

New Strategy of the Shill: ‘Consistent Moving’*

HAN Jing¹, WANG Lin²

1. Key Laboratory of Systems and Control, Institute of Systems Science, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, No.55, Zhong Guan Cun East Road, Beijing 100190, P. R. China
E-mail: hanjing@amss.ac.cn
2. Department of Automation, Shanghai Jiao Tong University, No.800, Dong Chuan Road, Shanghai 200240, P. R. China
E-mail: wanglin@amss.ac.cn

Abstract: The collective behavior of a multi-agent system is the macroscopic phenomenon, such as swarm intelligence, flocking, synchronization, pattern formation, etc. This paper addresses the problem of ‘soft-control’: how to coordinate the collective behavior of multi-agent systems where the interaction rules of the already-existing agents can not be re-designed. One possible way to do is to add one or a few ‘shills’ which can be designed differently from the normal agent, but is treated as a normal agent by the other normal ones. As a case study, we propose a new strategy for the shill, called ‘consistent moving’, for the purpose of synchronization of the Vicsek’s model. This strategy avoids the limitation of ‘cheating heading’ in previous work. The computer simulation shows that it is possible for a ‘clever’ shill with ‘consistent moving’ strategy can change the collective behavior of a group.

Key Words: Collective Behavior, Multi-Agent System, Soft-Control, Synchronization, Vicsek’s Model

1 INTRODUCTION

Multi-agent methodology is a natural and popular way to model a system consisting of many locally interacting individuals(units). Collective behavior is the macroscopic phenomenon in the system level, different from the bottom level. It would be very different from what the collection of units would exhibit in the absence of the other units. For examples, phase transition, flocking/schooling/herding [1], pattern formation, swarm intelligence, synchronization, crowd panic [2], locust collective motion [3], group decision [4] and so on.

In a distributed multi-agent system, usually agents interact locally. The set of rules/functions/mechanism describing the interaction between two agents are called ‘local rules’. If the local rule is elaborately designed, the system will show expected and useful function, such as Swarm Intelligence. However, for some systems, the emergent collective behavior is not what we want.

There are at least two ways to intervene in the collective behavior:

1) Re-design the multi-agent system. This intervention is destructive to the system.

For example, re-design the local rule of the agents. Examples are formation control [5] for robots, ant colony algorithm [6], etc. By doing this, this system will self-organize to the desired collective behavior.

Put some special agents into the system, such as leaders [7] [8] and mediators [9], etc. These both require additional abilities of the already-existing normal agents to recognize and interact with the special ones.

2) Keep the current system and the intervention is non-destructive. Of all MASs, some systems can not be redesigned, nor can the local rules of the existing agents be changed. Meanwhile, in these decentralized systems there is no central controller who sends orders to agents, no global

parameter to adjust to change the collective behavior. In this case, how do we intervene in the system and guide the collective behavior? We should search other opportunity to put influence ‘softly’ on the system. For different system, the method would be different.

The soft-control framework[10] is just for the above problem: add one or a few special agents, called the shill, to allure the other agents in the system. The shill is not like the leader or mediator which is treated differently by normal agents. Obviously leaders have more influence on others. In the nondestructive intervention case, normal agents do not change their way to interact with others after one or some shills are added. For example, when a normal agent 1 interacts with agent 2, agent 1 does not need to know whether agent 2 is a normal agent or a shill. Agent 1 treats agent 2 as a normal one.

The similar idea of soft-control can be found in some approaches: Couzin et.al. [4] study how a few individuals(can be regarded as shills) with information about where to go influence the moving decision of animal groups. In [15], some robotic cockroaches are added to influence the shelter selection of real cockroaches. But in these two approaches, the strategies of shills are very simple. Neither the informed individual nor the robotic cockroach accepts feedback which is crucial to intelligence.

In this paper we approach soft-control with a subtle and ‘clever’ shill by a case study. The multi-agent system we considered is the vicsek’s model, which is widely studied by the physicist and the mathematician and control scientist in the current decade. In this model, each agent moves with a constant speed and its heading is updated based on the average headings of its neighbors(includes itself). It is a simple but nontrivial model, showing rich phenomena, for example, flocking/schooling/herding behavior, strong coupled and dynamical interaction network, phase transition of nonequilibrium system. So far as we know, the sufficient and necessary condition for synchronization is unknown yet. But this does not prevent our approach of intervention to the system even though the theory of self-organized behavior is not fully

*This work was supported by National Nature Science Foundation under Grants 60334040 and 60221301, and also supported by Knowledge Innovation Project of CAS under Grant KJCX3-SYW-S01.

built.

There are some related works of soft-control on the vicsek's model. Paper [16] adds a special agent which always maintains the desired moving direction to guide the group based on the linear-Vicsek's model. They point out that the key point is to maintain the connectivity of union of neighbor graphs consisting of all agents within some contiguous and bounded time intervals. However, they do not provide the algorithm of how the special agent moves to guarantee such connectivity. Liu et.al. [24] studied the case of adding numbers of special agents that always moving with the desired direction to guide the group. They give a theoretical proof of the proportion of special agents needed to guide the group. So in their work, a number of 'simple' shills are added, instead of a 'clever' shill. Their study focuses on the number of shills, not the strategy of the shill. In [10], we see how a shill is added into the system and guide the group to synchronization. A strategy was proposed there, but it has a major problem: the moving direction is not consistent with the heading of the shill. So it is like a 'jumping' shill with 'cheating heading'. Thus it is not a complete algorithm. In this paper, a new strategy called 'consistent moving' of the shill is proposed and the above problem is solved. The shill is moving in the direction which is consistent with the heading. It solves a basic problem of how a shill moves from one location to another location without putting negative effects on the group. The computer simulation shows that the system synchronize by using a shill with the new strategy.

This paper is organized as follows. In Section 2 the vicsek's model is described and the current study of the self-organize behavior about synchronization is introduced. Then in Section 3, the main part of this paper, the new strategy of the shill to guide the system synchronization is presented. The computer simulations of this new strategy are given in Section 4. We finish with some conclusions and comments of future applications of soft-control in Section 5.

2 THE VICSEK'S MODEL

In 1995, Vicsek et al. introduce a multi-agent model. It is actually a simplified version of the Reynolds's model [1] which only keeps the alignment local rule - steer towards the average heading of neighboring agents. Two agents are neighboring to each other if the distance between them is not larger than a given constant r . In spite of its simplicity, the self-order motion, a flocking-like cooperative behavior, emerged in this simplified system. It has been recently studied in the areas of physics [11] [12], control [16]- [21], robots [17] and mathematics [22]- [26]. By their efforts, more and more theoretical parts of this model are revealed but many still remain unknown.

In this discrete-time multi-agent system, there are n agents represented by heading set $\theta = (\theta_1, \theta_2, \dots, \theta_n)$ and location set $X = (X_1, X_2, \dots, X_n)$, where $X_i = (x_i, y_i) \in R^2$ denotes the x -coordinate and the y -coordinate of the location of agent i . They are all moving simultaneously in the two-dimensional space. The velocity of agent i at time t is $v_i(t) = (v \cos(\theta_i(t)), v \sin(\theta_i(t)))$, constructed to have a constant absolute value v and a heading given by the angle $\theta_i(t) \in [0, 2\pi]$. Neighbors of agent i at time t are those agents which are either in or on a circle of a constant radius

r centered at agent i 's current position. So the neighborhood is defined as

$$N_i(t) = \{j \mid \|X_j(t) - X_i(t)\| \leq r\}.$$

Agent i updates its heading and position according to (noise is ignored in this paper):

$$\theta_i(t+1) = \arctan\left(\frac{\sum_{j \in N_i(t)} \sin(\theta_j(t))}{\sum_{j \in N_i(t)} \cos(\theta_j(t))}\right), \quad (1)$$

$$X_i(t+1) = X_i(t) + v_i(t). \quad (2)$$

So $\theta_i(t+1)$ is the angle of the sum of the velocity vectors of neighbors of agent, not simply taking the average of all headings (see footnote 2 and 3 in [10] for details).

The computer simulation of this model shows flocking behavior. Given different initial configurations, the system will exhibit different phenomena: agents soon self-organize to a same heading, i.e., synchronization; agents separate into several groups and never merge together. One of the most important and basic questions is: in what condition, the system will self-organize to the same heading — synchronization? Many scientists have been working on this problem [16] [18]- [26]. Yet, the necessary condition is still unknown. The math behind this model is not trivial because of the nonlinear dynamical relationship although the model seems simple. Knowing more about the self-organized behavior usually will help us to design the intervention method. But this is not necessary before we begin our approach of intervention to the system. We can try to control the system even though the property of self-organized behavior is not totally understood. Note that this doesn't mean the model of the system is unknown, nor is global information unknown. In fact in the strategy this paper proposed, the shill knows all information (locations and headings of normal agents) of the system at every time step. In this paper, we try to intervene in the system to guide it to synchronization even though the necessary condition for synchronization of the system is unknown yet. This shows the difference between approaches of self-organization and intervention.

3 DESIGN OF THE SHILL

As we know from the simulation, starting from many initial configurations (initial locations and headings of n agents) of the Vicsek's model, agents will not self-organize to a same heading. In this case, how we intervene in and guide the system to synchronization without any destruction to the local rules? Synchronization is an important and basic task for a multi-agent system in many real world applications, such as robots formation. Moreover, once the system synchronizes, usually it is easier to guide the group to turn or move to a destination.

The soft-control idea suggests adding a special agent, called a shill, into the system. The shill will affect a normal agent if it is in the neighborhood of that agent, i.e., distance between them is not larger than r . The strength of the effect of a shill is as same as normal agents. So the power of a shill is limited, but it can dramatically change the collective behavior.

The key question is how we design the strategy/algorithm of the shill. A strategy should include two basic parts:

(1) The scheme of target agent selection. How does the shill select the next target agent to affect? What are the criteria?

(2) The moving route of the shill from one location to the next target agent location. This is not an obvious problem. Note that if the shill is not always moving with the desired direction, it will have some negative effects on the neighboring normal agents. However, the shill could never get to some places if it can only move with a fixed direction.

In [10], a strategy for the shill called u_β is proposed to guide the system to synchronize to the direction of π : in every time step, the shill jumps to the current ‘worst agent’ and affects it with a fixed heading of π . ‘worst agent’ is the agent with a heading which is farthest away from π . This strategy is proved analytically and numerically to lead the asymptotic synchronization of the system if the initial headings of the group belong to $[0, \pi]$. However, there is a major problem in the strategy: the shill is jumping from one location to another with a ‘cheating heading’, because the moving direction is not always consistent with the heading of the shill.

Problem definition: considering the vicsek’s model consisting of n normal agents with initial headings of $(-\pi/2, \pi/2)$ randomly distributed inside a finite size area (a circle with radius R), a shill is added to guide all normal agents synchronized to heading angle zero. What is the strategy for the shill?

The basic idea of ‘consistent moving’ for the shill is:

1. Speed of the shill can be higher than the speed of normal agents. Speedup or slowdown is allowed during the time. But there is a maximal shill speed.
2. The shill periodically affects every agent with heading zero. In a period of M steps (or less than M steps), all agents are affected by the shill at least once.
3. When it moves from agent 1 to the next target agent 2, it first moves forward(moves to the right side) with a faster speed with heading zero till it is not in any agent’s neighborhood, then it tries to move to the left side of the next target agent and affects that agent with heading zero (see Fig. 1).
4. It should always keep heading zero if it has neighboring agents; only when it is not in any agent’s neighborhood can it have a non-zero heading. So the effect of the shill to the group is always positive, i.e., making the group more close to heading zero.

3.1 Moving Route

The simplest idea for a shill to move from one location to another target agent location is shown in Fig. 1(a). With a much faster speed the shill can make this kind of route so it will not meet any normal agents during the U-turn even considering the normal agents are moving. Obviously the route of Fig. 1(a) is not very efficient. In fact shorter routes are found and shown in Fig. 1(b). The ‘U-turn’ can go inside

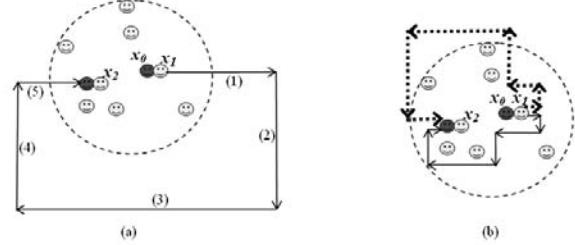


Fig. 1 An example of the moving route from agent 1 to agent 2 of a shill. (a) The circle indicates the group area. After the shill affects agent 1, it moves to affect agent 2: (1) First it goes forwards to a location which is much far away from the whole group, then it makes a big U-turn (2)-(3)-(4) far away outside the group area and gets back to the left side of the whole group, (5) finally goes forward and affects agent 2. Its heading is always set to be zero in part (1) and (5), while it can have different headings during (2)-(3)-(4). (b) Two more efficient routes (solid-line route bypasses from underneath and dash-line route bypasses from above) for the shill moving from agent 1 to agent 2. From location of agent 1 to the left side of agent 2 location, it tries to find a shorter route which maintains a ‘safe’ distance (larger than r) away from any normal agent when its heading is not zero. Note that for simplicity, figure (a) and (b) suppose all normal agents do not move, while showing the idea of the shill route. But the real consistent moving strategy will consider that the normal agents are moving when the shill moves, which is much more complicated.

the group area with a more ‘aggressive’ path as long as it avoids meeting any normal agents. This shortcut is more subtle in this case. Some heuristics are used in the algorithm of ‘consistent moving’ for this purpose. The detailed idea of how a shill decides its heading at every time step is shown as a state transition diagram in Fig. 2. Explanations of Boolean condition functions for each transition are listed below:

Affected: it returns true after the target is affected by the shill.

TargetAhead: it returns true if keeps moving with heading zero the shill will hit the target agent.

Around(k): it returns true if after k steps, the neighborhood of the shill will be not empty (i.e., the shill might affect one or some normal agents).

TargetAboveCentroid: it returns true if $\text{target}.y > \text{avg}.y$. Where $\text{target}.y$ is the y -coordinate of the target agent, and $\text{avg}.y = (\sum_{i=1}^n y_i)/n$ is the y -coordinate of the centroid of n normal agents.

TargetUp: it returns true if the location of the target agent is above the shill.

TargetLeft: it returns true if the target agent locates in the shill’s left side.

AroundArea: it returns true if it will not hit any normal agent during its movements upwards/downwards to the horizontal position of $\text{target}.y$. In the case we studied here(i.e.,initial headings of all normal agents belong to $(-\pi/2, \pi/2)$), agents need to be considered are inside a triangle area.

otherwise: This is not a specified condition function. It is true if conditions on other transitions from this state are all false. For example, if from state 1, it does not satisfy conditions to transit to state 0 or state 2, it will keep state 1.

The computation of each Boolean condition function is not high. *Affected*, *TargetAhead*, *TargetAboveCentroid*, *TargetUp* and *TargetLeft* are very simple. To reduce the computation, *Around(k)* and *AroundTriangleArea* are considered in the worst case, because exact prediction is not required in this heuristic. For example, when *Around(k)* is checked, it is not necessary to simulate the whole system for k steps to see whether after k steps the shill will meet any agent. Instead, we check whether there are agents inside the circle of radius kr .

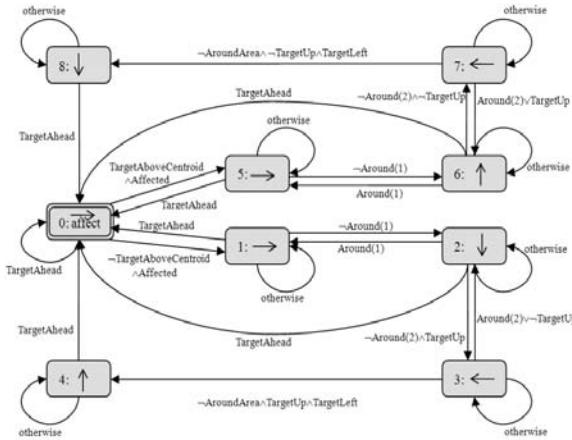


Fig. 2 State transition diagram of the moving direction decision of the shill in ‘consistent moving’. The heading of the shill in each state is indicated by the arrow inside the state box. For example, heading of state 0, state 1 and state 5 is zero, while heading of state 2 and state 8 is $-\pi/2$. State 0 is the beginning and ending state, once a target agent is affected, the new loop begins for the next target agent. Loop of state $0 \leftrightarrow 1 \leftrightarrow 2 \leftrightarrow 3 \rightarrow 4 \rightarrow 0$ is for the route bypasses from underneath. Loop of state $0 \leftrightarrow 5 \leftrightarrow 6 \leftrightarrow 7 \rightarrow 8 \rightarrow 0$ is for the route bypasses from above.

3.2 Target Agent Selection Schedule

Another heuristic is about the schedule of selecting target agent in a period. The simplest schedule in a period is to affect agents in a fixed order: $1 \rightarrow 2 \rightarrow 3, \dots, \rightarrow n$. But we found from simulations that if an agent shows the trend of moving far away from the group, the shill should select it with high priority. Otherwise the distance will become very large and the shill will have to take much more time to catch up with that agent. So the agent which has the biggest distance growth from the group will be picked up first. The heuristic criteria can be represented as: at time step t , agent j satisfies formula (3) is selected.

$$j = \arg \max_{\{i \in \{1, \dots, n\} \text{ and not affected}\}} \{\|C(t+m) - X_i(t+m)\| - \|C(t) - X_i(t)\|\}, \quad (3)$$

where $C(t+m)$ and $X_i(t+m)$ are the locations of the group centroid and agent i after m steps. Again, to reduce the com-

putation, approximate prediction of centroid $\hat{C}(t+m)$ and $\hat{X}_i(t+m)$ after m steps are used here. In the simulations, m sets to be 50, which is about the average number of steps for a shill to move from one target agent to the next target agent. For example, the approximation can be:

$$\begin{aligned} \hat{C}(t+m) &= C(t) + \sum_{i=1}^n v_i(t)/n \times m, \\ \hat{X}_i(t+m) &= X_i(t) + v_i(t) \times m. \end{aligned}$$

In this case, the complexity of the approximation is $O(1)$. So the selection schedule complexity is $O(n)$.

In fact, if agent j is affected by the shill, agents directly or indirectly neighboring to j can be regarded as affected. Obviously, this will shorten the period since a shill may affect several agents in one time step.

3.3 The Complete Algorithm

The following roughly shows the idea of the complete algorithm for ‘consistent moving’. After initialization of n normal agents and the shill, the system evolves until it is synchronized, i.e., $\max_i |\theta_i - \theta^*| = \max_i |\theta_i - 0| = \max_i |\theta_i| \leq \epsilon$. During evolution, normal agents update their heading and position according to formula (1) and (2). Meanwhile, the shill is moving according to the ‘consistent moving’ strategy. Once a target agent is affected, the procedure of target agent selection will return the next target agent from the unaffected-agent set in this period. After all agents are affected, a new period begins. The program will stop when none of the absolute headings is larger than ϵ . For convenience, we denote $X_0(t)$, $V_0(t)$ and $\theta_0(t)$ as the position, absolute value of the velocity and the heading of the shill at time t respectively.

Procedure main

0. $step=0;$
- initialization:
randomly distribute a shill and n agents with random headings in $(-\pi/2, \pi/2)$;
- $state = 0;$ //state in the state transition diagram of Fig.2
- $\theta_0(0) = 0;$ //heading of the shill
- $j = target_agent_selection;$
1. **while** $\max_i |\theta_i| > \epsilon$ **do** //check whether the group is synchronized
 2. $step = step + 1;$
 3. **if** $j = 0$ **then** $j = target_agent_selection;$
 4. updates $\theta_0(t)$ and $state$ according to the state transition diagram
if $affected=true$ **then** $j = 0.$
 5. updates headings and positions of n agents according to formula (1) and (2).
 6. updates position $X_0(t)$ of the shill.
7. **end while.**
- Procedure** *target_agent_selection*
1. **If** $unaffected-set=\emptyset$ **then** //all agents are affected in this period
 $unaffected-set=\{1, \dots, n\}$ // a new period begins

2. select j from *unaffected-set* according to formula (3)
3. removes j from *unaffected-set*
4. **return** j

4 COMPUTER SIMULATIONS

The goal of the computer simulation experiments is to empirically demonstrate that a shill using the strategy of ‘consistent moving’ can efficiently guide the system to synchronization. Specifically, the goal of this section is fourfold:

a) Can it synchronize the system within limited speed? Total number of steps for synchronization($\max_i |\theta_i| \leq \epsilon$) is measured.

b) Can it move from one agent location to the next target agent location within limited steps? Maximal number of steps from one target agent location to the next target agent location, and maximal number of steps of a period during each run are measured.

c) How do headings of agents change? $\max_i |\theta_i|$ at each time step during each run are observed.

d) How does the shill speed v_s affect the performance?

For each run, some quantities are measured:

totalStep — the total number of steps for convergence($\max_i |\theta_i| \leq \epsilon$);

maxOne — the maximal number of steps for moving from one target agent to the next target agent;

maxPeriod — the maximal number of steps for a period. In simulations, the total steps for the shill to affect all agents can be different every time. **maxPeriod** is to measure the maximum during one run. Usually it is much less than the period bound M.

maxHeading(t) — $\max_i |\theta_i(t)|$, the worst heading at every time step.

Tab. 1 Avg. values of maxOne, maxPeriod and totalStep for $v_s = 10v(10000runs), 20v(100000runs), 30v(10000runs) and \infty(10000runs)$. Value in bold font refers the minimum among different v_s ($10v, 20v, 30v$).

	N=30			N=50			N=100		
	maxOne	maxPeriod	totalStep	maxOne	maxPeriod	totalStep	maxOne	maxPeriod	totalStep
v_s	10v	169.45	1226.21	1797.70	195.29	1627.90	2692.22	217.30	1909.35
	20v	116.12	1160.69	1458.27	133.91	1782.10	2155.78	156.00	2511.96
	30v	97.90	995.41	1455.35	114.91	1684.45	2092.90	137.59	2669.18
	∞	50.96	371.34	631.20	63.46	737.46	1027.09	76.25	1633.12

The simulation results are shown in table 1. They all tell us that the shill with the ‘consistent moving’ strategy and limited speed can lead the system to converge to synchronization. The shill can move from one agent location to the next target agent location within limited steps.

Table 1 compares the avg. values of *maxOne*, *maxPeriod* and *totalStep* for $v_s = 10v, 20v, 30v$, as well as ∞ — the extremely case of very large speed. All minima lie in the row of $v_s = \infty$. If $v_s = \infty$ is excluded, almost all minima are related to $v_s = 30v$, while almost all the maxima are related to $v_s = 10v$. So we can conclude that higher speed of the shill usually leads to faster convergence. This is consistent with people’s intuition.

Figure 3 shows the convergence trend during evolution. Avg. values for 100,000 runs of $\max_i |\theta_i(t)|$ are measured

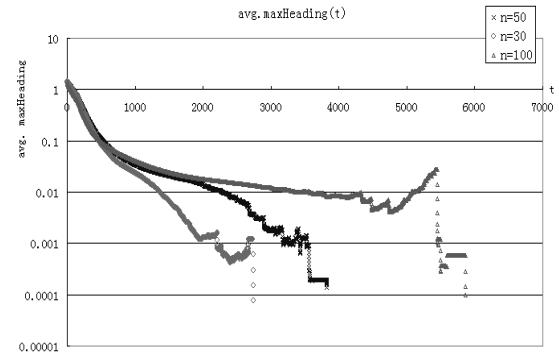


Fig. 3 Avg. value of $\max_i |\theta_i(t)|$ in log scale for $n = 30, n = 50$ and $n = 100$.

at each time step t and plotted in a log scale. It shows that the system converges very fast in the beginning, i.e., $\max_i |\theta_i(t)|$ gets very close to zero (about 0.1) within the first 500 steps, and then the speed of convergence slows down. Convergence is slower with larger n . The tail parts of all three different n are interrupted. This is because not many data are collected in this region since it is far away from the avg. value of *totalStep*. In many runs, the system has already stopped evolution since it has reached convergence.

5 CONCLUSIONS

This paper is a case study of ‘soft-control’. We proposed a new strategy called ‘consistent moving’ for the shill to coordinate synchronization of a multi-agent system defined by the Vicsek’s model. This new strategy solved the problem of ‘cheating heading’ in the previous work. It is the first complete and feasible algorithm for the shill. It is shown to be effective through computer simulations. In fact, by mathematical analysis we can prove (1) the shill can guide all normal agents to a desired heading, and (2) locations of all normal agents can be covered by a circle with a fixed radius at all time steps during the whole evolution. But due to the space limitations, the mathematical proof is omitted.

The ‘consistent moving’ strategy includes a method of how a shill moves from one agent location to another agent location without putting any negative effect on the group. This will be a base of further development of more efficient strategies for the shill, or strategies for other coordination purposes(such as circus movement of group, tracking of desired route, avoid obstacle, etc.) of the Vicsek’s model.

The case study in this paper is to lead the group to heading zero. In fact, the ‘consistent moving’ shill can synchronize the system to any heading, with some minor modifications on the state transition diagram. In fact, the main purpose of this paper not only lies on a new strategy of the shill for a special coordination task of a specific MAS, but also wants to drop a hint and suggest that the idea of soft-control is a potential method for coordination of many MASs as well. In the future, except to further refine the shill strategy for intervention to the Vicsek’s model, we will also consider applications of soft-control in other MASs. Not only man-made systems, but also social systems. For example, soft-control in a multi-player game to intervene in the equilibrium, soft-

control in crowd to avoid panic, etc.

On the other hand, the soft-control idea also provides a possible way for design of man-made MASs. Because totally depending on the locally interacting agents to achieve an expected collective behavior requires a lot of skill in designing the local interaction rule, add a ‘clever’ shill to coordinate the collective behavior will release us from the subtle design. Therefore the complexity of normal agents is also reduced. For example, designing a MAS with large amount of locally interacting microscopic robots aiming to form dynamic specified shape [27] is not easy, because agents have to share information about the whole group through locally interaction. But if an intricate shill is added, those agents may be simpler(their interaction rule may be just like the alignment rule in the vicsek’s model). Other possible applications would be helping design of swarm-like algorithms(the ant colony algorithm, the AER algorithm [28] for combinatorial optimization) by adding shill.

Under the framework of soft-control, there are lots of future studies. We hope through continuously investigations, a general theory of soft-control will be built step by step.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China (No. 60574068, No. 60821091, and No.60804043) and the Knowledge Innovation Program of the Chinese Academy of Sciences (Grant No. KJCX3-SYW-S01).Thanks to Dr. Wang Xin for his proof-reading.

REFERENCES

- [1] C. Reynolds. Flocks, birds, and schools: a distributed behavioral model. *Computer Graphics*, 1987, 21:25-34. The demo can be downloaded from <http://www.red3d.com/cwr/booids/applet/>.
- [2] D. Helbing, I. Farkas, T. Vicsek. Simulating dynamical features of escape panic, *Nature*, 2000, 407: 487-490.
- [3] J. Buhl, D.J.T. Sumpter, I.D. Couzin, et.al.. From disorder to order in marching locusts. *Science*, 2006, 312:1402 -1406.
- [4] I.D. Couzin, J. Krause, et.al.. Effective leadership and decision making in animal groups on the move, *Nature*, 2005, 433:513-516.
- [5] N. Ceccarelli, M. Di Marco, A. Garulli, and A. Giannitrapani. Collective circular motion of multi-vehicle systems, *Automatica*, 2008, 44 (12):3025-3035.
- [6] A. Colorni, M. Dorigo, and V. Maniezzo. Distributed optimization by ant colonies, in: F.J. Varela and P. Bourgine (Eds.), *Pro. of the 1st European Conference on Artificial Life*, Cambridge, MA, 1992: 134-142.
- [7] Y. Hong, G. Chen, L. Bushnell. Distributed observers design for leader-following control of multi-agent networks, *Automatica*, 2008, 44(3):846-850.
- [8] S. Mu, T. Chu, L. Wang. Coordinated collective motion in a motile particle group with a leader, *Physica A*, 2005, 351:2-11.
- [9] D. Monderer, M. Tennenholtz. Strong mediated equilibrium, *Artificial Intelligence*, 2009, 173(1):180-195.
- [10] J. Han, M. Li, and L. Guo. Soft control on collective behavior of a group of autonomous agents by a shill agent, *J.Systems Science and Complexity*, 2006, 19(1):54-62.
- [11] T. Vicsek, A. Czirok, I.B. Jacob, et.al.. Novel type of phase transitions in a system of self-driven particles, *Phys.Rev.Lett.*, 1995, 75: 1226-1229.
- [12] A. Czirok, A.L. Barabasi, and T. Vicsek. Collective motion of self-propelled particles:kinetic phase transition in one dimension. *Physical Review Letters*, 1999, 82:209-212.
- [13] H. Kunz, C.K. Hemelrijk. Artificial fish schools: collective effects of school size, body size, and body form, *Artif. Life*, 2003, 9(3):237-253.
- [14] M. Ballerini, N. Cabibbo, et.al.. Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study, *Proc. National Academy of Sciences*, 2008, 105:1232-1237.
- [15] J. Halloy, G. Sempo, G. Caprari, et al.. Social Integration of Robots into Groups of Cockroaches to Control Self-Organized Choices, *Science*, 2007, 318:1155-1158.
- [16] A. Jadbabaie, J. Lin, A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules, *IEEE Trans. Automat. Contr.*, 2003, 48:988-1001.
- [17] H. Yu, Y. Wang. Coordinated Collective Motion of Groups of Autonomous Mobile Robots with Directed Interconnected Topology , *Journal of Intelligent and Robotic Systems*, 2008, 53(1):87-98.
- [18] W. Ren, R.W. Beard, Consensus seeking in multi-agent systems under dynamically changing interaction topologies, *IEEE Trans. on Automatic Control*, 2005, 50(5):655-661.
- [19] M. Cao, A.S. Morse, and B.D.O. Anderson. Reaching a Consensus in a Dynamically Changing Environment: A Graphical Approach, *SIAM Journal on Control and Optimization*, 2008, 47(2):575-600.
- [20] M. Cao, A.S. Morse, and B.D.O. Anderson. Reaching a Consensus in a Dynamically Changing Environment: Convergence Rates, Measurement Delays and Asynchronous Events, *SIAM Journal on Control and Optimization*, 2008, 47(2):601-623.
- [21] L. Moreau. Stability of multiagent systems with time-dependent communication links, *IEEE Trans.Autom.Control*, 2005, 50(2):69-182.
- [22] Z.X. Liu, L. Guo. Connectivity and synchronization of Vicsek model, *Science in China Series F: Information Sciences*, 2008, 51(7):848-858.
- [23] L. Wang, L. Guo. Robust consensus and soft control of multi-agent systems with noises, *Journal of Systems Science and Complexity*, 2008, 21(3):406-415.
- [24] Z.X. Liu, J. Han, X.M. Hu. The number of leaders needed for consensus, to appear in *Proceedings of 48th IEEE Conference on Decision and Control*, 2009
- [25] F. Cucker, S. Smale. Emergent behavior in flocks. *IEEE Transactions on Automatic Control*, 2007, 52(5):852-862.
- [26] G.G. Tang, L. Guo. Convergence analysis of linearized Vicsek’s model, in: *Proc. of the 25th Chinese Control Conference*, Harbin, China, 2006:379-382.
- [27] S.C. Goldstein, J.D. Campbell, T.C. Mowry. Programmable Matter. *IEEE Computer*, June, 2005, 38(6):99-101.
- [28] J. Liu, J. Han, and Y. Y. Tang. Multi-agent oriented constraint satisfaction. *Artificial Intelligence*, 2002, 136(1):101-144.

Online supplementary material

Vicsek’s model consists of 30 normal agents. First the video shows the self-organized behavior without shills. As the result, the group does not synchronize. Then a shill using the consistent strategy is added to the group start evolving from the same initial configuration. The shill tries to guide them to direction zero (all move to the right side). Big black circle indicates the initialization area. The normal agent is represented by the little dot with red line pointing to the heading direction. The shill is represented by a little dot with blue arrow pointing to the heading direction, and the blue circle around it indicates the neighborhood area. The position of the centroid of the group is represented by a bigger black dot.

This vedio can be downloaded from

<http://complex.amss.ac.cn/hanjing/softcontrol/softcontrol.avi>.