius R and the number of neighbors, respectively. It should be noted that Boid model is not based on a real sheep flock in a precise sense because the sheep based on this model is always sticking by interaction with other sheep within the circular vision region. However, we would like to emphasize that we adapt this model for examining controller's performance designed in the shepherd by changing the group characteristics.



(a) Center-of-group targeting control (b) Online target switching control

(c) Farthest-agent targeting control

Figure 2. Shepherd controllers proposed in previous studies and this study

2.2 *Previous controller of the shepherd*

In this section, we discuss the modeling of the shepherd. Previous studies proposed an insightful shepherd controller –namely, *center-of-group targeting* [12], which was designed to approach the center of a sheep flock and tested in the real world. This paper proposes another controller, in which the shepherd attempts to pursue the sheep farthest from the goal zone. We refer to this controller as *farthest-agent targeting.*

2.2.1 Conventional controller: Center-of-group targeting

In center-of-group targeting, the shepherd attempts to approach the center of group agents (Figure 2(a)). The moving vector, $u(k)$, at step k is given by

$$u(k) = K_{d_1} u_1(k) + K_{d_2} u_2(k) + K_{d_3} u_3(k), \tag{6}$$

where K_{d_1}, K_{d_2}, and K_{d_3} denote the controller gains for $u_1(k) = \frac{\vec{DC}}{|\vec{DC}|}, u_2(k) = -\frac{\vec{DC}}{|\vec{DC}|^3}$, and $u_3(k) = -\frac{\vec{DG}}{|\vec{DG}|}$, respectively.

2.2.2 Online target switching

Next, we introduce another shepherd controller referred to as online target switching, which switches its target depending on the states of a sheep flock [13]. In this controller, the shepherd switches the target position according to the size of the flock for collecting and driving the flock (Figure 2(b)). The shepherd moves toward position Pc when the size of the flock (i.e., the maximum distance between each sheep and the center of the flock), $d_{farthest}$, is smaller than the threshold value of $f(N)$. The shepherd moves toward position Pd when $d_{farthest}$ is larger than the threshold value of $f(N)$. The shepherd stops when the shepherd is within c_0R from any agent (c_0 is compensation coefficient). The shepherd's moving vector, $u(k)$, at step k is given by

$$u(k) = \begin{cases} S_{d_1}u_1(k) & (d_{farthest} > f(N)) \\ S_{d_2}u_2(k) & (d_{farthest} < f(N)) \\ 0 & (L_{sD} < c_0R) \end{cases} \tag{7}$$

where S_{d_1} and S_{d_2} denote the controller gains for $u_1(k) = \frac{\vec{DPc}}{|\vec{DPc}|}$ and $u_2(k) = \frac{\vec{DPd}}{|\vec{Dd}|}$, respectively. L_{sD} is the length between the shepherd and any sheep. $f(N), Pc, Pd$ are given by $f(N) = c_1RN^{2/3}, Pc = c_2RN$, and $Pd = c_3R$, by referring to [13] (c_1, c_2, and c_3 are compensation coefficients).

2.3 *Proposed controller: Farthest-agent targeting*

The authors propose another controller referred to as *farthest-agent targeting*.

We consider that the shepherd can manipulate the flock efficiently by approaching the farthest agent from the center of the goal because the farthest agent is the most "problematic" agent in the flock when navigating. Only if the shepherd keeps approaching the farthest agent, sheep which are located far from the flock (goal zone) can be attracted to nearby sheep due to collective interaction based on Boid model. Because of that, the shepherd is presumed to gather every sheep gradually to the goal zone and finally enclose them into the circle zone.

In this method, the shepherd targets the sheep farthest from the goal position. For example, suppose that the farthest agent is T_i, as shown in Figure 2(c).

The moving vector, $u(k)$, of the shepherd is written as Equation (6), where $u_1(k) = \frac{\vec{DT_i}}{|\vec{DT_i}|}, u_2(k) = -\frac{\vec{DT_i}}{|\vec{DT_i}|^3}$, and $u_3(k) = -\frac{\vec{DG}}{|\vec{DG}|}$.

3. Analysis of global-camera-based navigation

We discuss global-camera-based navigation by comparing the results of center-of-group targeting, online target switching, and farthest-agent targeting.

3.1 *Evaluation index*

First, we propose evaluation indices to *quantitatively* examine navigation performance.

- Definition of navigation successes: every agent is present in the zone, whose origin is a given goal position and radius is r_g, within 500 steps, where the total number of agents is N.
- Navigation time is defined as spending step from the start to the finish.
- The dispersion of the flock is defined as the variance, σ, of group agents at each step given by

$$\sigma = \frac{1}{N}\sum_{k=1}^{N}|Cs_k|^2, \tag{8}$$

where N is the number of group agents, C denotes the center of the group, and s_k is the position of individual k.

We can simulate sheep navigation by setting the parameters of the sheep group and shepherd as shown in Table 1. Thirty sheep are placed in a circular area whose radius is 10 and center is $(-20, -20)$, a single shepherd is placed at $(15, 5)$, and a given goal zone is set to be a circle whose radius is 9.8 and center is $(20, 20)$.

Furthermore, we set the radius of the goal zone for the same density when all sheep are present in the goal zone.

3.2 *Navigation analysis*

We examine the performance of three controllers using a statistical approach.

3.2.1 Center-of-group targeting controller

We run 50 simulations by changing the values of repulsive force K_{s_1}, which corresponds to the dispersion of the sheep group.

Figure 3 shows the changes in sheep dispersion for different values of K_{s_1}.

The dispersion of sheep increases with K_{s_1}, and the sheep are more likely to split into groups. Table 2 and Table 3 show the change in the number of navigation successes and navigation time as K_{s_1} increases. It appears that navigation fails as K_{s_1} increases. Typical failure cases include splitting into groups, as shown in Figure 4. In addition, the results indicate that center-of-group targeting navigation works appropriately in the condition that the group of sheep is tightly connected.

3.2.2 Online target switching controller

Next, we describe group navigation based on online target switching control.

The values selected in the simulations are given as follows: $S_{d_1} = S_{d_2} = 3.0, c_0 = 0.001, f(N) = 1/2 \cdot 5 \cdot 30^{2/3} \approx 24.1, Pc = 3 \cdot 5 = 15, Pd = 1/5 \cdot 5 \cdot 30 = 30$. Table 2 and Table 3 show the simulation results for various values of K_{s_1}.

Table 1. Parameters in shepherd navigation

Symbol	Parameter	Values explored
R	Radius of view field	20
r	Size of each sheep	1
K_{s_1}	Coefficient of repulsive force among group agents	$10 - 100$
K_{s_2}	Coefficient of alignment force among group agents	0.5
K_{s_3}	Coefficient of attraction force among group agents	2.0
K_{s_4}	Coefficient of repulsion force from the shepherd	$5,000$
K_{d_1}	Coefficient of attraction force from sheep	10.0
K_{d_2}	Coefficient of repulsion force from sheep	200
K_{d_3}	Coefficient of repulsion force from goal	8.0

Table 2. Simulation results obtained from global-camera-based approach in each controller: Number of navigation successes in 50 trials

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Pre.1 : Center-of-group targeting	50	50	50	0	0	0	0	0	0	0
Pre.2 : Online target switching	0	0	0	50	50	50	0	0	0	0
Farthest-agent targeting	50	50	50	50	50	50	50	50	50	50

Table 3. Simulation results obtained from global-camera-based approach in each controller: Average navigation time in 50 trials

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Pre 1. Center-of-group targeting	66.74	102.08	155.64	-	-	-	-	-	-	-
Pre 2. Online target switching	-	-	-	216	136	119	-	-	-	-
Farthest-agent targeting	69.58	67.8	68.68	69.8	70.38	71.86	72.74	74.04	76.16	78.28

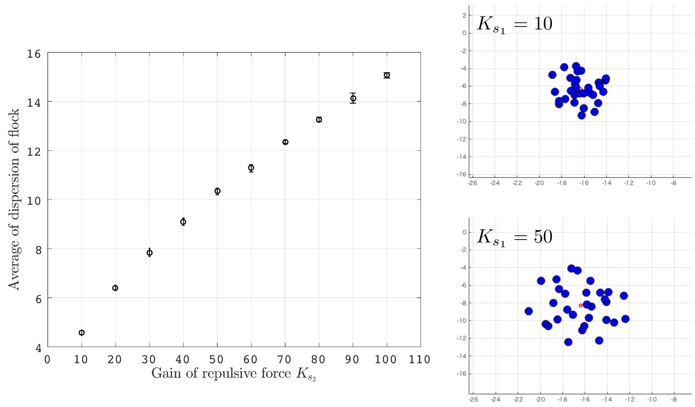


Figure 3. Changes in sheep dispersion for different values of K_{s_1}

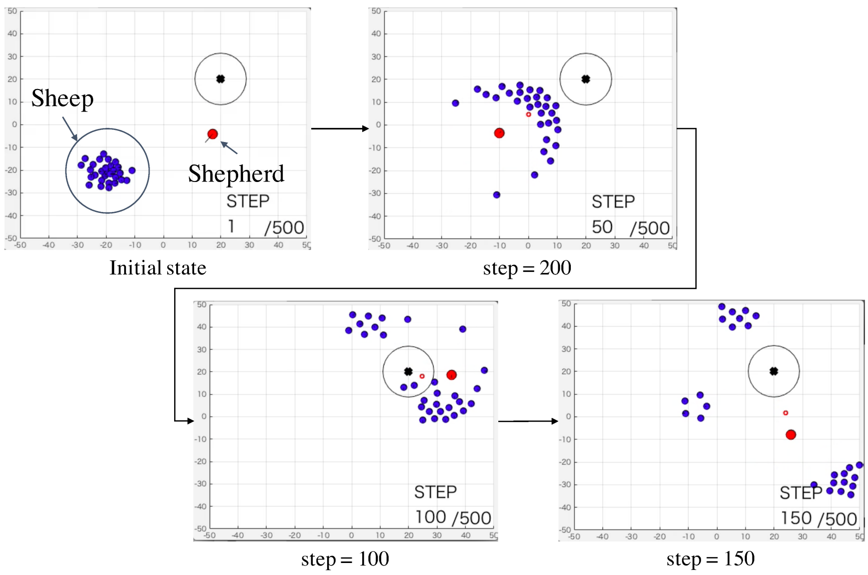


Figure 4. Shepherding failure: Sheep group splits into several groups

For $K_{s_1} = 10 \sim 30$, i.e., the size of flock is small, the shepherd tends to stop in front of the flock as Equation (7). So, the navigation fails because the shepherd can't come around behind the flock. Similarly, the navigaiton fails in the case of $K_{s_1} = 70 \sim 100$.

Figure 5 shows the snapshots of the navigation in the simulation. As shown in Figure 5, the shepherd pursues the agent farthest from the center of the flock to collect the sheep, even though the flock is navigated to the direction opposite to the goal. This is because this controller has two modes, i.e., collecting the flock and driving it to the goal. In the collecting mode, the shepherd attempts to gather the sheep to the center of flock, while in the driving mode, the shepherd attempts to navigate them to the goal. The shepherd was designed to adaptively switch its mode from driving to collecting. However, we can observe that the shepherd repeats the motions; after collecting the sheep in the collecting mode, the shepherd splits them into groups in the driving mode and then switches to the collecting mode again.

In [13], this controller is inspired by the real movement of a sheepdog when shepherding the real sheep, so this controller is not presumed to navigate the agents based on Boid model. It should be noted we conduct simulation analysis for this controller since we aim to compare this controller with the farthest-agent targeting controller.

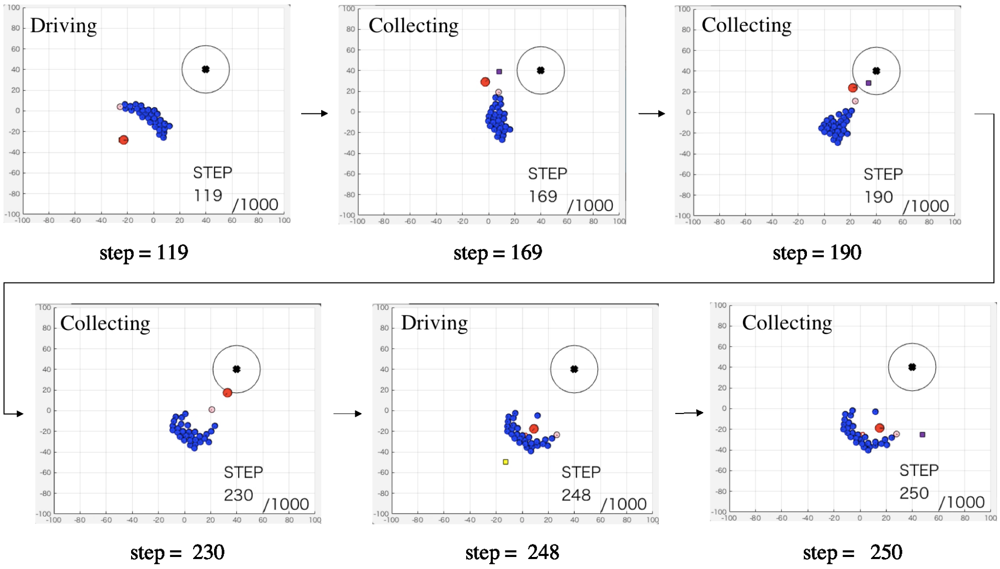


Figure 5. Example of shepherding based on the target-switching controller

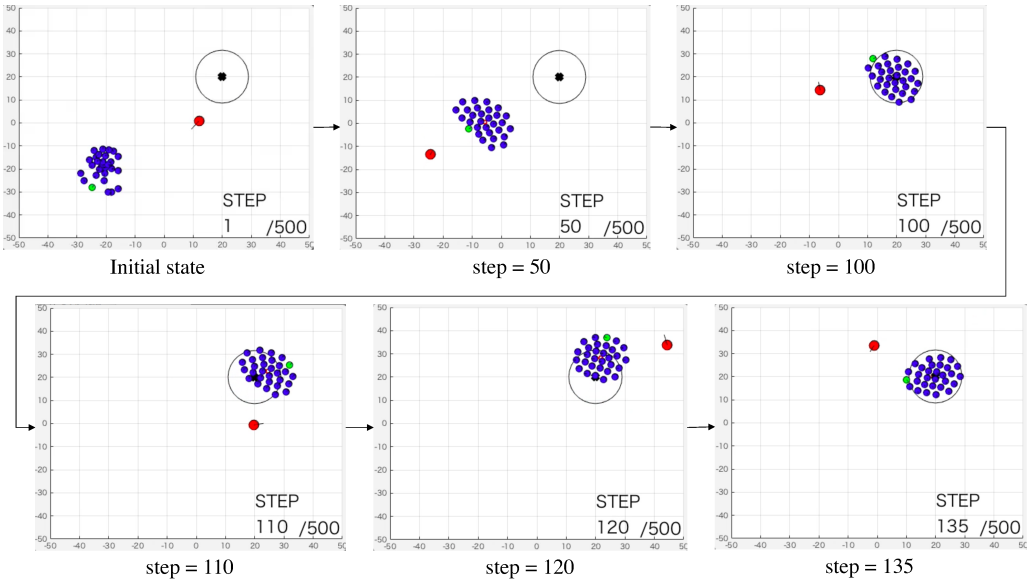


Figure 6. Snapshots of sheep and shepherd positions based on farthest-agent targeting control

3.2.3 Farthest-agent targeting

In this section, we discuss group navigation based on farthest-agent targeting control. We performed 50 simulations by changing K_{s_1}.

Figure 6 shows the snapshots of navigation based on farthest-targeting control. Table 2 and Table 3 show the simulation results for various values of K_{s_1}. Table 2 indicates that sheep navigation succeeds for larger values of K_{s_1}, even though navigation time slightly increases with K_{s_1} , as shown in Table 3. These results lead to the conclusion that the proposed controller works appropriately in the case of a loosely gathered group. The reason is that the shepherd does not attempt to approach the center of group but to *carefully* approach the group agents from the side.

First, we briefly discuss controller design. Throughout the simulations in which shepherd parameters K_{d_1} and K_{d_3} were changed, we found that navigation typically succeeded in the case of $K_{d_1} > K_{d_3}$. We simultaneously conducted the stability analysis for the guidance of the sheep flock, and found that in the case of a single sheep, the condition was $K_{d_1} > K_{d_3}$. Next,

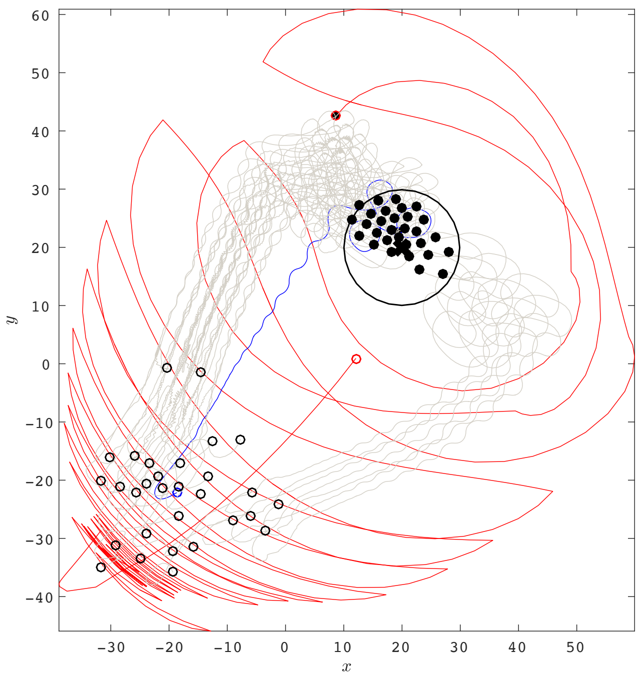
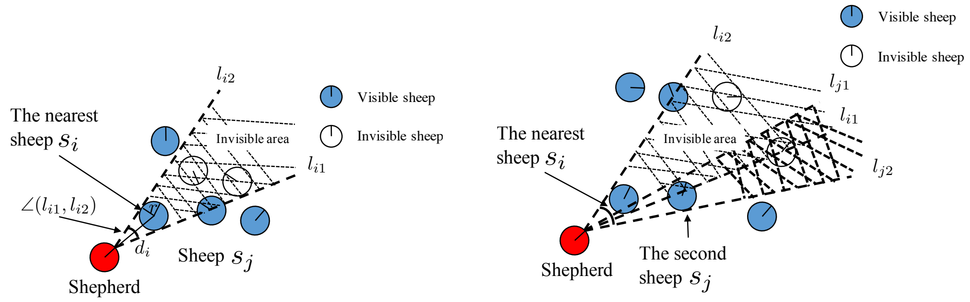


Figure 7. Trajectories of all agents: Red line is the shepherd's trajectory, blue line is the center of the flock's trajectory, and gray lines indicate each sheep's trajectory

we compare the proposed controller with the online target switching controller. The controllers appear to be similar; however, their performances are quite different. The farthest-agent targeting controller selects the target as the agent farthest from the goal. This provides the embedded function of collecting the sheep toward the goal position. For this reason, the guiding motion is automatically conducted with the collecting motion. Figure 7 shows one of the typical differences in the comparison of the proposed controller and online target switching controller. On the way to the goal zone, distance is maintained between the two groups into which the sheep are divided. This implies that the shepherd does not attempt to collect all sheep in one group, depending on the states of the sheep (e.g., the loosely gathered group). In addition, the two groups get closer as they approach the goal zone. In other words, the proposed controller simultaneously performs the two functions of collecting the sheep farthest from the goal and driving a group to the goal.

4. Analysis of local-camera-based navigation

This section examines the navigation performance of the local-camera-based approach by testing center-of-group targeting, farthest-agent targeting, and online target switching.



(a) Dead angle region generalized by the nearest sheep

(b) Integrated dead angle region created by the first and second nearest sheep

Figure 8. Definition of the invisible area

4.1 *Definition of local-camera view*

To investigate local-camera effects in a simple manner, we define shepherd vision as follows: Figure 8(a) and 8(b) show that the sheep that can be perceived from the local view of the shepherd. We draw two tangent lines, l_{i1} and l_{i2}, to the nearest sheep from the center of shepherd. The angle to sheep S_i between the lines is calculated as follows:

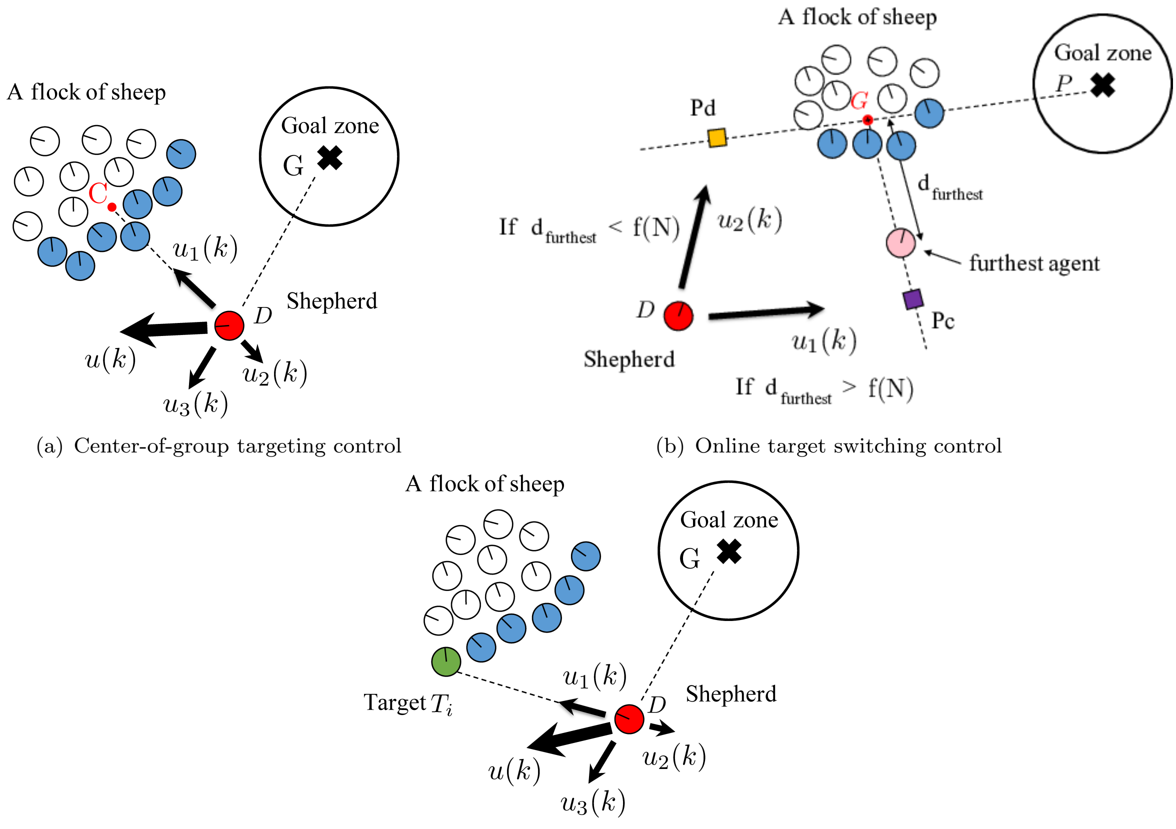
$$\angle(l_{i1}, l_{i2}) = 2\sin^{-1}\frac{r}{d_i} \tag{9}$$

As shown in Figure 8(a), there exists a dead angle region, which is created by the nearest sheep. In the same manner, for the second nearest sheep, s_j, we define the *invisible area* as the sum of the dead angle regions. We suppose that the shepherd perceives the sheep that are present in the area except for the invisible area and that the shepherd can perceive the position of a sheep if a part of it is in the visible area.

Here, we adjust the local-camera-based algorithms, i.e., center-of-group targeting control and farthest-agent targeting control. In center-of-group targeting, the shepherd attempts to approach the center, C, of group agents, as shown in Figure 9(a). The moving vector, $u(k)$, of the shepherd is determined by the following three forces: the approaching force to point C, the repulsive force from the point, and the repulsive force from the goal, where $u_1(k), u_2(k)$, and $u_3(k)$ are written as $u_1(k) = \frac{\vec{DC}}{|\vec{DC}|}, u_2(k) = -\frac{\vec{DC}}{|\vec{DC}|^3}$, and $u_3(k) = -\frac{\vec{DG}}{|\vec{DG}|}$. The vector, $u(k)$, at step k is integrated by the sum of the following three inputs as Equation (6).

In online target switching controller, the shepherd switches the target position according to the size of the flock for collecting and driving the flock (Figure 9(b)). The shepherd moves toward position Pc when the size of the flock (i.e., the maximum distance between sheep which the shepherd can perceive and the center of the visible sheep G), $d_{farthest}$, is smaller than the threshold value of $f(N)$. The shepherd moves toward position Pd when $d_{farthest}$ is larger than the threshold value of $f(N)$. The shepherd stops when the shepherd is within $c_0 R$ from the visible sheep (c_0 is compensation coefficient). The shepherd's moving vector, $u(k)$, at step k is given by Equation (7).

In farthest-agent targeting control, the shepherd attempts to approach sheep T_i selected from the visible sheep group, as shown in Figure 9(c). The moving vector, $u(k)$, of the shepherd is determined by the following three forces: the approaching force, the repulsive force from agent T_i, and the repulsive force from the goal, where $u_1(k), u_2(k)$, and $u_3(k)$ are written as $u_1(k) = \frac{\vec{DT_i}}{|\vec{DT_i}|}, u_2(k) = -\frac{\vec{DT_i}}{|\vec{DT_i}|^3}$, and $u_3(k) = -\frac{\vec{DG}}{|\vec{DG}|}$. The vector, $u(k)$, at step k is calculated by the sum of the following three inputs as Equation (6).



(a) Center-of-group targeting control

(b) Online target switching control

(c) Farthest-agent targeting control

Figure 9. Controllers in consideration of shepherd view

4.2 *Center-of-group targeting controller*

We examine local-camera-based shepherd navigation for the center-of-group targeting controller. We run 50 simulations by changing repulsive gain K_{s_1}. Table 4 and Table 5 show the simulation results.

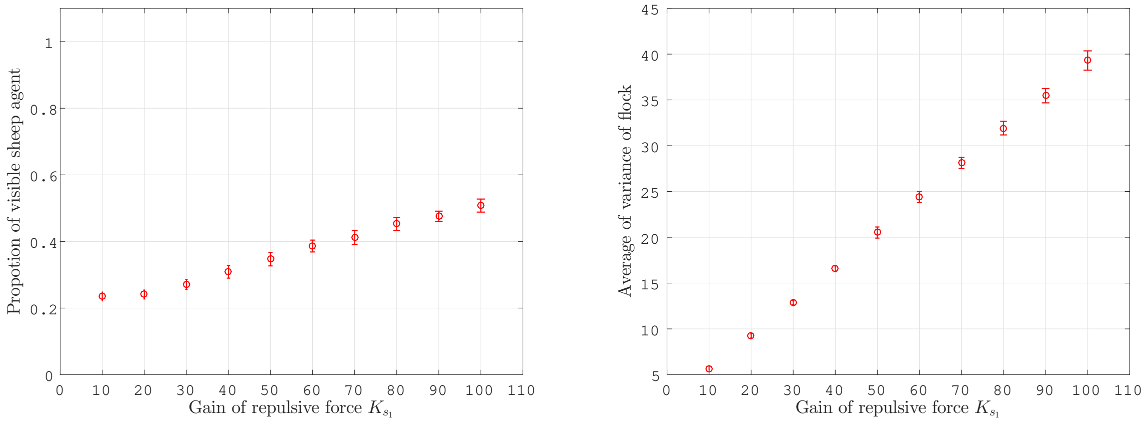
The center-of-group targeting method succeeds in navigation for small values of K_{s_1}. We found that navigation performance does not change, based on comparison with the results of the global-camera-based approach.

Table 4. Simulation results obtained from local-camera-based approach in each controller: Number of navigation successes in 50 trails

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Pre.1 : Center-targeting control	50	50	50	50	0	0	0	0	0	0
Pre.2 : Target-switching control	0	1	0	0	0	0	0	0	0	0
Farthest-agent control	50	50	50	50	50	50	50	50	50	50

Table 5. Simulation results obtained from local-camera based approach in each controller: Average time of navigation in 50 trials

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Pre 1. Center-targeting control	63.66	64.7	67.78	84.02	-	-	-	-	-	-
Pre 2. Target-switching control	-	490	-	-	-	-	-	-	-	-
Farthest-agent control	70.04	67.04	67.68	68.9	70.22	70.6	71.94	73.6	74.98	78.08



(a) Proportion of visible sheep

(b) Changes in sheep dispersion

Figure 10. Analysis of situations in local-camera cases for farthest-agent targeting control

4.3 *Online target switching controller*

We examine local-camera-based shepherd navigation for the online target switching controller. We run 50 simulations by changing repulsive gain K_{s_1}. Table 4 and Table 5 show the simulation results. This method fails in navigation in most cases. We found that the shepherd does not approach the agent which is located far from the flock because a position of center flock is not accurate due to the vision of the sheep.

4.4 *Farthest-agent targeting controller*

Table 4 and Table 5 show the navigation results for 50 simulations performed by changing the value of K_{s_1}. Figure 10(a) shows the proportion of visible sheep for all sheep, and Figure 10(b) shows the dispersion of the sheep group.

From Figure 10(a), we found that the proportion of the visible sheep increases with K_{s_1}. For example, in the case of $K_{s_1} = 100$, navigation succeeds even if the shepherd cannot perceive the positions of half of the sheep group. Furthermore, navigation performance is similar, based on a comparison with the results given in Table 3. Group size decreases with K_{s_1}. This leads to decrease in the proportion of sheep perceivable by the shepherd. For example, the shepherd can see 20% of the group in the case of $K_{s_1} = 10$ and 50% of the group in the case of $K_{s_1} = 100$. We can conclude that navigation always succeeds throughout the simulations, even if the shepherd can see less than 50% of the entire group.

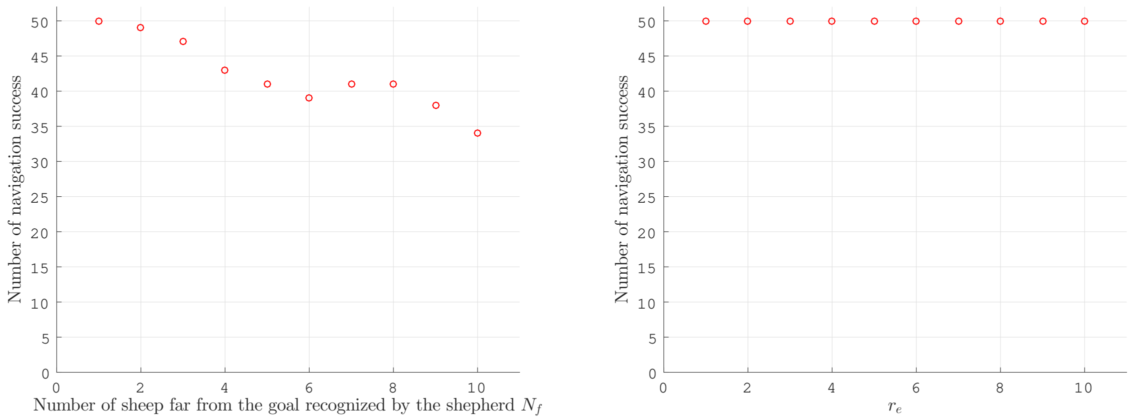
5. Influence of local camera accuracy on navigation

In this section, we discuss the robustness of the proposed controller. First, we examine the effect of selecting the target from the farthest agents instead of selecting the farthest agents for analyzing the agent-loss case. Second, we examine navigation performance by adding positional error to the sheep for considering camera accuracy in the real world.

At last, we analysis the effect of the initial positions of the shepherd and sheep for navigation performance.

5.1 *Design of far-side-agent targeting control*

We expand the proposed controller, named by far-side-agent targeting control, for selecting one target from the farthest agents. In each step, the shepherd selects its target randomly from N_f



(a) Number of successes in 50 trials based on far-side-agent targeting control

(b) Number of successes in 50 trials based on far-side-agent targeting control with position error

Figure 11. Local-camera-based simulation results for farthest-agent targeting control

sheep. The target is selected as the farthest sheep among the N_f sheep that are farther from the goal. We conducted simulations by changing the values of N_f. Figure 11 shows the proportion of navigation successes. We found that farthest-agent targeting control works appropriately and navigation becomes more difficult as N_f increases. We may say that the proposed method can be applied if the shepherd happens to perceive the position of the farthest sheep.

5.2 *Analysis of sensor accuracy*

Next, we examine navigation performance by adding position error for evaluating camera accuracy in the real world. The shepherd is supposed to know positions $(\hat{x}_i, \hat{y}_i)$, which are obtained

Table 6. Simulation results obtained from global-camera-based approach in farthest-targeting control: Number of navigation successes in 50 trials in regards to patterns of three initial positions

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Case 1 :	50	50	50	50	50	50	50	50	50	50
Case 2 :	50	50	50	50	50	50	50	50	50	50
Case 3 :	50	50	50	50	50	50	50	50	50	50

Table 7. Simulation results obtained from global-camera-based approach in farthest-targeting control: Average time of navigation in 50 trials in regards to patterns of three initial positions

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Case 1 :	74.1	70.64	68.46	69.42	70.38	72.94	73.62	75.52	78.18	79.78
Case 2 :	46.38	46.48	47.62	49.32	50.92	51	51.92	53.24	54.38	56.54
Case 3 :	66.02	63.26	62.84	62.78	63.2	63.52	63.98	65.26	66.62	68.2

Table 8. Simulation results obtained from local-camera-based approach in farthest-targeting control: Number of navigation successes in 50 trials in regards to patterns of three initial positions

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Case 1 :	50	50	50	50	50	50	50	50	50	50
Case 2 :	50	50	50	50	50	50	50	50	50	50
Case 3 :	50	50	50	50	50	50	50	50	50	50

Table 9. Simulation results obtained from local-camera-based approach in farthest-targeting control : Average time of navigation in 50 trials in regards to patterns of three initial positions

K_{s_1}	10	20	30	40	50	60	70	80	90	100
Case 1 :	71.58	67.8	67.9	68.54	69.88	71.12	71.68	73.42	74.94	78.24
Case 2 :	72.36	68.1	67.2	68.34	69.46	70.9	72.3	73.46	74.74	78.2
Case 3 :	67.04	63.18	62.84	63	62.98	63.12	63.3	64.58	65.98	67.56

by adding statistical error to actual positions (x_i, y_i), as follows:

$$\begin{pmatrix} \hat{x_i} \\ \hat{y_i} \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix} + r_e \begin{pmatrix} \cos\theta_{rand} \\ \sin\theta_{rand} \end{pmatrix} \tag{10}$$

where r_e denotes the maximum values added to the position, and θ_{rand} denotes random numbers from 0 to 2π with a normal distribution. We calculate the dead angle region using positions $(\hat{x_i}, \hat{y_i})$. Figure 11 shows the proportion of navigation successes. We can conclude that navigation succeeds if shepherd perception includes positional errors for sheep.

5.3 *Analysis of the effect of the initial position for navigation performance*

At last, we analysis navigation performance by changing initial positions between the flock and the shepherd. In this paper, we set 3 initial situations:

Case 1 Center of the goal zone is (20, 20), the shepherd is placed at (0, 0), and the sheep are placed in a circular area whose radius is 10 and center is (-20, -20).

Case 2 Center of the goal zone is (0, 0), the shepherd is placed at (20, 20), and the sheep are placed in a circular area whose center is (-20, -20).

Case 3 Center of the goal zone is (0, 0), the shepherd is placed at (-20, -20), and the sheep are placed in a circular area whose radius is 10 and center is (-20, -20).

We simulate navigations in these situations for various values of the gain of sheep's repulsive vector K_{s_1}. Table 6 and Table 7 show the simulation results in global-camera based approach. And Table 8 and Table 9 show the simulation results in local-camera based approach. These results indicate that the navigation based on the proposed method succeeds even though the shepherd is placed across the goal from the flock or is placed at inside of the flock.

6. Conclusions and future work

This paper proposes shepherd-like navigation using a localcamera-based approach. After deriving a sheep model, we propose a shepherd controller, in which a shepherd attempts to herd a sheep group from the side. The results of statistical analyses show that farthest-agent targeting control works appropriately if the sheep group is loosely gathered and that shepherd navigation succeeds even if the shepherd cannot perceive the positions of all sheep.

Moreover, we demonstrate navigation in the case of selecting the target sheep from the farthest group and in the case of including positional errors for sheep positions.

Future work will focus on the realization of local-camera-based shepherding navigation in a robotic system. Moreover, we will investigate navigating the group to different goals after splitting it into two groups.

Acknowledgments

This work was supported by JST CREST Grant Number JPMJCR14D5, Japan.

References

[1] Hemelrijk CK, Hildenbrandt H. Self-organized shape and frontal density of fish schools. Ethology. 2008;114(3):245–254.

[2] Katz Y, Tunstrøm K, Ioannou CC, Huepe C, Couzin ID. Inferring the structure and dynamics of interactions in schooling fish. Proceedings of the National Academy of Sciences. 2011;108(46):18720–18725.

[3] Romey WL. Individual differences make a difference in the trajectories of simulated schools of fish. Ecological Modelling. 1996;92(1):65–77.

[4] Hildenbrandt H, Carere C, Hemelrijk CK. Self-organized aerial displays of thousands of starlings: a model. Behavioral Ecology. 2010;21(6):1349–1359.

[5] Ballerini M, Cabibbo N, Candelier R, Cavagna A, Cisbani E, Giardina I, Orlandi A, Parisi G, Procaccini A, Viale M, et al.. Empirical investigation of starling flocks: a benchmark study in collective animal behaviour. Animal behaviour. 2008;76(1):201–215.

[6] Cavagna A, Cimarelli A, Giardina I, Parisi G, Santagati R, Stefanini F, Viale M. Scale-free correlations in starling flocks. Proceedings of the National Academy of Sciences. 2010;107(26):11865–11870.

[7] Kube CR, Zhang H. Collective robotics: From social insects to robots. Adaptive behavior. 1993; 2(2):189–218.

[8] Balch T, Hybinette M. Social potentials for scalable multi-robot formations. In: Robotics and automation, 2000. proceedings. icra'00. ieee international conference on. Vol. 1. IEEE. 2000. p. 73–80.

[9] Brambilla M, Ferrante E, Birattari M, Dorigo M. Swarm robotics: a review from the swarm engineering perspective. Swarm Intelligence. 2013;7(1):1–41.

[10] Şahin E. Swarm robotics: From sources of inspiration to domains of application. In: International workshop on swarm robotics. Springer. 2004. p. 10–20.

[11] Soysal O, Bahçeci E, Şahİn E. Aggregation in swarm robotic systems: Evolution and probabilistic control. Turkish Journal of Electrical Engineering & Computer Sciences. 2007;15(2):199–225.

[12] Vaughan R, Sumpter N, Frost A, Cameron S. Robot sheepdog project achieves automatic flock control. In: Proc. fifth international conference on the simulation of adaptive behaviour. 1998.

[13] Strömbom D, Mann RP, Wilson AM, Hailes S, Morton AJ, Sumpter DJ, King AJ. Solving the shepherding problem: heuristics for herding autonomous, interacting agents. Journal of the royal society interface. 2014;11(100):20140719.

[14] Vaughan R, Sumpter N, Henderson J, Frost A, Cameron S. Experiments in automatic flock control. Robotics and autonomous systems. 2000;31(1):109–117.

[15] Go CK, Lao B, Yoshimoto J, Ikeda K. A reinforcement learning approach to the shepherding task using sarsa. In: Neural networks (ijcnn), 2016 international joint conference on. IEEE. 2016. p. 3833–3836.

[16] Lien JM, Bayazit OB, Sowell RT, Rodriguez S, Amato NM. Shepherding behaviors. In: Robotics and automation, 2004. proceedings. icra'04. 2004 ieee international conference on. Vol. 4. IEEE. 2004. p. 4159–4164.

[17] Schultz A, Grefenstette JJ, Adams W. Roboshepherd: Learning a complex behavior. Robotics and Manufacturing: Recent Trends in Research and Applications. 1996;6:763–768.

[18] Reynolds CW. Flocks, herds, and schools: A distributed behavioral model. ACM SIGGRAPH Computer Graphics. 1987;21(4):25–34.

[19] Couzin ID, Krause J, James R, Ruxton GD, Franks NR. Collective memory and spatial sorting in animal groups. Journal of theoretical biology. 2002;218(1):1–11.