2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Animal Group Behavioral Model with Evasion Mechanism

Zhiping Duan
Department of Electronic Engineering, Fudan University
Shanghai 200433, China
duanzp@fudan.edu.cn

Xiaodong Gu
Department of Electronic Engineering, Fudan University
Shanghai 200433, China
xdgu@fudan.edu.cn

***Abstract*— Modeling behavioral mechanism of animal group promotes the development of group animation and other fields involving crowd simulation. This paper introduces a model to mimic behaviors of animal group. We proposes a swarming intelligence algorithm, Evasion Mechanism Artificial Fish School Algorithm (EM-AFSA) in our model, in which AFSA often focuses on optimization. The EM-AFSA introduces a new mechanism, i.e. evasion, which enables the group to avoid obstacles and collisions and to evade predation. It also includes flocking, foraging and tailgating. It is convenient to show the dynamic demonstration of our model and the model vividly mimics the real animal group behavior, which could potentially be used in designing group animation.**

Keywords—EM-AFSA; swarm intelligence algorithm; group animation; group behavior; evasion mechanism

I. INTRODUCTION

The past decades witnessed a flourish in distributed artificial intelligence (DAI) theory, which mainly concentrates on the research of structure and coordination of concurrent multi-agent. To some extent, distributed artificial intelligence structure is 'smarter' than traditional artificial intelligence. Because in the real world, knowledge and resources of a single agent is spatial and temporal limited. So social coordination and cooperation within the colony is necessary for members who want to achieve their own ends. DAI not only relates to the theoretical basis of joint activities, distributed computing model and computational ecology, but also involves practical applications [1]. One of these applications is group animation.

From the very beginning, computer animation was drawn by hand frame by frame, which cost a lot of time and human resources. Then the key-frame technology arose and liberated animators from amount of repeating labor. But it is still an unrealizable job to modify and display group activities if the number of the group is large. Designing motion path and behavior for each individual in the group was not feasible until C.W.Reynolds proposed the concept of group animation in [2]. In the bird-oid (BOID) model he proposed, he defined three behavior, i.e. cohesion, alignment and separation. Each individual can make decisions on its behavior next step according to the state of other individuals nearby and the current state of its own synthetically. From then on, the development of group animation accelerated and related swarm intelligence algorithms also thrived.

This work was supported in part by National Natural Science Foundation of China under grant 61371148 and Shanghai National Natural Science Foundation under grant 12ZR1402500.

Another important model that should not be ignored is Artificial Fish proposed by Tu [3], [4]. Just like the BOID, each fish can be regarded as an Agent, observing rules animators defined. But it also has vital signs such as intention, habit, perception and attention. Based on computer graphic technology, the model can simulate the visual complexity and randomness of natural fish colonies, which satisfies the demand of visual realism well and creates vivid animation.

Nowadays, the research on group animation mainly targets on two directions: simulating colonies and behavior control. The former focuses on modeling group activities for applications in Artificial Life, Human–Machine Interaction, Virtual Reality, and so on. While behavior control focuses on the algorithms and rules of colonies, which are important for simulating traffic scenes and evacuation in crowded public places, searching for best paths, and modeling autonomous intelligent vehicles. In fact, these two directions is inseparable [5], [6] if we ignore their difference in applications.

Apart from the models mimicking natural groups, swarm intelligence algorithm mentioned before, which also shares the idea of DAI, is another research direction impelled by natural colonies. For instance, Particle Swarm Optimization (PSO) proposed by Eberhart and Kennedy in 1995 [7], [8] originates in mimicking the predation process of birds. Based on the principle of information sharing among members within the group, the whole colony evolved into a stable optimal state from disorder ones in solving space. Ant Colony Optimization (ACO) proposed by Marco Dorigo [9], [10] in 1999 originates in behaviors of ants searching for food, which is often used in finding the optimal path in the graph. Besides, we also see many other swarm intelligence algorithms, such as Artificial Bee Colony Algorithm (ABC) [11] and others related to bee colony, Bacterial Foraging Optimization (BFO) [12] and Bacterial Chemotaxis Algorithm (BC) [13], Glowworm Swarm Optimization (GSO) [14], as well as Artificial Fish School Algorithm (AFSA) [15], [16], and so on. All these algorithms focus on bionic optimization for practical engineering problems with characteristics like high dimension, multi-extremums, non-linearity, and undifferentiability.

Rules designed for agents in group animation are usually complicated. So people developed ways substituting for this huge project. For example, some example-based methods [17]-[19] and data-driven methods [19]-[24] were developed to simplify the process of making group animation.

It is natural to assume that applying improved swarm intelligence algorithm to group animation should be a way alternative. In this paper, for the purpose of combining swarm intelligence algorithms with group animation, we proposed EM-AFSA based on AFSA by bringing in evasion mechanism which enhances the visual complexity and realism of the model. Based on the algorithm, we establish models to simulate behaviors of animals, such as avoiding obstacles, avoiding collisions, flocking, foraging, tailgating, predation and evading predation, etc.

It is always difficult to develop verification platforms for making group animation, which is always time-consuming and requires a lot of computational resource. The high hardware requirements and long development cycle are not endurable when we want to verify algorithms proposed. In this paper, we also establish a platform which makes it convenient to realize the dynamic demonstration of swarm intelligence algorithm applied in group animation. In the platform, the dynamic effect of the improved algorithm is simulated. All behavioral mechanisms, especially the evasion mechanism, are tested. We also make experiment to test the effect of the model's target function.

The remainder of this paper is organized as follows. The second section introduces the overall plan to establish simulation system. The third section explains the algorithm we have improved and the whole model of animals' behavioral mechanism. Experimental results are given in the fourth section. Finally, the last section concludes our work and gives directions for further research.

II. ESTABLISHING A VERIFICATION PLATFORM

It is always a time-consuming and computational expensive job to establish a group animation simulating system if we take the vivid visual effect into consideration. In fact, it is not necessary to spend a lot of effort to develop an extreme perfect and vivid system when we just need a platform to verify algorithms. So, a scheme which is easy to construct and beneficial to quickly and conveniently demonstrating the dynamic effect of algorithm is needed.

A. Virtual World Modeling

The virtual reality scene used for dynamic demonstration is held in an ASCII file. It can be made by directly programming with virtual reality modeling language (VRML). But for cases containing objects whose number is large or with complex shapes, directly programming to complete the modeling will be a mission impossible. Some 3D modeling softwares help us to realize the modeling. For example, some products of Autodesk such as Maya, 3D-MAX and AutoCAD all support virtual reality modeling and can export files with format we need. Matlab also contain toolbox integrated with V-Reaml builder and 3Dworld editor, which are helpful to edit and modify the model we made.

Each ASCII file contains some specific father nodes. The WorldInfo node contains document title, author, copyright and other identity information. The background node contains information describing the background of the model. The switch node includes many subentries, which contain the vertex information and the three-dimensional coordinates of these vertex. Meanwhile, the group node consists of child nodes. In the virtual reality model, roles such as animal individuals, obstacles, and predators are all bonded in group node, so they are child nodes in the model. Each child node consists of several fields, which define the viewport, location, spin, scale, appearance and other characteristics of the object respectively. For instance, by controlling the translation field and rotation field, we can change the location and direction of corresponding objects. In the platform we built, there are mainly three kinds of roles, i.e. zooplankton colonies, predators and obstacles.

B. Interface Function

The interface function bridges the virtual reality world and algorithms describing the behavioral mechanisms. By invoking relevant functions, we can get and control the property parameters we want and have objects in the virtual reality world behave the way our algorithms define.

In Matlab, there are limited numbers of functions that control or describe the Vrworld. But in our scheme, it's completely enough. Commands that are frequently used are listed in TABLE I.

In practice, we usually have to control quantities of members in the group and it is not wise to call these commands respectively for each member. It is necessary to write loop programs to get the parameters or change them conveniently. It is a feasible schedule to get all the parameters relevant in the beginning and have them stored in data bank. The algorithm works on the data bank and modifies the data in the mechanism designed. Then we write the data relevant into the Vrworld and realize the purpose of controlling behaviors of target members.

TABLE I. COMMANDS ON VRWORLD

Function	Operation corresponding
close()	Close a Vrworld
delete()	Delete a closed Vrworld from memory
get()	Get parameters of a Vrworld or a node
nodes()	List the nodes available in Vrworld
open()	Open a Vrworld
reload()	Reload the Original Vrworld
save()	Save a Vrworld to a VRML file
set()	Set properties of a Vrworld or a node
fields()	List the fields of a node
vrnode()	Create a new node in the Vrworld

We will describe the animal group behavioral model in detail in next section.

III. GROUP BEHAVIORAL ALGORITHM ADDED EVASION MECHANISM

A. Principles of Behavior Modeling

Inspired by ecology, researchers on artificial life proposed a series of guidelines on behavioral mechanism design. And they have been already properly applied to some group animation models. These principles are as follows [3], [25], [26].

1) Behavioral priority

Establish priority level for behaviors of agent. It should satisfy physical criteria visually as much as possible. Behaviors to avoid life-threatening danger should be superior to other acts.

2) Opportunity

The agent should have the ability to interrupt the current behavior for an act with higher behavioral priority.

3) React instantaneously

The agent should be able to react to stimulus instantaneously. That is to say, real-time simulation is required.

B. The Overall Framework of EM-AFSA

In this paper, the colony we mainly simulate is zooplankton. The mechanism we designed are as follows: avoiding static obstacles, avoiding predators, avoiding mutual collision, flocking, foraging and tailgating, which are arranged by priority from high to low.

The evasion mechanism we proposed reflects in the first three mechanisms. Avoiding static obstacles and avoiding predators make the zooplankton get away from potential threats. The third mechanism enable agents in group to avoid congestion, which also guarantees different agents not occupying the same location. Flocking imitates the habit of zooplanktons, which always gather together in colony. It is also the necessary condition that guarantees the agent moving to region with higher density of food during the process of foraging. Tailgating enables agents to approach neighbors with best condition. It is a supplementary condition that accelerates flocking. According to principles of behavior modeling and regulations zooplankton in natural world always obeying, we designed a complete group behavioral model with EM-AFSA. The algorithm flow is revealed in Fig. 1.

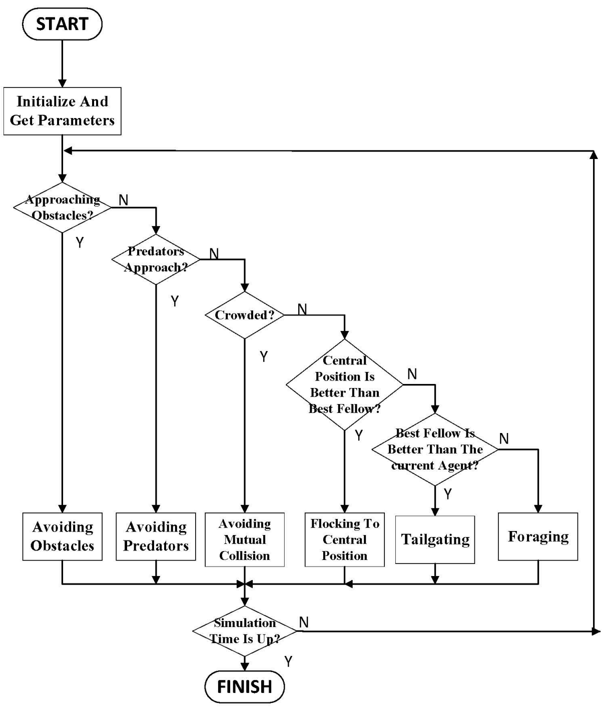

Fig. 1. Algorithm flow of animal's group behavioral model

C. Behavioral Mechanisms

We assume that in the three dimensional space there is a zooplankton colony with N individuals. The position of each individual is $\vec{P}_i(t)=(p_1,p_2,p_3), i=1,2,\cdots,N$, where p_1,p_2,p_3 represent the position coordinates of the individual. Individuals in the colony own following functional parameters: perception range V, step length S, number of times to perform foraging in current location $ForagNum$, number of times to perform tailgating in current location $TailNum$ and crowding distance C_{danger}. We must adjust these parameters according to different situations we set. We also should define range of motion $Range$ in Vrworld. After initializing all these parameters, the simulation process can start.

1) Avoiding static obstacles

a) Define $Flag_1 = 0$.

b) Search obstacles' position $\vec{P}_{obstacle}$ within V.

c) If

$$\vec{P}_{target} \in V \wedge \left\|\vec{P}_{target} - \vec{P}_i(t)\right\| < C_{danger} \tag{1}$$

where $\vec{P}_{target} = \vec{P}_{obstacle}$, the current agent steps back by random length in the opposite direction along agent and obstacle, i.e. to update the coordinate of current agent to a new one $\vec{P}_i(t+1)$ in next cycle according to equation :

$$\vec{P}_i(t+1) = \vec{P}_i(t) + \frac{R \bullet S \bullet (\vec{A}-\vec{B})}{\left\|\vec{A}-\vec{B}\right\|}, \tag{2}$$

where $\vec{A}=\vec{P}_i(t)$, $\vec{B}=\vec{P}_{obstacle}$ and $R\in(0,1]$ is a random positive number.

d) If (1), set $Flag_1 = 1$, which records that avoiding static obstacles happened.

e) Check whether

$$\vec{P}_i(t+1) \in Range, \tag{3}$$

is satisfied. If not, set values of border coordinate on relevant dimensions to components of $\vec{P}_i(t+1)$.

2) Avoiding predators

a) Define $Flag_2 = 0$.

b) Search predators' position $\vec{P}_{pre}$ within V.

c) If (1), where $\vec{P}_{target} = \vec{P}_{pre}$, update the coordinate of current agent to $\vec{P}_i(t+1)$ in next cycle according to equation (2), where $\vec{A}=\vec{P}_i(t)$ and $\vec{B}=\vec{P}_{pre}$.

d) If (1), set $Flag_2 = 1$, which records that the behavior avoiding predators happened.

e) Check whether (3) is satisfied and react correspondingly.

3) Avoiding mutual collision

a) Define $Flag_3 = 0$.

b) Search other agents' position within V , which is denoted as $\vec{P}_j(t), j = 1,2,\cdots N, j \neq i$.

c) If (1), where $\vec{P}_{target} = \vec{P}_j(t)$, update the coordinate of current agent to $\vec{P}_i(t+1)$ in next cycle according to equation (2), where $\vec{A} = \vec{P}_i(t)$ and $\vec{B} = \vec{P}_j(t)$.

d) If (1), set $Flag_3 = 1$, which records that the behavior avoiding mutual collision happened.

e) Check whether (3) is satisfied and react correspondingly.

By now, the evasion mechanism is accomplished. Note that the priority of flags is different, i.e. $Flag_1 > Flag_2 > Flag_3$. In simulation, if any of the flags equals to 1, update $\vec{P}_i(t)$ to $\vec{P}_i(t+1)$ according to corresponding cases.

4) Flocking

Before describing flocking, we must have a look at the objective function of the system. In the searching space, we defined an objective function to imitate the density of food resource distributed in the space.

Different kinds of objective function are tested. Normal distribution function or sampling function is usually employed. We denote $f(\vec{X})$ as the objective function of the system, which represents the food density in location $\vec{X}$.

a) Initialize counter and center position, which are denoted as $F\_num = 1$ and $\vec{P}_{center} = \vec{P}_i$.

b) Search other agents' position within V . If detected, update the counter by the equation: $F\_num = F\_num + 1$

and update the center position by the equation:

$$\vec{P}_{center} = \frac{\vec{P}_{center} + \vec{P}_j}{F\_num}, \vec{P}_j \in V, j \neq i . \tag{4}$$

e) After iterating through all the agent within V of $\vec{P}_i(t)$, we compute the value of objective function on $\vec{P}_i(t)$ and $\vec{P}_{center}$.

f) If $f(\vec{P}_{center}) > f(\vec{P}_i(t))$, update $\vec{P}_i(t)$ by equation (2), where $\vec{A} = \vec{P}_{center}$ and $\vec{B} = \vec{P}_i$.

g) Check whether (3) is satisfied and react correspondingly.

h) If $f(\vec{P}_{center}) < f(\vec{P}_i(t))$, the current agent executes foraging.

5) Foraging

a) Regard the current position as the center and search for a new position:

$$\vec{P}_i^k(\mathrm{t}) = \vec{P}_i(\mathrm{t}) + R_1 \bullet S \bullet V, k = 1,2,\cdots ForagNum , \tag{5}$$

where $R_1 \in [-1,1]$.

b) If $f(\vec{P}_i^k(t)) > f(\vec{P}_i(t))$ then the current agent moves towards this new position with a random length step, i.e. update $\vec{P}_i(t)$ by equation (2), where $\vec{A} = \vec{P}_i^k(t)$ and $\vec{B} = \vec{P}_i(t)$. Terminate the foraging.

c) If $f(\vec{P}_i^k) < f(\vec{P}_i(t))$, repeat a-b until the number of searching times reaches $ForagNum$. If there is no new position within V satisfies $f(\vec{P}_i^k) > f(\vec{P}_i(t))$ in the end, the current agent moves one step randomly:

$$\vec{P}_i(t+1) = \vec{P}_i(\mathrm{t}) + R_1 \bullet S , \tag{6}$$

and terminate foraging.

d) Check whether (3) is satisfied and react correspondingly.

6) Tailgating

a) Search for other agents within V and find out the agent with best condition and denote its position as $\vec{P}_{\max}(t)$, i.e.

$$\forall \vec{P}_m, f(\vec{P}_{\max}(t)) > f(\vec{P}_m(t)), \tag{7}$$

where $\vec{P}_{\max} \in V, \vec{P}_m \in V$ and $m \neq \max \neq i$. If there is no other agents within then set $f(\vec{P}_{\max}(t)) = 0$. If there are too many agents of the same kind in V , the maximum number of searching times is $TailNum$.

b) If $f(\vec{P}_{\max}(\mathrm{t})) > f(\vec{P}_i(\mathrm{t}))$ then current agent moves towards the position $\vec{P}_{\max}(t)$ by equation (2), where $\vec{A} = \vec{P}_{\max}(t)$ and $\vec{B} = \vec{P}_i(t)$.

c) Check whether (3) is satisfied and react correspondingly.

d) If not, the current agent executes foraging.

7) Predation

We also defined predator in our model. In fact, the evasion mechanism of predator is almost the same as zooplankton. The difference lies in the foraging mechanism. For a predator, it doesn't depend on objective function to execute behavior next step when foraging. Instead, it chases after the nearest target in its perception range until it succeeds.

We also design an extra mechanism, swallowing, to enrich the visual effect. If the zooplankton is within the swallowing range of predator, set the coordinate of this zooplankton the same as the predator. So the zooplankton is hidden in the predator and looks like it is swallowed.

IV. Simulation And Experimental Results

A. Experiment 1

The platform's size is grids. This grid is the default unit in Maya. In the Vrworld, we arranged three elements: zooplankton colony, obstacles, and a predator. Food resource is normally distributed in the space. In the region with highest density of food, we place a ring to put a mark. In the initial status, the zooplanktons are scattered randomly in a region far away from the food region. Between the above two regions, there is a region occupied by elliptical obstacles. The number of agents in the system is 120, whose perception ranges are all of 8 grids in semi diameter and step lengths are all 0.28 grid. *ForagNum* and *TailNum* are 5 and 10 times respectively. C_{danger} is 0.75 grid. The initial status is shown in Fig. 2.

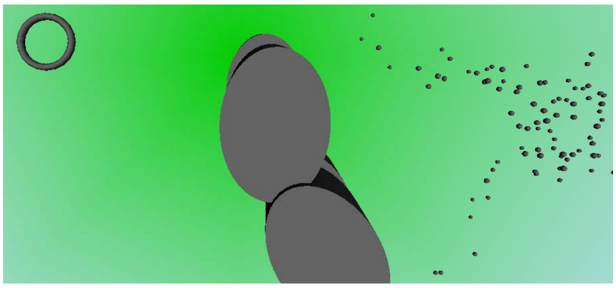

Fig. 2. The initial status of the virtual world

If the behavioral model we proposed does work, the zooplanktons should demonstrate corresponding dynamic group effect in the simulation. And if the simulation outcome is satisfied, it is feasible to transplant swarm intelligence algorithm to the field of group animation. In fact, the experimental results prove our scheme can perform well.

Fig. 3 and Fig. 4 show statuses after 1,000 and 1,800 cycle respectively. In Fig. 3, we can see that the agents flock to a colony from the scattered status. In Fig. 4, the agents bypass obstacles and move towards the region with more food in lines, which demonstrates evasion, foraging and tailgating mechanism at the same time.

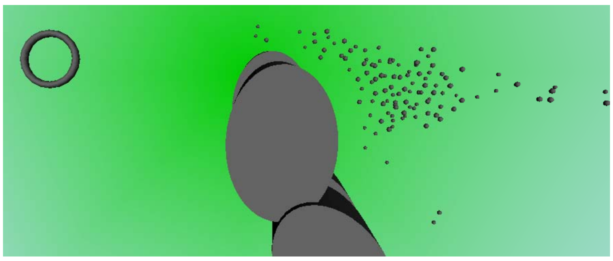

Fig. 3. The status of the virtual world after 1,000 cycle

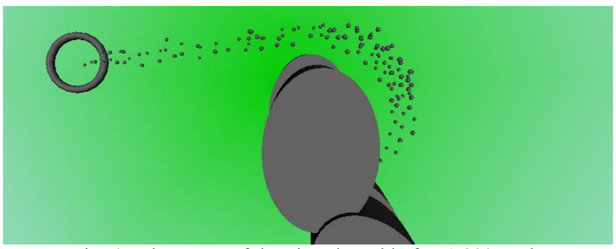

Fig. 4. The status of the virtual world after 1,800 cycle

B. Experiment 2

For the purpose of verifying the evasion mechanism we added, we design supplementary experiment to examine the outcome. In the basis of last experiment, we add another obstacle in the way the zooplankton colony used to choose. The result in Fig. 5 illustrates that the evasion mechanism works well. For comparison, we also demonstrate the visual effect of our model on condition that evasion mechanism is removed. The result is shown in Fig. 6. We can see that the members in this zooplankton colony flock together and move toward their target in lines straightly, passing through the space obstacles occupied. Obviously it looks unreal.

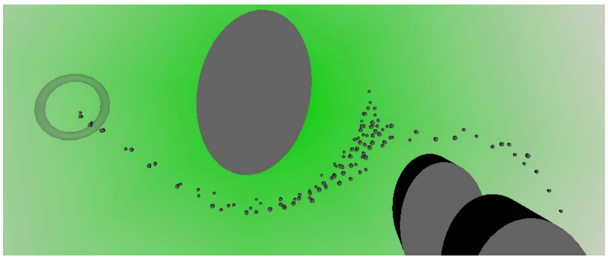

Fig. 5. The supplementary experiment on evasion mechanism

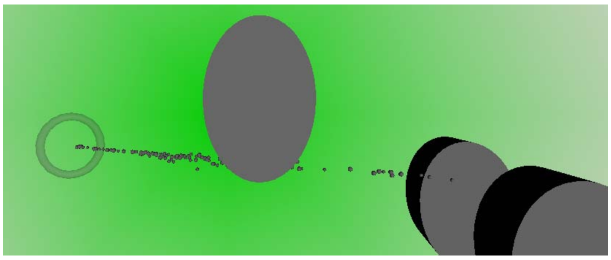

Fig. 6. The effect without evasion mechanism

C. Experiment 3

The evasion mechanism not only enables agents to avoid static obstacles, but also gives agents the ability to get away from moving threats. Related experimental results are shown in Fig. 7. In Fig.7, the zooplankton colony catches on predator and agents in the front of the colony chose to change their original planned path to evade dynamic danger. Then after the predator moving away, agents in the colony begin to recover their order as a group and keep on moving to food resource.

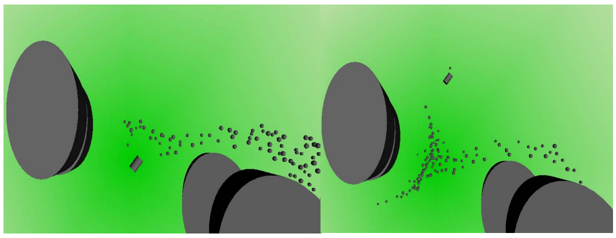

Fig. 7. Animal group evade predator during foraging

In this section, we demonstrate experimental results in simulation. The results show the behavioral model based on swarm intelligence algorithm we improved can display the dynamic effect of behavioral mechanism we designed in the model, which breeds the embryo of group animation designing.

V. CONCLUSION

In this paper, we improve a swarm intelligence algorithm by bringing in evasion mechanism to model animal's group behavior. In the simulation platform we build, we efficiently demonstrate the dynamic visual effect of this model. The mechanisms it includes, such as avoiding obstacles, avoiding mutual collision, getting away from predator, flocking, foraging and tailgating, all contribute to an encouraging display of group behavior. It is feasible to apply improved swarm intelligence algorithm to the designing of group animation.

In the future work, we will improve the simulation platform to create more visual realistic group animation and explore new mechanisms to optimize the model. Giving more intelligent characteristics to animal group such as learning ability is also under consideration. Besides, the model's capability to generate typical crowd phenomena [27] is still remaining to be tested.

REFERENCES

[1] E. H. Durfee and J. S. Rosenschein, "Distributed problem solving and multi-agent systems: Comparisons and examples," Ann Arbor 1001, 1994.

[2] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," Vol. 21. No. 4., 1987.

[3] X. Tu, "Artificial animals for computer animation: biomechanics, locomotion, perception, and behavior," No. 1635, Springer, 1999.

[4] Terzopoulos, Demetri, X. Tu and R. Grzeszczuk, "Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world," Artificial Life, Vol.1. No.4., pp. 327-351, 1994.

[5] S. Berman et al, "Bio-inspired group behaviors for the deployment of a swarm of robots to multiple destinations," in Robotics and Automation, 2007 IEEE International Conference on, 2007.

[6] L. Steels, "The artificial life roots of artificial intelligence," Artificial Life, Vol.1. No.1., pp. 75-110, 1993.

[7] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," Micro Machine and Human Science, 1995. MHS'95, Proceedings of the Sixth International Symposium on. IEEE, 1995.

[8] J. Kennedy, "Particle swarm optimization,"Encyclopedia of Machine Learning, Springer US, pp. 760-766,2010.

[9] M. Dorigo, Gianni Di Caro and Luca M. Gambardella, "Ant algorithms for discrete optimization," Artificial life, Vol.5. No. 2., pp. 137-172, 1999.

[10] M. Dorigo and Gianni Di Caro, "Ant colony optimization: a new meta-heuristic," in Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on. Vol. 2. IEEE, 1999.

[11] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Techn. Rep. TR06, Erciyes Univ. Press, Erciyes, 2005.

[12] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," Control Systems, IEEE, Vol.22. No. 3., pp. 52-67, 2002.

[13] S. D. Muller, et al, "Optimization based on bacterial chemotaxis," Evolutionary Computation, IEEE Transactions on, Vol.6. No.1., pp. 16-29, 2002.

[14] K. N. Krishnanand and D. Ghose, "Detection of multiple source locations using a glowworm metaphor with applications to collective robotics," in Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE, 2005.

[15] X. Li, "A new intelligent optimization method-artificial fish school algorithm," Hangzhou: Zhejiang University, 2003..

[16] X. Li and J.-x. QIAN, "Studies on Artificial Fish Swarm Optimization Algorithm based on Decomposition and Coordination Techniques," Journal of Circuits and Systems, Vol.1., pp. 1-6, 2003.

[17] E. Ju, "Morphable crowds," ACM Transactions on Graphics (TOG), Vol.29. No.6., p. 140, 2010.

[18] M. Lau, Ziv Bar-Joseph and James Kuffner, "Modeling spatial and temporal variation in motion data," ACM Transactions on Graphics(TOG). Vol. 28. No. 5. ACM, 2009.

[19] A. Lerner, Yiorgos Chrysanthou and Dani Lischinski, "Crowds by example," Computer Graphics Forum. Vol. 26. No. 3., 2007.

[20] N. Courty and Thomas Corpetti, "Crowd motion capture," Computer Animation and Virtual Worlds, Vol.18. No.4., pp. 361-370, 2007.

[21] M. Flagg and Jim Rehg, "Video-Based Crowd Synthesis," Visualization and Computer Graphics, IEEE Transactions on, Vol.19. No.11., pp. 1935-1947, 2012.

[22] S. Ali, K. Nishino and D. Manocha, "Modeling, Simulation, and Visual Analysis of Crowds," in The International Series in Video Computing, Springer US, 2014, pp. 335-357.

[23] Ondřej Jan, "A synthetic-vision based steering approach for crowd simulation," ACM Transactions on Graphics (TOG), Vol.29. No.4., p. 123, 2010.

[24] K. H. Lee, et al, "Group behavior from video: a data-driven approach to crowd simulation," in Proceedings of the 2007 ACM SIGGRAPH/Eurographics symposium on Computer animation. Eurographics Association, 2007.

[25] H. A. Simon, "A mechanism for social selection and successful altruism," Science(Washington), Vol.250. No.4988., pp. 1665-1668, 1990.

[26] P. Maes, "A bottom-up mechanism for behavior selection in an artificial creature," in Proceedings of the first international conference on simulation of adaptive behavior on from animals to animates, 1991.

[27] D. Helbing, "Self-organized pedestrian crowd dynamics: Experiments, simulations, and design solutions," Transportation science, Vol.39. No.1.,pp. 1-24, 2005.