Proceedings of the 40th Chinese Control Conference
July 26-28, 2021, Shanghai, China

Sheepdog Driven Algorithm for Sheep Herd Transport

Yu Liu[1], Xin Li[1], Meiqi Lan[1], Yaqi He[1], He Cai[1], Huanli Gao[1]

1. School of Automation Science and Engineering, South China University of Technology, Guangzhou, 510641, China.
E-mail: hlgao@scut.edu.cn

Abstract: So far, most of the existing works on swarm systems are concerned with the collective motions. In this work, a somehow reverse problem is investigated, i.e., how an agent can interact with/control a swarm system. In particular, we consider the sheep herd transport problem: a herd of sheep is driven by a single sheepdog from the start place to the sheepfold. The rules governing the behaviors of the sheep are preset and known to the sheepdog, and the objective is to design the rule for the sheepdog to fulfil the transport task. Inspired by the behaviors of true sheepdogs, we have developed a backward semi-circle reciprocation algorithm featuring dynamic turn-around point selection. Numerical results are provided to evaluate the proposed algorithm.

Key Words: artificial intelligence, bio-inspired algorithm, sheep herd transport, swarm system

1 Introduction

Animal behavior inspired algorithm design is an interesting and important research direction for artificial intelligence [1]. Regarding swarm systems, so far, most of the existing works are concerned with the collective motions, such as schooling of fish and flocking of birds [2]. It has long been one of the key problems how various ordered motion patterns are formed, kept, and reorganized during flying, swimming and migrating for swarm systems, and what is the reason behind the quick adaption to environment stimuli. It is now widely recognized that these diverse ordered motion patterns are the results of simple local interactions among agents within the swarm system [3–6], and this phenomena is particularly called "emergence". This common recognition is largely due to the research on swarm system modeling. Specifically, agent based model construction and computer simulation help people reveal the internal mechanism of collective behaviors.

Generally speaking, the modeling for swarm system fall into two main categories: SAC based modeling and non-SAC based modeling. The SAC based modeling were proposed in the 1980s' for fish school and bird flock [7–10], where SAC refers to separation, alignment and cohesion. When all these three simple interaction rules are taken into consideration, collective behaviors emerge using computer simulation. For a long time, velocity averaging is at the core of most of the modeling methods, which indicates that the consensus in motion is the result of the velocity matching with neighboring agents. Typical works include the Couzin model [11–13], the Vicsek model [6, 14], and the social force model [15]. Nevertheless, velocity averaging was not validated by experiments, and from the practical point of view, there have been some effort put in non-velocity averaging based modeling, such as the asynchronous stochastic model [16] and the fuzzy logic model [17]. The most distinct feature for the asynchronous stochastic model is that each agent updates its state in an asynchronous way and chooses neighbors stochastically depending on relative distance. On the other side, Bajec introduced fuzzy logic into the SAC rules which made the obtained collective behaviors much closer to the true case. Besides SAC based modeling, there are also some remarkable results for non-SAC based modeling. For example, Buhl proposed a pursuit-evasion model inspired by the well known locust phase transition [18]. Moussaid proposed a crowd modeling based on line of sight which exhibited evident advantage for panic-stricken stampede modeling [19]. Some other works for non-SAC based model can be found in [20–22].

Most of the existing works are concerned with the collective motions of swarm systems. In this work, a somehow reverse problem is investigated, i.e., how an agent can interact with/control a swarm system with the priori knowledge of the local interaction rules for the agents within the swarm system. In particular, we consider the sheep herd transport problem: a herd of sheep is driven by a single sheepdog from the start place to the sheepfold. The rules governing the behaviors of the sheep are preset and known to the sheepdog, and the objective is to design the rule for the sheepdog to fulfil the transport task. Inspired by the behaviors of true sheepdogs, we have developed a backward semi-circle reciprocation algorithm featuring dynamic turn-around point selection. It is shown by numerical simulations that the proposed algorithm can duplicate the behaviors of the sheep herd and the sheepdog to a great extent.

The following notation are adopted in this paper. Let $\mathbb{R}$ and $\mathbb{N}$ denote the sets of real numbers and positive integers, respectively. Given $x \in \mathbb{N}$, $\underline{x} \triangleq \{y | y \in \mathbb{N}, y \leq x\}$. For a nonzero vector $x \in \mathbb{R}^2$, we let $\mathbf{o}(x) = x/||x||$ denote the unit vector which has the same orientation as x. Given a point $x \in \mathbb{R}^2$ and a region $\mathbb{X} \subseteq \mathbb{R}^2$, the distance between x and $\mathbb{X}$ is defined by

$$\mathbf{d}(x, \mathbb{X}) = \inf_{y \in \mathbb{X}} ||x - y||. \tag{1}$$

Given angle θ, the rotation matrix is defined as $\mathbf{R}(\theta) = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$. For a nonzero vector $x \in \mathbb{R}^2$, the

This work was supported in part by National Natural Science Foundation of China under Grant 61703167, and in part by the Science and Technology Planning Project of Guangdong Province under Grant 2019A050510015, 2021A1515012584, in part by the Guangdong Basic and Applied Basic Research Foundation under Grant 2020B1515120071, and in part by the Fundamental Research Funds for the Central Universities of SCUT under Grant 2020ZYGXZR059. Corresponding author: Huanli Gao.

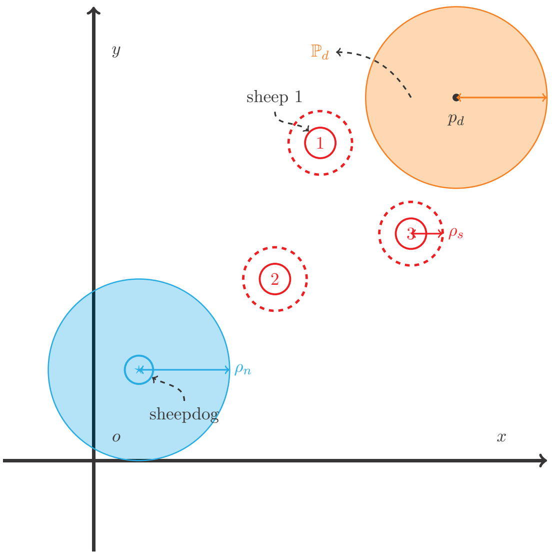


Fig. 1: An example of sheep herd transport with 3 sheep.

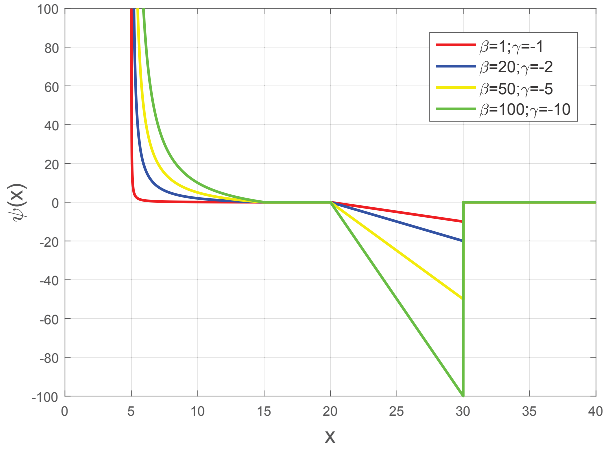


Fig. 2: Profiles of $\psi(x)$ with different gains.

left-hand side area of x is defined by

$$\mathbb{P}_l(x) = \left\{y \in \mathbb{R}^2 | \exists \theta \in (0, \pi),\ s.t.,\ \mathbf{o}(y) = \mathbf{R}(\theta)\mathbf{o}(x)\right\}, \tag{2}$$

and the right-hand side area of x is defined by

$$\mathbb{P}_r(x) = \left\{y \in \mathbb{R}^2 | \exists \theta \in (-\pi, 0),\ s.t.,\ \mathbf{o}(y) = \mathbf{R}(\theta)\mathbf{o}(x)\right\}. \tag{3}$$

Given a set $\mathbb{P}$ with finite elements, let $\mathbf{C}(\mathbb{P})$ denote the cardinality of the set $\mathbb{P}$.

2 Task Description

In this paper, as illustrated by Fig. 1, we consider the sheep herd transport problem driven by a single sheepdog. Throughout this paper, it is assumed that the positions of all the sheep and the sheepdog are expressed in a common coordinate system.

Suppose the sheep herd consists of N sheep. For $i \in \underline{N}$, the motion of the ith sheep is described by the following equation:

$$p_i(k+1) = p_i(k) + Tv_i(k) \tag{4}$$

where T denotes the sampling period, $p_i(k), v_i(k) \in \mathbb{R}^2$ denote the position and velocity vector of the ith sheep at the kth step, respectively. Let $\rho_s > 0$ denote the minimal inter-sheep safety distance.

The motion of the sheepdog is described by the following equation:

$$q(k+1) = q(k) + Tu(k) \tag{5}$$

where $q(k), u(k) \in \mathbb{R}^2$ denote the position and velocity of the sheepdog at the kth step, respectively. In what follows,for $i = 1, \ldots, N$, we let

$$p_i^q(k) = p_i(k) - q(k). \tag{6}$$

For $i \in \underline{N}$, the velocity of the ith sheep can be divided into two parts:

$$v_i(k) = v_{di}(k) + \mathbf{R}(\theta_i(k))v_{si}(k) \tag{7}$$

where

$$\theta_i(k) = a_i \cdot \frac{\pi}{180} \cdot \sin(\omega_i kT) \tag{8}$$

where a_i and ω_i are parameters; $v_{di}(k)$ and $v_{si}(k)$ represent the reaction to the dog and other sheep, respectively, whose specific forms are given by:

$$v_{di}(k) = \varphi(||p_i^q(k)||)\mathbf{o}(p_i^q(k)) \tag{9a}$$

$$v_{si}(k) = \sum_{j=1, j\neq i}^{N} \psi(||p_i(k) - p_j(k)||)\mathbf{o}(p_i(k) - p_j(k)) \tag{9b}$$

with

$$\varphi(x) = \begin{cases} \alpha\left(\frac{1}{x} - \frac{1}{\rho_n}\right), & 0 < x \leq \rho_n \\ 0, & x > \rho_n \end{cases} \tag{10a}$$

$$\psi(x) = \begin{cases} \beta\left(\frac{1}{x-\rho_s} - \frac{1}{\rho_r - \rho_s}\right), & \rho_s < x \leq \rho_r \\ 0, & \rho_r < x \leq \rho_g \\ \gamma\left(x - \rho_g\right), & \rho_g < x \leq \rho_d \\ 0, & x > \rho_d \end{cases} \tag{10b}$$

where $\alpha, \beta > 0$, $\gamma < 0$ are gains, $\rho_n, \rho_r, \rho_g, \rho_d > 0$ are distance parameters. The profiles of $\psi(x)$ with different gains are shown by Fig. 2 with $\rho_s = 5$, $\rho_r = 15$, $\rho_g = 20$, $\rho_d = 30$.

For $i, j \in \underline{N}$, $i \neq j$, if $\mathbf{o}(p_i^q(k)) = \mathbf{o}(p_j^q(k))$ and $||p_j^q(k)|| < ||p_i^q(k)||$, then the ith sheep is called *vision-blocked* by the jth sheep from the viewpoint of the sheepdog. Otherwise, it is called *not vision-blocked* by the jth sheep. Furthermore, if a sheep is not vision-blocked by any other sheep in the sheep herd and satisfies $||p_i^q(k)|| \leq \rho_v$, then it is called *visible* to the sheepdog. Otherwise, it is called *invisible* to the sheepdog. Here, ρ_v denotes the vision radius of the sheepdog. For convenience, the following index function is needed:

$$\Lambda_i(t) = \begin{cases} 1, & \text{the } i\text{th sheep is visible to the sheepdog;} \\ 0, & \text{the } i\text{th sheep is invisible to the sheepdog.} \end{cases} \tag{11}$$

The visible sheep set for the sheep dog can be defined as follows:

$$\mathbb{V}(k) = \{i | i \in \underline{N},\ \Lambda_i(k) = 1\}. \tag{12}$$

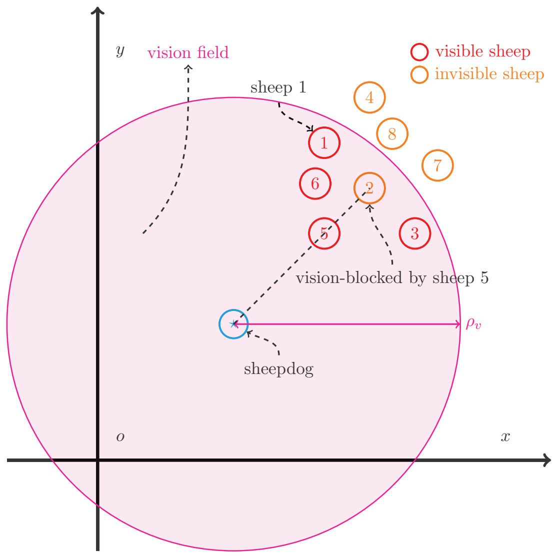


Fig. 3: An example of visible sheep set.

An example is shown by Fig. 3, where sheep 2 is vision-blocked by sheep 5 and $\mathbb{V}(k) = \{1, 3, 5, 6\}$. If $\mathbf{C}(\mathbb{V}(k)) > 0$, then the estimated center of the sheep herd is defined by

$$p_c(k) = \frac{\sum_{i \in \mathbb{V}(k)} p_i(k)}{\mathbf{C}(\mathbb{V}(k))}. \tag{13}$$

The sheep herd polygon is defined by

$$\begin{aligned} \mathbb{P}_s(k) = \Bigg\{ & p(k) | p(k) \in \mathbb{R}^2, \ p(k) = \sum_{i=1}^{N} \gamma_i p_i(k), \\ & 0 \leq \gamma_i \leq 1, \ \sum_{i=1}^{N} \gamma_i = 1. \Bigg\} \end{aligned} \tag{14}$$

Given $p_d \in \mathbb{R}^2$, the sheepfold is defined by

$$\mathbb{P}_d = \{p | p \in \mathbb{R}^2, \ ||p - p_d|| \leq \rho_o\} \tag{15}$$

for some $\rho_o > 0$. In what follows,for $i = 1, \ldots, N$, we let

$$p_i^d(k) = p_i(k) - p_d. \tag{16}$$

The task considered in this paper is to design $u(k)$ of the sheepdog, under the initial condition

$$\mathbf{d}(q(0), \mathbb{P}_s(0)) > 0, \tag{17}$$

such that for all $i \in \underline{N}$,

$$\lim_{k \to \infty} \mathbf{d}(p_i(k), \mathbb{P}_d) = 0. \tag{18}$$

3 Algorithm

In this paper, we design the driven rule of the sheepdog as a backward semi-circle reciprocation. The sheepdog runs behind the sheep herd. Since there are repulsive forces among sheep, the driven force from the sheepdog will propagate over the entire sheep herd. Therefore, the sheep herd will be controlled by the sheepdog and finally get the sheepfold

To begin with, we first introduce some important directions

$$\begin{aligned} D^{cd}(k) &= \mathbf{o}(p_d - p_c(k)) \\ D^{qd}(k) &= \mathbf{o}(p_d - q(k)) \end{aligned} \tag{19}$$

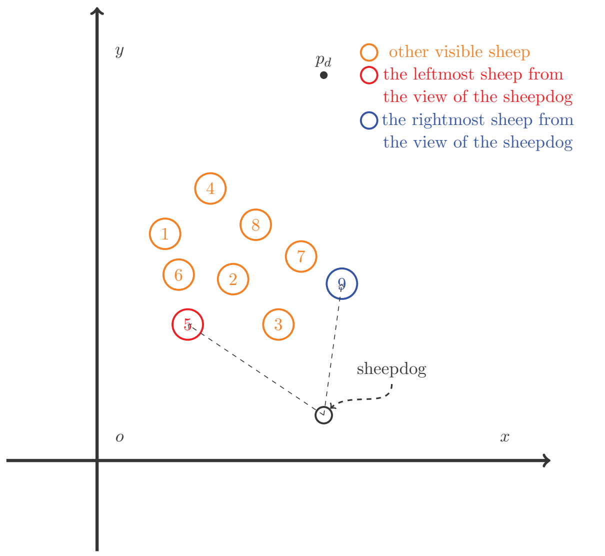


Fig. 4: An example of the rightmost and leftmost sheep from the view of the sheepdog.

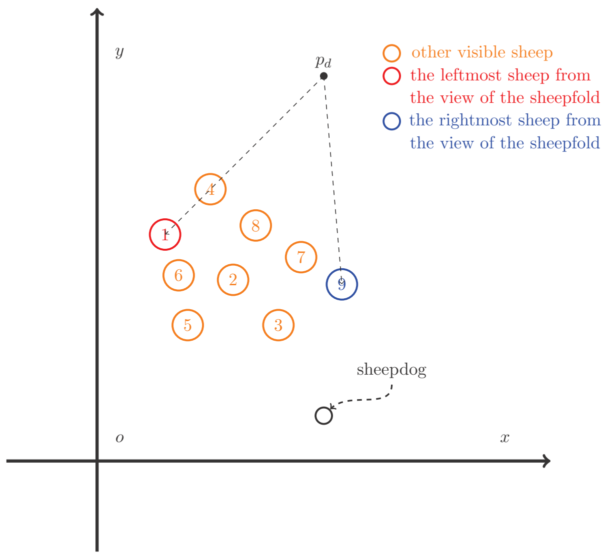


Fig. 5: An example of the rightmost and leftmost sheep from the view of the sheepfold.

Let

$$\begin{aligned} D_r(k) = \{ & p_i(k) | i \in \mathbb{V}(k), \forall j \in \mathbb{V}(k), j \neq i, \\ & p_j^q(k) \in \mathbb{P}_l(p_i^q(k)) \} \end{aligned} \tag{20}$$

and

$$D_l(k) = \{p_i(k) | i \in \mathbb{V}(k), \forall j \in \mathbb{V}(k), j \neq i, \\ p_j^q(k) \in \mathbb{P}_r(p_i^q(k))\}. \tag{21}$$

In (20) and (21), the ith sheep is called the rightmost and leftmost sheep from the view of the sheepdog, respectively. As shown in Fig. 4, all the other visible sheep are on the left-hand side of the vector from the sheepdog to the 9th sheep. Note that by the definition of visible sheep, there will be no visible sheep in the same direction. By the above definition, the 9th sheep is the rightmost sheep from the view of the sheepdog. Similarly, the 5th sheep is the leftmost sheep from the view of the sheepdog.

Let

$$C_r(k) = \{p_i(k) | i \in \mathbb{V}(k), \text{ s.t.}, \forall j \in \mathbb{V}(k), j \neq i, \text{ it follows} \\ p_j^d(k) \in \mathbb{P}_l(p_i^d(k)), ||p_i^d(k)|| > ||p_j^d(k)||\} \tag{22}$$

and

$$C_l(k) = \{p_i(k) | i \in \mathbb{V}(k), \text{ s.t.}, \forall j \in \mathbb{V}(k), j \neq i, \text{ it follows} \\ p_j^d(k) \in \mathbb{P}_r(p_i^d(k)), ||p_i^d(k)|| > ||p_j^d(k)||\}. \tag{23}$$

In (22) and (23), the ith sheep is called the rightmost and leftmost sheep from the view of the sheepfold, respectively.

As shown in Fig. 5, the 1st sheep and the 4th sheep are in the same direction from the view of the sheepfold, while all the other visible sheep are on the left side of the vector from the sheepfold to the 1st sheep. Since the 1st sheep is farther to the sheepfold than the 4th sheep, we call the 1st sheep the leftmost sheep from the view of the sheepfold. Similarly, we call the 9th sheep the rightmost sheep from the view of the sheepfold.

Furthermore, let

$$\mathbb{Q}_l(k) = \{x \in \mathbb{R}^2 | p_i(k) \in \mathbb{S}_r(p_d - x)\} \\ \mathbb{Q}_r(k) = \{x \in \mathbb{R}^2 | p_i(k) \in \mathbb{S}_l(p_d - x)\}. \tag{24}$$

By the above definition, if $q(k) \in \mathbb{Q}_l(k)$, then then all the sheep are on the right-hand (left-hand) side of $D^{qd}(k)$.

Define

$$R_c(k) = \frac{\langle D^{cd}(k), q(k) - C_r(k)\rangle}{||D^{cd}(k)|| \cdot ||q(k) - C_r(k)||} \tag{25}$$

and

$$L_c(k) = \frac{\langle D^{cd}(k), q(k) - C_l(k)\rangle}{||D^{cd}(k)|| \cdot ||q(k) - C_l(k)||}. \tag{26}$$

Moreover, let θ_t be the positive threshold for $R_c(k)$, and $L_c(k)$.

While herding, sheepdog will go straight or detour according to sheepherd. When the sheep go in the sheepfold, the sheepdog will stop. If the distance of the sheepdog and the sheep flock is far, the sheepdogs would go straight towards the rightmost or leftmost sheep relative to sheepdog. It will save the energy of the sheepdog. And in most cases, in order to get the sheep gathered and not to deviate from the route to the expected area, the sheepdogs would choose to detour, according to the sheep flock. When the sheepdog detours, the direction of the sheepdog is towards the rightmost or leftmost sheep relative to sheepdog with a rotation angle.

Let $\lambda(k)$ be a flag function indicating the current state of the sheepdog. The driven algorithm consists of two phases: initialization and iteration.

1) initialization: select design parameters $\theta_r, \theta_l, r_a, \theta_t, \gamma_a, \gamma_b$; let $\lambda(0) = 1$; set up upper time limit K.
2) iteration: run Algorithm 1 until k reaches K or the task objective (18) is achieved.

Algorithm 1 Sheepdog Driven Algorithm

Input: $p_1(t), \ldots, p_n(t), q(t), \lambda(k)$.
Output: $u(k)$.

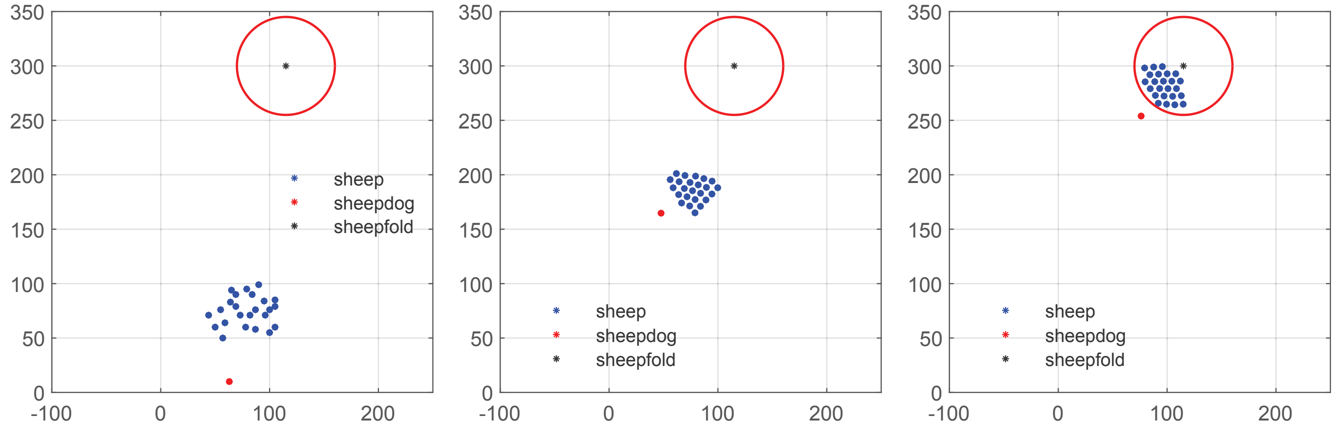
Set $\varpi = 0$.
for $(i = 1, i \leq \underline{N}, i = i + 1)$ **do**
  if $\mathbf{d}(p_i(t), \mathbb{P}_d) = 0$ **then**
    $\varpi = \varpi + 1$.
if $\varpi < \underline{N}$ **then**
  if $q(k) \in \mathbb{Q}_l(k)$ & $L_c(k) > \theta_t$ **then**
    $\lambda(k) = 0$,
    if $||q(k) - D_r(k)|| \geq r_a$ **then**
      $u(k) = \gamma_a \mathbf{o}(q(k) - D_r(k))$.
    else
      $u(k) = \gamma_b \mathbf{R}(\theta_r)\mathbf{o}(q(k) - D_r(k))$.
  else if $q(k) \in \mathbb{Q}_r(k)$ & $R_c(k) > \theta_t$ **then**
    $\lambda(k) = 1$,
    if $||q(k) - D_l(k)|| \geq r_a$ **then**
      $u(k) = \gamma_a \mathbf{o}(q(k) - D_l(k))$.
    else
      $u(k) = \gamma_b \mathbf{R}(\theta_l)\mathbf{o}(q(k) - D_l(k))$.
  else if $\lambda(k) = 1$ **then**
    if $||q(k) - D_l(k)|| \geq r_a$ **then**
      $u(k) = \gamma_a \mathbf{o}(q(k) - D_l(k))$.
    else
      $u(k) = \gamma_b \mathbf{R}(\theta_l)\mathbf{o}(q(k) - D_l(k))$.
  else
    if $||q(k) - D_r(k)|| \geq r_a$ **then**
      $u(k) = \gamma_a \mathbf{o}(q(k) - D_r(k))$.
    else
      $u(k) = \gamma_b \mathbf{R}(\theta_r)\mathbf{o}(q(k) - D_r(k))$.
else
  $u(k) = 0$.
return result

4 Simulation

In this section, we implement the algorithms on the two platforms, namely MATLAB and ROS, to check the performance of the proposed algorithm.

4.1 MATLAB Simulation

The algorithm parameters and systems initial positions are given as follows. $p_d = (115, 300)$, $\rho_o = 45$, $N = 24$, $q(0) = (63, 10)$, $p_1(0) = (57, 50)$, $p_2(0) = (44, 71)$, $p_3(0) = (59, 64)$, $p_4(0) = (73, 71)$,$p_5(0) = (78, 60)$, $p_6(0) = (82, 71)$, $p_7(0) = (87, 58)$, $p_8(0) = (96, 71)$, $p_9(0) = (55, 76)$, $p_{10}(0) = (64, 83)$, $p_{11}(0) = (69, 79)$, $p_{12}(0) = (50, 60)$, $p_{13}(0) = (87, 76)$, $p_{14}(0) = (95, 84)$,$p_{15}(0) = (100, 76)$, $p_{16}(0) = (105, 79)$, $p_{17}(0) = (65, 94)$, $p_{18}(0) = (69, 90)$, $p_{19}(0) = (105, 85)$, $p_{20}(0) = (79, 95)$, $p_{21}(0) = (84, 90)$, $p_{22}(0) = (90, 99)$, $p_{23}(0) = (100, 55)$, $p_{24}(0) = (105, 60)$, $a_i = 0.1$, $\omega_i = 0.1$, $\alpha = 7000$, $\beta = 1400$, $\gamma = -140$, $\rho_n = 50$, $\rho_s = 5$, $\rho_r = 15$,



(a) T=0.0001,k=0 (b) T=0.0001,k=25000 (c) T=0.0001,k=47000

Fig. 6: Three different stages that the sheepdog drive the sheep herd to the sheepfold using MATLAB.

$\rho_g = 20$, $\rho_d = 30$, $\theta_t = \frac{2\pi}{3}$, $\theta_l = -\frac{\pi}{4}$, $\theta_r = \frac{\pi}{4}$, $r_a = 40$, $\gamma_a = 450$, $\gamma_b = 375$.

The simulation results are shown by Fig. 6, and it can be seen that the sheep herd have been driven to the sheepfold by the sheepdog successfully. Fig. 6-a shows the initial positions of the sheep herd and the sheepdog, where the sheep are within some certain area, and the sheepdog is behind sheep herd. Fig. 6-b shows that sheep herd have been gathered driven by the sheepdog. Fig. 6-c shows that the sheep herd get the sheepfold and the sheepdog stops.

4.2 ROS Simulation

We implement the algorithms on gazebo. Note that on this platform, the sheep and sheepdog are viewed as mobile agents whose motion equations are described by the following unicycle modelling:

$$\begin{aligned} p(k+1) &= p(k) + Tv(k)\begin{pmatrix} \cos\theta(k) \\ \sin\theta(k) \end{pmatrix} \\ \theta(k+1) &= \theta(k) + T\omega(k) \end{aligned} \tag{27}$$

where T denotes the sampling period, $p(k), \theta(k)$ denote the position and the heading direction of the mobile agent at the kth step, respectively, $v(k), \omega(k)$ denote the linear and angular velocity of the mobile agent at the kth step, respectively.

Obviously, system (27) is nonlinear. To make the proposed algorithm in this paper applicable to system (27), we perform the following coordinate transformation:

$$p_h(k) = p(k) + d\begin{pmatrix} \cos\theta(k) \\ \sin\theta(k) \end{pmatrix} \tag{28}$$

where $p_h(k)$ denotes the position that is d-distance biased away from the actual position of the mobile agent, where d is chosen small enough. Then it follows that

$$\begin{aligned} & p_h(k+1) - p_h(k) \\ \approx & Tv(k)\begin{pmatrix} \cos\theta(k) \\ \sin\theta(k) \end{pmatrix} + dT\omega(k)\begin{pmatrix} -\sin\theta(k) \\ \cos\theta(k) \end{pmatrix} \\ = & T\begin{pmatrix} \cos\theta(k) & -d\sin\theta(k) \\ \sin\theta(k) & d\cos\theta(k) \end{pmatrix}\begin{pmatrix} v(k) \\ \omega(k) \end{pmatrix} \\ \triangleq & Tv_h(k). \end{aligned} \tag{29}$$

Note that equation (29) is in the single integrator form as studied in this paper and thus the proposed algorithm is applicable. The linear and angular velocity inputs can be calculated by:

$$\begin{pmatrix} v(k) \\ \omega(k) \end{pmatrix} = \begin{pmatrix} \cos\theta(k) & -d\sin\theta(k) \\ \sin\theta(k) & d\cos\theta(k) \end{pmatrix}^{-1} v_h(k). \tag{30}$$

The algorithm parameters and systems initial positions are given as follows. $p_d = (10, 10)$, $\rho_o = 4$, $N = 7$, $q(0) = (-10, -10)$, $p_1(0) = (-8.5, -8)$, $p_2(0) = (-7.5, -7)$, $p_3(0) = (-7.5, -8)$, $p_4(0) = (-7, -7.5)$, $p_5(0) = (-6.8, -8)$, $p_6(0) = (-6, -7)$, $p_7(0) = (-7, -6.5)$, $a_i = 0.1$, $\omega_i = 0.1$, $\alpha = 75$, $\beta = 0.375$, $\gamma = -0.5$, $\rho_n = 5$, $\rho_s = 0.2$, $\rho_r = 0.5$, $\rho_g = 2$, $\rho_d = 20$, $\theta_t = \frac{3\pi}{4}$, $\theta_l = \frac{\pi}{4}$, $\theta_r = -\frac{\pi}{4}$, $r_a = 3$, $\gamma_a = 15$, $\gamma_b = 12.5$.

Fig. 7 shows the initial positions of the sheep and the sheepdog, where the sheep are relatively scattered, and the sheepdog is behind the sheep herd. Fig. 8 shows the sheep herd are herded by the sheepdog. Fig. 9 shows the sheep herd are driven close to the sheepfold. Fig. 10 shows the sheep herd are herded to the sheepfold and the sheepdog stops.

5 Conclusion

This paper considers the sheep herd transport problem: a herd of sheep is driven by a single sheepdog from the start place to the sheepfold. The rules governing the behaviors of the sheep are preset and known to the sheepdog. Inspired by the behaviors of true sheepdogs, a backward semi-circle reciprocation algorithm featuring dynamic turn-around point selection has been proposed. It has been shown by numerical simulations that the sheepdog can successfully drive the sheep herd to the sheepfold. In the future, we will consider the problem of obstacle avoidance during the sheepherding.

References

[1] R. Pfeifer, M. Lungarella, F. Lida, Self-organization, embodiment, and biolocially inspired robotics, *Science Robotics*, 318: 1088–1093, 2007.

[2] I. Couzin, Collective Minds, *Nature*, 445: 715, 2007.

[3] Parrish J K, Edelstein-Keshet L. Complexity, pattern, and evolutionary trade-offs in animal aggregation. *Science*, 284: 99–101, 1999.

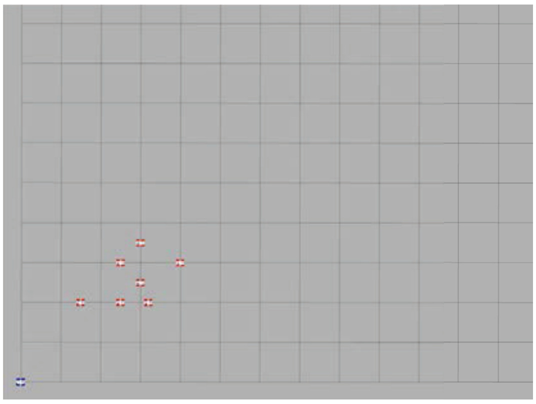


Fig. 7: The initial state of the sheep and sheepdog.

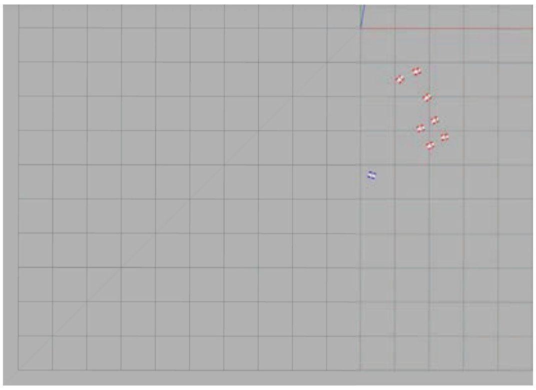


Fig. 8: The process of herding (a).

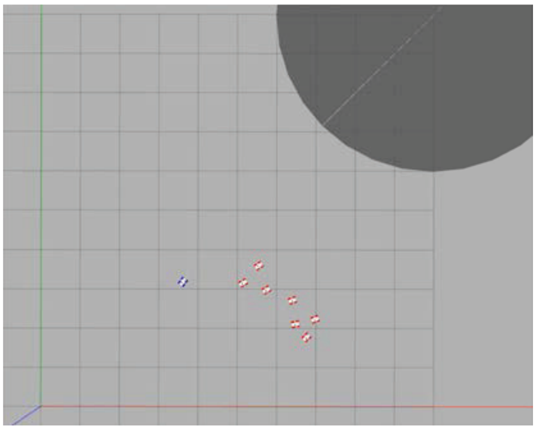


Fig. 9: The process of herding (b).

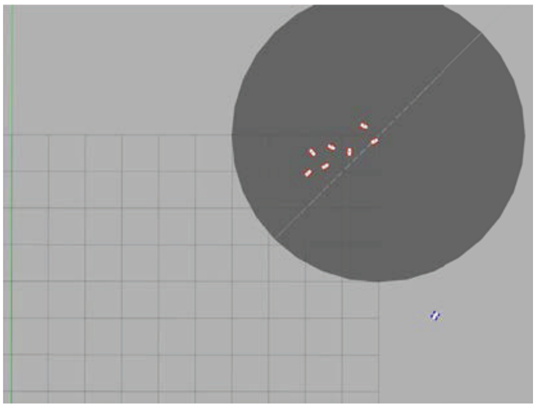


Fig. 10: The end of herding.

[4] Couzin I D, Krause J. Self-organization and collective behavior in vertebrates. *Adv Stud Behav*, 32: 1–75, 2003.

[5] Giardina I. Collective behavior in animal groups: Theoretical models and empirical studies. *HFSP J*, 2: 205–219, 2008.

[6] Vicsek T, Zafeiris A. Collective motion. *Phys Rep*, 517: 71–140, 2012.

[7] Aoki I. A simulation study on the schooling mechanism in fish. *Bull Jpn Soc Sci Fish*, 48: 1081–1088, 1982.

[8] Huth A, Wissel C. The simulation of the movement of fish schools. *J Theor Biol*, 156: 365–385, 1992.

[9] Huth A, Wissel C. The simulation of fish schools in comparison with experimental data. *Ecol Model*, 75: 135–146, 1994.

[10] Reynolds C W. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graph*, 21: 25–34, 1987.

[11] Couzin I D, Krause J, James R, et al. Collective memory and spatial sorting in animal groups. *J Theor Biol*, 218: 1–11, 2002.

[12] Couzin I D, Krause J, Franks N R, et al. Effective leadership and decision-making in animal groups on the move. *Nature*, 433: 513–516, 2005.

[13] Couzin I D, Ioannou C C, Demirel G, et al. Uninformed individuals promote democratic consensus in animal groups. *Science*, 334: 1578–1580, 2011.

[14] Huepe C, Aldana M. New tools for characterizing swarming systems: A comparison of minimal models. *Physica A*, 387: 2809–2822, 2008.

[15] Romanczuk P, Bär M, Ebeling W, et al. Active brownian particles. *Eur Phys J-Spec Top*, 202: 1–162, 2012.

[16] Bode N W F, Franks D W, Wood A J. Limited interactions in flocks: Relating model simulations to empirical data. *J R Soc Interface*, 8: 301–304, 2011.

[17] Bajec I L, Zimic N, Mraz M. Simulating flocks on the wing: The fuzzy approach. *J Theor Biol*, 233: 199–220, 2005.

[18] Buhl J, Sumpter D J T, Couzin I D, et al. From disorder to order in marching locusts. *Science*, 312: 1402–1406, 2006.

[19] Moussaid M, Helbing D, Theraulaz G. How simple rules determine pedestrian behavior and crowd disasters. *Proc Natl Acad Sci USA*, 108: 6884–6888, 2011.

[20] Peruani F, Deutsch A, Bar M. Nonequilibrium clustering of self-propelled rods. *Phys Rev E*, 74: 030904, 2006.

[21] Grossman D, Aranson I S, Jacob E B. Emergence of agent swarm migration and vortex formation through inelastic collisions. *New J Phys*, 10: 023036, 2008.

[22] Ginelli F, Peruani F, Bar M, et al. Large-scale collective properties of self-propelled rods. *Phys Rev Lett*, 104: 184502, 2010.