Genetic Algorithms for Optimization of Boids Model

Yen-Wei Chen[1,2], Kanami Kobayashi[1], Xinyin Huang[3], and Zensho Nakao[4]

[1] School of Information Science and Eng., Ristumeikan Univ., Shiga 525-8577, Japan
[2] College of Elect. and Information Eng., Central South Forest Univ., Changsha 410004, China
[3] School of Education, Soochow University, Suzhou, Jiangsu 215006, China
[4] Faculy of Eng., Univ. of the Ryukyus, Okinawa 903-0213, Japan

Abstract. In this paper, we present an extended boids model for simulating the aggregate moving of fish schools in a complex environment. Three behavior rules are added to the extended boids model: following a feed; avoiding obstacle; avoiding enemy boids. The moving vector is a linear combination of every behavior rule vector, and the coefficients should be optimized. We also proposed a genetic algorithm to optimize the coefficients. Experimental results show that by using the GA-based optimization, the aggregate motions of fish schools become more realistic and similar to behaviors of real fish world.

1 Introduction

Simulating the aggregate moving of a fish school or a bird flock is an important issue in the areas of computer animation and artificial life. In 1986, Relond proposed a computer model of coordinated animal motion such as bird flocks and fish schools, which is called as boids [1]. The Boids model has three basic behavior rules, which are avoiding collision against neighbors; matching and coordinating own moves with neighbors; gathering together. The boids model has been used for modeling of fish [2]. In this paper, we present an extended boids model for simulating the aggregate moving of fish schools in a complex environment. Three behavior rules are added to the extended boids model: following a feed; avoiding obstacle; avoiding enemy boid. Each rule is represented by a vector. The direction and amplitude of the vector are adaptive to the environment. The moving vector of the boid (fish) is a linear combination of every behavior rule vector. As increasing the behavior rules, the setting of the coefficients becomes complex and difficult. We also proposed a genetic algorithm[3,4] to optimize the coefficients. Experimental results show that by using the GA-based optimization, the aggregate motions of fish schools become more realistic and similar to behaviors of real fish world.

The paper is organized as following: the extended boids model is presented in Sec.2, the genetic algorithm for optimization of coefficients is presented in Sec.3 and the experimental results are shown in Sec.4. Finally, the conclusion is given in Sec.5.

2 Extended Boids Model

The boids model is an example of an individual-based model. Each simulated boid (fish) is implemented as an independent actor that navigates according to its local perception of the dynamic environment. The global behavior of the school is

B. Gabrys, R.J. Howlett, and L.C. Jain (Eds.): KES 2006, Part II, LNAI 4252, pp. 55 – 62, 2006.

simulated by a large number of interacting individual boid (fish). In the extended boids model, each boid is an agent that follows following five behavior rules: avoiding collision against schoolmates; gathering together; following a feed; avoiding obstacle; avoiding enemy boids. The first two rules are Reynold's and the last three rules are our newly proposed ones.

2.1 Avoiding Collision Against Schoolmates

The first rule is avoiding collision against schoolmates. The rule is illustrated in Fig.1. The vector determined by the first rule is shown in Eq.(1).

$$\mathbf{V}_1 = \begin{cases} \left(\dfrac{|\mathbf{BoidVec}|}{fKeepDist} - 1 \right) \cdot \dfrac{\mathbf{BoidVec}}{|\mathbf{BoidVec}|} & \left(|\mathbf{BoidVec}| \le fVisibleDist \right) \\ 0 \quad \left(|\mathbf{BoidVec}| > fVisibleDist \right) \end{cases}, \tag{1}$$

where *fVisibleDist* is the visible distance of the boid (fish), *fKeepDist* is the safe distance for avoiding collision against schoolmates, and **BoidVec** is the vector from the boid to the nearest schoolmate. As shown in Eq.(1), when the distance to the nearest schoolmate is smaller than *fKeepDist*, a vector (force) is acted in opposite direction in order to keep away from the schoolmate. On the other hand, when the distance to the nearest schoolmate is larger than *fKeepDist*, a vector (force) is acted in the same direction in order to close to the schoolmate.

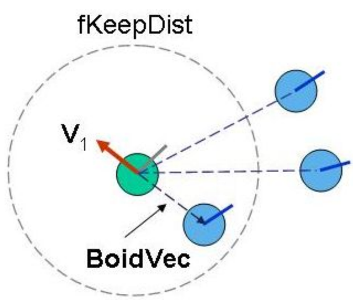


Fig. 1. Rule 1: avoiding collision against schoolmates

2.2 Gathering Together

The second rule is gathering together. A vector (force) is acted in the direction to the center (average position) of the neighborhood (fish school) in the view as shown in Fig.2. The vector is given by Eq.(2).

$$\mathbf{V}_2 = \begin{cases} \dfrac{\mathbf{CenterVec}}{|\mathbf{CenterVec}|} & \left(|\mathbf{CenterVec}| \le fVisibleDist \right) \\ 0 \quad \left(|\mathbf{CenterVec}| > fVisibleDist \right) \end{cases}, \tag{2}$$

where **CenterVec** is the vector from the boid to the center of the neighborhood.

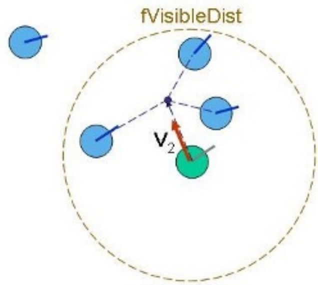


Fig. 2. Rule 2: gathering together

2.3 Following a Feed

The third rule is following a feed. A vector (force) is acted in the direction to the feed as shown in Fig.3. The vector is given by Eq.(3).

$$\mathbf{V}_3 = \frac{\mathbf{FoodVec}}{|\mathbf{FoodVec}|}, \tag{3}$$

where **FoodVec** is the vector from the boid to the feed.

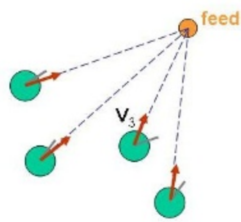


Fig. 3. Rule 3: following a feed

2.4 Avoiding Obstacles

The fourth rule is avoiding obstacles. Obstacle avoidance allowed the boids to fly through simulated environments while dodging static objects. The rule is illustrated in Fig.4. Assuming the avoiding angle be α and the size of obstacle be *ObsMag*.

$$\cos\alpha = \frac{\sqrt{|\mathbf{ObsVec}|^2 - \left(\dfrac{ObsMag}{2}\right)^2}}{|\mathbf{ObsVec}|}, \tag{4a}$$

where **ObsVec** is the vector from the boid to the center of obstacle as shown in Fig.4. The vector acted for avoiding obstacle is given as

$$\mathbf{V}_4 = \begin{cases} -\cos\theta \cdot \left(1 - \dfrac{|\mathbf{ObsVec}|}{fVisibleDist}\right) \cdot \dfrac{\mathbf{ObsVec}}{|\mathbf{ObsVec}|} & (\cos\theta \geq \cos\alpha) \\ 0 \qquad (\cos\theta < \cos\alpha) & \end{cases}, \tag{4b}$$

where θ is the angle of current direction of the boid with the obstacle.

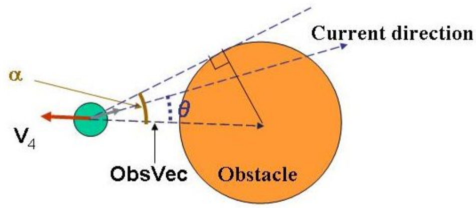


Fig. 4. Rule 4: Avoiding obstacles

2.5 Avoiding Enemy Boids

The fifth rule is avoiding enemy boid. When the boid finds an enemy boid in the visible distance, a vector (force) is acted in the opposite direction to the enemy boid as shown in Fig.5. The vector is given by

$$\mathbf{V}_5 = \begin{cases} \left(\dfrac{|\mathbf{OtherVec}|}{fVisibleDist} - 1\right) \cdot \dfrac{\mathbf{OtherVec}}{|\mathbf{OtherVec}|} & (|\mathbf{OtherVec}| \leq fVisibleDist) \\ 0 \qquad (|\mathbf{OtherVec}| > fVisibleDist) & \end{cases}, \tag{5}$$

where **OtherVec** is the vector from the boid to the enemy boid.

Fig. 5. Rule 5: Avoiding enemy boids

2.6 The Moving Vector

The moving vector of each boid is determined by above five rules. The moving vector can be considered as a linear combination of the five vectors as

$$\mathbf{V} = w_1\mathbf{V}_1 + w_2\mathbf{V}_2 + w_3\mathbf{V}_3 + w_4\mathbf{V}_4 + w_5\mathbf{V}_5 \tag{6}$$

where w_i is the coefficients used to balance the five rules and the coefficients should be optimized.

3 Genetic Algorithms for Optimization of the Moving Vector

As shown in Eq.(6), the moving vector of each boid is a linear combination of five vectors which are determined by each rule and the coefficients should be optimized. In this paper, we propose to use a genetic algorithm (GA) [3,4] for optimization of coefficients. GA applies the principles of evolution found in nature to the problem of finding an optimal solution. Since GA starts with a population of candidate solutions, it is easy to find a global optimum. In our previous works, we have applied the GA to image processing [5-7].

The flowchat of GA is shown in Fig.6. We use a real coding to represent chromosomes. The chromosome has five bits and each bit corresponds to w_1, w_2, w_3, w_4 and w_5, respectively. A roulette wheel selection is used as a selection operator. A two points crossover is used to generate two children from two selected parents. In the two points crossover, two points are randomly selected and everything between the two points is swapped between the parent organisms. We use Eq.(7) for mutation.

$$x' = x_l + \beta(x_u - x_j) \tag{7}$$

where x_u and x_l are upper limit and lower limit of the coefficients, respectively. β is a random value between 0 and 1. The chromosome and the bit for mutation are randomly selected.

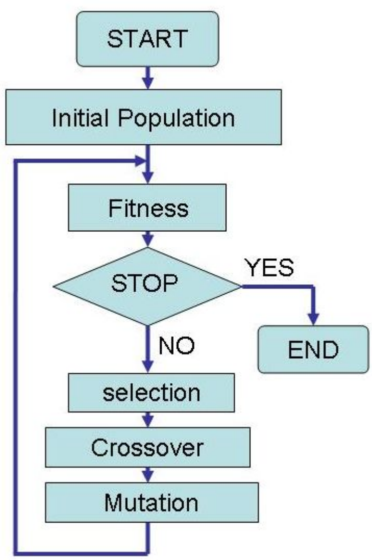


Fig. 6. Flowchat of GAs

The cost function and fitness function are shown in Eqs (8)-(14).

$$Cost1=\begin{cases}\dfrac{fVisibleDist\cdot\left(\left|\mathbf{BoidVec}\right|-fKeepDist\right)^2}{fKeepDist^2} & \left(\left|\mathbf{BoidVec}\right|\le fKeepDist\right)\\[2ex] \dfrac{fVisibleDist\cdot\left(\left|\mathbf{BoidVec}\right|-fKeepDist\right)^2}{\left(fVisibleDist-fKeepDist\right)^2} & \left(\left|\mathbf{BoidVec}\right|> fKeepDist\right)\end{cases}, \quad (8)$$

$$Cost2=\begin{cases}\dfrac{fVisibleDist\cdot\left(\left|\mathbf{FoodVec}\right|-\frac{1}{3}\cdot fVisibleDist\right)^2}{\left(\frac{1}{3}\cdot fVisibleDist\right)^2} & \left(\left|\mathbf{FoodVec}\right|\le\frac{1}{3}\cdot fVisibleDist\right)\\[2ex] 0 & \left(\frac{1}{3}\cdot fVisibleDist<\left|\mathbf{FoodVec}\right|<\frac{2}{5}\cdot fVisibleDist\right)\\[2ex] \dfrac{fVisibleDist\cdot\left(\left|\mathbf{FoodVec}\right|-\frac{2}{5}\cdot fVisibleDist\right)^2}{\left(\frac{2}{5}\cdot fVisibleDist\right)^2} & \left(\left|\mathbf{FoodVec}\right|\ge\frac{2}{5}\cdot fVisibleDist\right)\end{cases}, \quad (9)$$

$$Cost3=\left\{\dfrac{\left(\left|\mathbf{ObsVec}\right|-3\cdot fKeepDist\right)^2}{\left(\frac{ObsMag}{2}-3\cdot fKeepDist\right)^2},\quad \left(\left(\frac{ObsMag}{2}-fFishSize\right)<\left|\mathbf{ObsVec}\right|,\ \left|\mathbf{ObsVec}\right|\le 3\cdot fKeepDist\right)\right., \quad (10)$$

$$Cost4=\left\{\dfrac{\left(\left|\mathbf{OtherVec}\right|-2\cdot fKeepDist\right)^2}{2\cdot fKeepDist^2}\qquad \left(\left|\mathbf{OtherVec}\right|\le 2\cdot fKeepDist\right)\right., \quad (11)$$

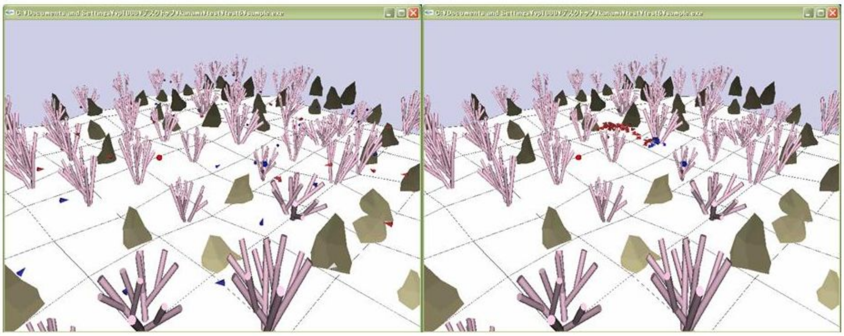
$$Cost5=\begin{cases}\dfrac{2\cdot\left(fDirVecLen-1.5\cdot UniqueSpeed\right)^2}{fVisibleDist} & \left(1.5\cdot UniqueSpeed<fDirVecLen,\ fDirVecLen\le 10\cdot UniqueSpeed\right)\\[2ex] 0 & \left(0.5\cdot fSchoolSpeed\le fDirVecLen\le 1.5\cdot fSchoolSpeed\right)\\[2ex] \dfrac{fVisibleDist\cdot\left|fDirVecLen-0.5\cdot UniqeSpeed\right|}{0.5\cdot UniqueSpeed} & \left(fDirVecLen<0.5\cdot UniqueSpeed\right)\end{cases}, \quad (12)$$

$$Cost=Cost1+Cost2+Cost3+Cost4+Cost5\,. \quad (13)$$

$$Fitness=\frac{1}{1+Cost}\,. \quad (14)$$

4 Experimental Results

We have made an interactive fish school system [8] based on the extended boids model and the system is made by Open GL [9]. The examples of the system are shown in Fig.7. Two fish schools are simulated in the system. Figure 7(a) is a result without GA-based optimization and Fig.7(b) is a result with GA-based optimization (after 100 generations). It can be seen that by using the GA-based optimization, the aggregate motions of fish schools become more realistic and similar to behaviors of real fish world.



(a) (b)

Fig. 7. Experimental results. (a) without GA, (b) with GA.

5 Conclusions

In this paper, we proposed an extended boids model for simulating the aggregate moving of fish schools in a complex environment. Three behavior rules were added to the extended boids model: following a feed; avoiding obstacle; avoiding enemy boids. We also proposed a genetic algorithm to optimize the coefficients. Experimental results showed that by using the GA-based optimization, the aggregate motions of fish schools become more realistic and similar to behaviors of real fish world.

This work was supported in part by the "Open Research Center" Project for Private Universities: matching fund subsidy from MEXT (Ministry of Education, Culture, Sports, Science, and Technology).

References

1. Reynolds, C. W.: Flocks, Herds, and Schools: A Distributed Behavioral Model. Computer Graphics, 21(1987) 25-34.
2. DeAngelis, D. L., Shuter, B. J., Ridgeway, M. S., Blanchfield, P., Friesen, T., and Morgan, G. E.: Modeling early life-history stages of smallmouth bass in Ontario lakes. Transaction of the American Fisheries Society, (1991) 9-11.
3. Goldberg, D.: *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, (1989).
4. Forrest, S.: Genetic algorithms: principles of natural selection applied to computation. Science, 261(1993) 872-878.
5. Chen, Y. W., Nakao, Z., Arakaki, K., Fang, X., and Tamura, S.: Restoration of Gray Images Based on a Genetic Algorithm with Laplacian Constraint. Fuzzy Sets and Systems, 103 (1999)285-293.

6. Chen, Y. W., Nakao, Z., Arakaki, K., and Tamura, S.: Blind Deconvolution Based on Genetic Algorithms. IEICE Trans. Fundamentals, E-80-A (1997)2603-2607.
7. Mendoza, N., Chen, Y. W., Nakao, Z., and Adachi, T.: A hybrid optimization method using real-coded multi-parent EA, simplex and simulated annealing with applications in the resolution of overlapped signals. Applied Soft Computing, 1(2001)225-235.
8. Kobayashi, K.: Interactive fish school generation system using GA. Graduation thesis of Ritsumeikan Univ., (2005).
9. Shreiner, D.: Open GL Reference Manual: The Official Reference Document to Open Gl, Version 1.4. Addison-Wesley, (2004).