Volume: 3 Issues: 8 [June, 2018] pp. 51-56]
Journal of Information System and Technology Management
eISSN: 0128-6666
Journal website: www.jistm.com

UNDERSTANDING STEERING MECHANISMS FOR SIMULATING FISH BEHAVIOURS

Chen Kim Lim[1]
Kian Lam Tan[2]

[1,2]Faculty of Art, Computing & Creative Industry, Sultan Idris Education University, 35900 Tanjong Malim, Perak, Malaysia
Email: kim@fskik.upsi.edu.my

Accepted date: 3 May 2018 **Published date**: 2 July 2018

To cite this document:

Lim, C. K., & Tan, K. L. (2018). Understanding Steering Mechanisms for Simulating Fish Behaviours. *Journal of Information System and Technology Management*, *3*(8), 51–56.

Abstract: *ICT transformation modernized the global solutions with the aid of technology. In the computer science industry, procedural simulation is one of the ways to illustrate visual representation of a given scenario through mathematical data and algorithms. This paper proposed a framework to simulate school of fishes of heterogeneous behaviours using the modified classical boid algorithm. Before the simulation of school of fishes, it is vital to understand the steering mechanisms. The keystone of this paper is to allow flexible adaptations by two-level approaches where autonomous agent can instantly changes its micro and macro behaviours through the parameter alternation. This model is found to be able to support the general hypothesis about the behaviours pattern when they live individually and, in a group, observed in the fishes.*

Keywords: *Heterogeneous Behaviours, Boid Algorithm, Autonomous Agents, Steering Force.*

Introduction

Learning basic knowledge of animals' behaviours is vital in shaping imagination of young learners about the ecosystem. However, they might not have the experience to see the animals physically by themselves. This simulation framework is to help to educate learners to cultivate love for sea creatures and understand how these aquatic animals are living in groups. Besides, young learners are usually very enthusiastic in learning about animals' behaviours. However, gaining the knowledge from the experience of learning theories that are taught by our educators might not be adequately interesting.

By having interactive options of other animal representations, young learners can explore a variety of species experience of marine creatures that performs the common behaviours. Nonetheless, animal behaviours can be sometimes unpredictable and unexpected. A careful analysis of behavioural patterns and sequences allow us to identify all the relevant components and link to wider context of physical and biological environment of the animals. Through this proposed simulation framework, an illustration of changes of animal behaviours in real-time is developed in order to speculate upon its function and the factor which control the changes of the behaviours.

In order to add the heterogeneity of the animal behaviours, the framework is designed in such a way where it allows flexible adaptations by two-level approaches: autonomous agent that changes its (a) micro-behaviours (caused by changes of individual steering forces) and (b) macro-behaviours (caused by changes of steering behaviours of other animals) when the parameters are altered in real-time. At the end of the simulation, the tested hypothesis will be carried out to validate the changes of animals' micro and macro behaviours can be influenced by local interactions (obstacle avoidance among their other animals) and global interactions (obstacle avoidance with the environment) among themselves respectively.

Previous and Related Works

Recently, [1] improvised further improvement to the boid algorithm. There are other factors that influenced the general direction of the flock such as influence of strong wind or current and desire or tendency of a sparse flock towards a particular place made the animal behaviours even more unpredictable. In order to diverge the flock, [2] has lately introduced the n-wavefront algorithm so that the birds will not fly into a forbidden area where the aircraft lands. However, coordination problem in these two researches is ubiquitous when each agent is having their own personal goals. To overcome this problem, [3] proposed to develop distributed learning techniques that help in future behaviour by learning through previous successful experiences on case-to-case basis. In order to create a believable and compelling interaction between animals, the planning of movements is equally important even though animal behaviours is indeed less complex as compared to human behaviours. [4] suggested to integrate information about animal relationships and contrainsts into both single and multi-agent search that can be layered to select following and fleeing behaviours. Thus, collecting right information and setting the appropriate rules for the visualisation of the animal behaviours are crucial. With these, [5] has proposed to construct a simulation model that is suitable for multi-agent behaviour learning from the observation of original behaviours and resulted that the learned behaviour produced aggregate behaviour that is very similar to the synthetic foraging simulated behaviour.

Mixing more behaviours would permit in creating more complex steering behaviours. One of the complex interactions either among animals or humans would be a vigorous quarrel. [6] proposed a two-level hierarchical state machine (hFSM) for disclosing a quarrel. Small groups consisting of only three agents are kept in a triangular formation and their parameters of steering behaviours are changed. Besides, the use of state machine to decide which actions are taken, the concept of cellular automata is also extended. [7] proposed to simulate thousands of different cells that cater different and huge terrestrial animal groups in a living system. Therefore, the boids can then behaved based on the neighbourhood grid and brute force algorithm. Another complex animal interaction would be a hunting activity. As ants hunt in a more organised and concise manner, ant-hunting behaviour could be a suitable and testable model. [8] proposed to validate and compare quantitatively between the live and simulated ants to understand their hunting behaviours by simulating two scenarios called subdued and successful delivery.

[9] proposed to simulate agent egress in virtual environment. Not only the human will evacuate during the chaotic circumstances, the animals will also have cases where they shifted to new nest or barn. This research proposed to have grouping restrictions to stay or live with the families and other groups that travel together. [10] proposed to accommodate high-level socially-aware behavioural realism by a coordination model where the particles are split up and later reunite to accomplished a joint goal. These social attachments are also actually applicable to the animal taxonomy. Thus, there can be more possible conducts and constraints in boid algorithm depending on the situation's needs and applications.

Implementation Framework

Figure 1 shows the simulation modules of the simulated underwater world animals. There are six modules namely the 1) animal systems functionalities, 2) animal movement dynamics, 3) internal state repository, 4) modeller interface, 5) image generator interface and 6) the evaluation. However, at this stage, the particles are only replaced with the fishes.

In the animal systems functionalities module, five different steering forces are implemented. The steering forces of containment, cohesion, alignment and separation are each over 25% as the maximum parameters that can be adjusted while the obstacle avoidance can be adjusted to maximum of a 100%. Animal movement dynamics module controls the vision distance, and angle as well as the maximum speed. The internal state repository module consists of two types of flock controls: 1) Classic control that executes only one type of action per state and 2) Direct control that executes combinations of multiple forces per state.

In the modeller interface module, there are inspector and controller that are meant for parameter alterations. Simple graphical user interface is designed such as radio buttons for flock controlling and types of animals, slider for adjusting steering forces, checkbox button for displaying vision and textbox button for add or remove an animal.

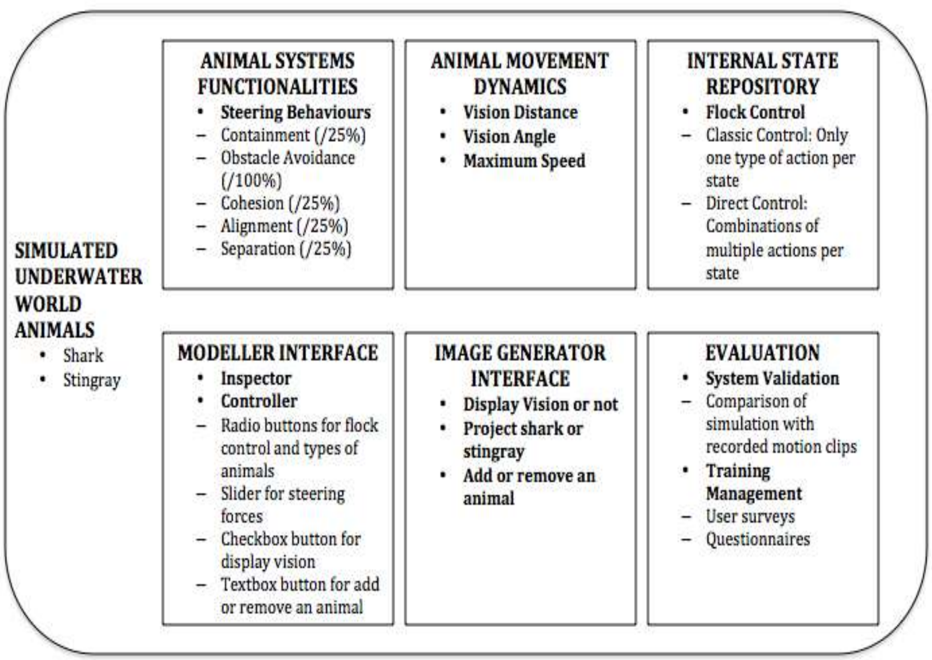

Figure 1. Simulation Modules

The image generator interface is the back-end to improve the animal representations to portray the animals living individually or in a group. Therefore, three functions where embedded in this module where a user can choose to display vision or not, projection of other aquatic animals such as shark and stingray, and add or remove an animal so that they can appear and disappear within the predefined boundary.
Finally, the evaluation module is to validate the system and to manage training for learners who are interested in ethology. The system validation is carried out by comparing the simulation with the recorded motion clip. Through the comparison, the individual and group behaviours among the animals are displayed. During the training management process, user surveys and questionnaires are carried out for further refinement of the simulator.

Results and Discussion

The aim of this research is in allowing flexible adaptations by two-level approaches where autonomous agents can instantly change its micro behaviour through local interactions and macro behaviours through global interactions. Both local and global interactions are manipulated by a parameter alteration controller. Detailed scenario-based visual observations are needed to validate the system by each force that changes its behaviours. Each fish is represented by particles at the early stage of the simulation. The alignment, cohesion, separation, containment and obstacle avoidance are implemented based on classical boid algorithm. The path-followingbehaviour is also seen when some particles are heading towards to a pursuit location. To give a more realistic simulation of the behaviours of the particles, the speed clamping is also performed.

A. Seeking, Fleeing, and Arriving Behaviour

In order to compute a new velocity, the desired velocity for each of these behaviours is determined. All the desired velocities can be computed through the position of the target and position of the particle. The desired velocity (VD) is computed as follows:

$$VD = \frac{P-Pt}{||P-Pt||} * \text{current Target Speed} \quad (1)$$

where is $\frac{P-Pt}{||P-Pt||}$ the unit vector that gives the desired velocity, P is the position of the virtual agent and Pt is the targeted position and current Target Speed is the speed the virtual agents wants to reach. The Steering Force [11] is then calculated as follows:

SteeringForce = DesiredVelocity – Current Velocity (2)

The algorithm for seeking behaviour is as shown in Figure 2 and Figure 3 shows the illustration of the computation of the steering force:

SEEK	Desired force through a vector pointing from the location to the target
NORMALISE	The desired velocity and scale to the maximum speed
COMPUTE	Steering force = desired velocity – current velocity
LIMIT	To maximum steering force

Figure 2 Algorithm for Seeking Behaviour [11]

For the fleeing behaviour, it is computed based on the above algorithm except that the subtraction is based on the position of the virtual agent with the position of the target as illustrated in Figure 4. As for the arriving behaviour, after the computation of the desired velocity and provided that the distance is inferior to the arrival distance (Distance from which the virtual agent starts to slow down), the steering force is computed through:

$$\text{SteeringForce} = \text{DesiredVelocity} * \frac{(\text{target} - \text{position})}{\text{ArrivalDistance}} - \text{CurrentVelocity} \qquad (3)$$

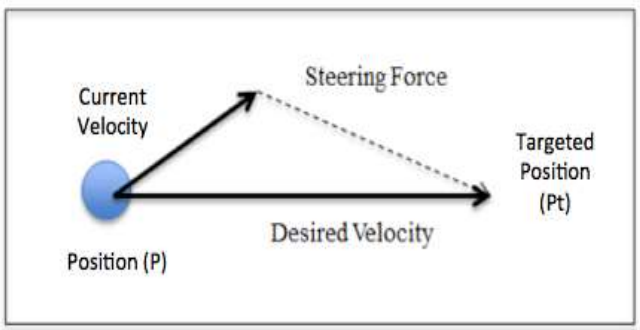

Figure 3. Illustration of the Computation of Steering Force

This will slow down the virtual agent as the virtual agent approaches the target as illustrated in Figure 5 because the factor will always be between 0 and 1.

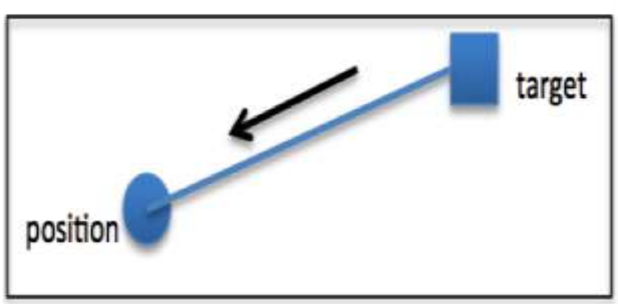

Figure 4. Illustration of the Fleeing Force

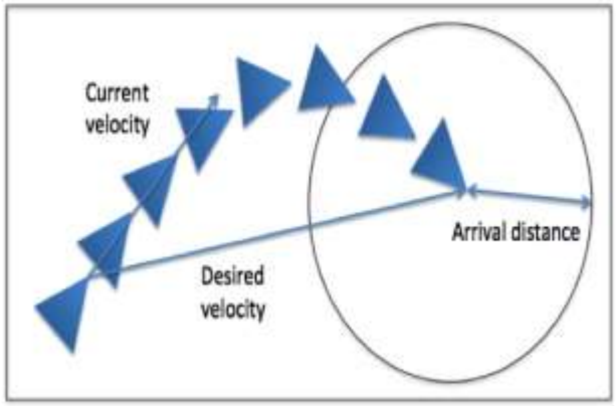

Figure 5. Illustration of the Arrival Force [11]

B. Wandering Behaviour

The wandering behaviour is more of an individual behaviour which is moving in one direction for a little while, then wandering off to the next for a little bit and so forth. Basically, three parameters which are the wander distance, the wander radius and the wander angle are needed to be pre-determined to make the wandering behaviour happen as shown in Figure 6. The wander angle is the displacement and changes from each time step. In Figure 6, the steering direction is represented by a vector in red, the big circle constraints the steering and the small circle constraints the random offset of the steering. The small circle determines how much the virtual agent will steer. Only small displacement will happen and the forward direction of the virtual agent is maintained so that it will not suddenly move from left to right very drastically due to big changes of angles.

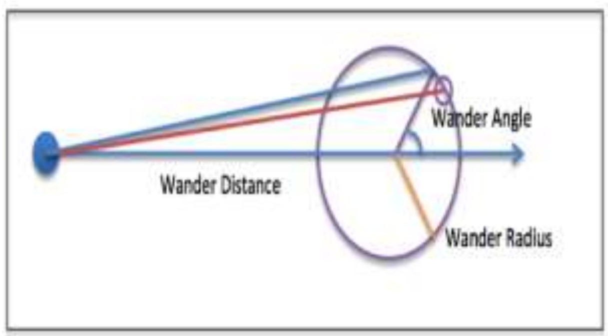

Figure 6. Illustration of Wander [11]

C. Simulation Run

In the simulation, the competitive and direct alternate method has been tested and finally the traditional Monte Carlo [12] method. With the proposed higher-level controller scheme, the parameter adaptions for each of the fishes is automated in real-time. In order to apply the Boid algorithm in the animal simulation, there is a need to change a few standard rules to match the changing goals. In general, Figure 7 shows the final simulations of the fishes that can be changed in real-time through instant parameter alteration controller.

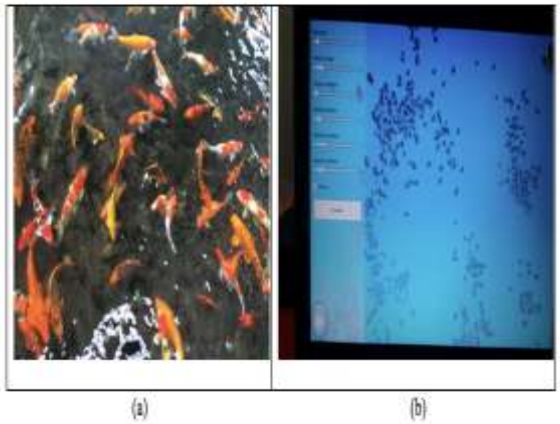

Figure 7. (a) Real Clip of Koi Fishes Taken from Bukit Jambul Country Club, Penang (b) Simulation of School of Koi Fishes

Table 1 shows the ranges of parameters used for simulating the individual and group behaviours respectively. By using the classical control of an arbituary value of 100 as the maximum force, the behaviours are simulated where only one type of action is executed per frame. By altering the parameters during the simulation runs, the graphics designers can achieve the ideal behaviours of aquatic animals in displaying the individualised and group behaviours when teaching ethology.

Table 1. Ranges of Parameters Used for Simulating Individual and Group Behaviours.

Types of Behaviours	Individual Behaviours	Group Behaviours
Containment Steering Forces	17-18 (10%)	25-26 (12%)
Obstacle Avoidance Steering Forces	64-66 (40%)	97-100 (48%)
Cohesion Steering Forces	0 (0%)	28-30 (14%)
Alignment Steering Forces	17-18 (10%)	25-26 (12%)
Separation Steering Forces	64-66 (40%)	25-26 (12%)

Conclusion

As a conclusion, the animals simulated is able to display individual as well as group behaviours. Through the boid simulator, the objective of having agent-agent and agent-environment interactions are achieved. The future direction would be to merge this physically-based simulation with the learning theory of ethology as a pedagogic tool to educate the young learners about the fish behaviours.

Acknowledgment

This research is fully supported by the University GPU Grant (No. 2015-0121-109-01). The authors would like to acknowledge the many helpful suggestions of the anonymous reviewers.

References

[1] M. Sajwan, D. Gosain, and S. Surani, “Flocking Behaviour Simulation: Explanation and Enhancement In Boid Algorithm,” *International Journal Of Computer Science And Information Technology (IJCSIT)*, vol. 5, no. 4, pp.5539-5544, 2014.

[2] S. Gade, A.A. Paranjape, and S.-J. Chung, “Herding a flock Of bird approaching an airport using an unmanned aerial vehicle,” *AIAA Guidance, Navigation, And Control Conference (SciTech)*, pp.1-17, 2015.

[3] A. Garland, and R. Alterman, “Autonomous agents that learn to better coordinate, autonomous agents and multi-agent systems,” B*randeis university Technical Report CS-TR-03-237*, pp.1-39, 2003.

[4] N.R. Sturtevant, “Moving path planning forward,” M. Kallmann and K. Bekris (Eds.): *Motion in Games (MIG2012*, pp.1-6, 2012.

[5] B. Hrolenok, and T. Balch, “Learning executable models of multiagent behaviour from live animal observation, “ *In ICML 2013 Workshop on Machine Learning For System Identification, 2013.*

[6] C. Brom, P. Babor, M. Popelova, M. Bida, J. Tomek, and J. Gemrot, “Controlling three agents in a quarrel: lessons learnt,” M. Kallmann and K. Bekris (Eds.): *Motion in Games (MIG2012)*, pp.158-169, 2012.

[7] M. Joselli, E.B. Passos, J.R. Silva, M. Zamith, and E. Clua, “A Ffocking boids simulation and optimization structure for mobile multicore architectures,” *Proceedings Of SBGames, pp.* 83-92, 2012.

[8] Y.-T. Yang, A. Quitmeyer, B. Hrolenok, H., Shang, D.B. Nguyen, T.R. Balch, T. Medina, C. Sherer, and M. Hybinette, “Ant hunt: towards a validated model of live ant hunting behavior,” *In FLAIRS Conference*, pp.110-115, 2012.

[9] S. Rodriguez, A. Giese, N.M. Anato, S. Zarrinmehr, F. Al-Douri, and M.J. Clayton, “Environmental effect on egress simulation,” M. Kallmann and K. Bekris (Eds.): *Motion in Games (MIG2012*), pp. 7-18, 2012.

[10] S.I. Park, C. Peng, F. Quek, and Y. Cao, “A crowd modeling framework for socially plausible animation behaviour,” M. Kallmann and K. Bekris (Eds.): *Motion in Games (MIG2012, pp.* 146-157, 2012.

[11] D. Shiffman, “The nature of code: simulating natural systems with processing, Edition 1, 2012.

[12] Z. Cui, and Z. Shi, “Boid particle swarm optimization,” *International Journal of Innovative Computing and Applications*, vol. 2, no. 10, pp.77-85, 2009.