ICIC Express Letters
Part B: Applications
Volume **7**, Number **10**, October **2016**

ICIC International ©2016 ISSN 2185-2766
pp. 2177–2182

FAST SIMULATION OF FISH SCHOOLING UNDER REALISTIC SEA WATER

Dexing Wang, Yu Ge, Mengzhao Yang*, Hongchun Yuan
Yang Yang, Ting Ning and Jia Cai

College of Information
Shanghai Ocean University
No. 999, Huchenghuan Rd., Nanhui New City, Shanghai 201306, P. R. China
*Corresponding author: myang@shou.edu.cn

Received April 2016; accepted July 2016

Abstract. *Fish schooling is an important area of research with relevant developments on biomechanics, game animation, robotics and mathematical modeling. In order to simulate swimming behavior of reality, we present a novel simulation method of fish swimming under realistic sea water. In this paper, using spring-mass system, we simulate fish muscles locomotion and acquire behavioral animation featuring artificial fishes, which can yield realistic individual and collective motions. Also, in order to enhance the realism of fish schooling under sea water, we give a GPU-based (Graphic Processing Unit) realistic rendering of sea water via simulation of optical effects. The result shows that we could acquire realistic schooling behavior under sea water and obtain a real-time rendering speed.*

Keywords: Fish schooling, Sea water, Spring-mass system, Graphic processing unit, Real-time rendering

1. Introduction.

Fish schooling behaviors are some of the most prominent social and group activities exhibited by fishes. Fish may school for many reasons, including foraging, preying and defense from predators. Studying fish schooling enables researchers to design vehicles and robots that can skillfully navigate aquatic environments under sea water [1]. In order to perform such studies, we have to require design of fish locomotion and interaction with sea water environments.

Early, in fish schooling simulations, schooling phenomenon can be well simulated by some ecological models [2,3], and these models for coexistence of prey and predator have been researched effectively. In order to give fast schooling for fish, NaturalMotion [4] has to fully understand the biomechanics of a real creature and implement a fast rendering. But it tends to lack the uniqueness and personality of manual animation. Its use on the big screen is usually restricted to background elements. Then, procedural animation of fish [5] is presented to overcome its scalability limitations in [4], and create fish swimming animation entirely procedurally. The animation is quite realistic, however the fish is only able to swim sinusoidally forward and the rendering speed is not high. The motion of the fish in many researches is assigned and swimming behaviors are not flexible. In recent years, many efforts are put on fluid dynamics [6] and fish locomotion is simulated more flexibly. Nardinocchi and Teresi [7] simulate muscles contraction by using the notion of distortions, and emphasize the kinematical role of muscle and the generation of movement. Nowadays, with the increasing requirements of real-time computation, the main challenge of simulation tends to develop efficient method to fit real-time animation and GPU realization [8]. Considering an individual-based model for fish schooling, Li et al. [9] describe GPU implementation and present computational experiments illustrating the power of this technology for fish schooling. Exploring the multi-core architecture, Borges et al. [10] show their experience developing a hybrid MPI+OpenMP version of the parallel and distributed individual-oriented fish schooling simulator. On the other hand, by developing robot fish to mimic the style of fish swimming [11,12], underwater vehicles with high efficiency can be also designed and applied to autonomous control domains. However, when fast requirements for real-time performance such as 3D Game and interactive animation are focused on, simulation speed of most methods above is still relatively low, and the key optical and motion characteristics of sea water have also not been fully considered.

Our focus differs in that we make extensive use of GPU rendering pipeline to meet the fast animation requirement and realize key optical effects of sea water. In this paper, we simulate behavioral animation featuring artificial fish model with spring-mass system. These behaviors yield realistic individual and collective motions with minimal intervention from the animator. Also, we give a realistic rendering of sea water via simulation of optical effects, and enhance the realism of fish schooling under sea water. Our method is based on GPU implementation and acquires a higher rendering speed.

The rest of the paper is organized as follows. Simulation of fish locomotion via spring-mass system is given in Section 2. Then optical effects of sea water are shown in Section 3. At last, experimental results are discussed in Section 4, and finally conclusions are given in Section 5.

2. Simulation of Fish Locomotion via Spring-Mass System.

The effective method is to simulate fish motion behavior as a network of masses and springs (called spring-mass system), and then attach muscle controllers to the model that will generate the forces necessary for locomotion in sea water. Muscles are modeled as springs with variable rest lengths by decreasing and increasing the rest lengths of a muscle, and the muscle will contract and relax accordingly.

The basic vibration model of spring-mass system consists of a mass, a massless spring, and a damper, which can be shown in Figure 1.

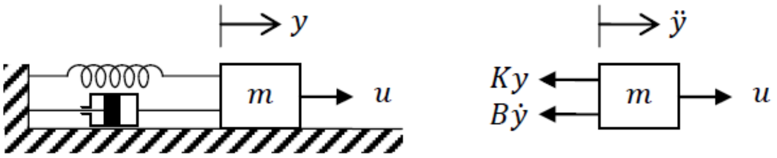

FIGURE 1. Spring-mass system

As shown in Figure 1, m is the mass in kg. y is the position in m. $\dot{y}$ and $\ddot{y}$ are first-order derivative and second-order derivative respectively. u is the control input in N. K is the linear spring constant in N/m. B is the damping coefficient in Ns/m.

In this paper, the spring-mass system can be redefined by set of nodes N and springs S. Let each node have a mass m and position p. Let each spring have a stiffness K_s, and have its rest length l_0 defined as the vector between two nodes. Such a system is inherently stable until some external forces are applied, since all springs remain at their rest lengths. The forces applied to each node by the springs connected to it are:

$$F_i^s = f^s(p_i, p_j) = K_s \frac{p_j - p_i}{|p_j - p_i|}(|p_j - p_i| - l_0) \tag{1}$$

$$F_j^s = f^s(p_j, p_i) = -f^s(p_i, p_j) = -F_i^s \tag{2}$$

These forces conserve momentum $\left(F_i^s + F_j^s = 0\right)$ and are proportional to the elongation of the spring $(|p_j - p_i| - l_0)$. Alternatively, one can make them proportional to the *relative* elongation by replacing the constant K_s with the constant K_s/l_0. The damping forces are:

$$F_i^d = f^d(p_i, v_i, p_j, v_j) = K_d(v_j - v_i) \cdot \frac{p_j - p_i}{|p_j - p_i|} \tag{3}$$

$$F_j^d = f^d(p_j, v_j, p_i, v_i) = -f^d(p_i, v_i, p_j, v_j) = -F_i^d \tag{4}$$

where v is the velocity on the one point. K_d is the damping coefficient.

With the stable fish structure being defined, the system finds equilibrium at its original position. Muscle contractions are defined by the muscle springs in the procedurally generated spring-mass system. Muscles form two groups, one for each side of the fish, and these are actuated as a group by the system.

The damping forces are proportional to the velocity difference projected onto the spring and are momentum conserving as well. Let us combine the two forces into one unified spring force and the final solution for the system is:

$$F(p_i, v_i, p_j, v_j) = F_i^s + F_i^d = f^s(p_i, p_j) + f^d(p_i, v_i, p_j, v_j) \tag{5}$$

In order to simulate fish swimming behaviors, an external force must act on the fish. This force is generated when the fish pushes against the water and the water pushes back. The force generated is the normal opposite to the body and proportional to the volume of the water displaced. Here we approximate this force by the spring-mass system related above to obtain realistic simulation of fish locomotion.

3. **Optical Effects of Sea Water.**

Properly rendering the sea water is crucial for achieving a plausible animation. Reflection and refraction are two major components that contribute to the realism of sea water.

We consider mirror *reflection* when rendering the water. More precisely, the light from a distant point source in the direction of $\overrightarrow{s}$ is reflected into a range of directions about the perfect mirror directions.

$$\overrightarrow{m} = 2\left(\overrightarrow{n} \cdot \overrightarrow{s}\right) \times \overrightarrow{n} - \overrightarrow{s} \tag{6}$$

where $\overrightarrow{n}$ is the normal on the incoming point. $\overrightarrow{s}$ is the incoming direction from light source. $\overrightarrow{n} \cdot \overrightarrow{s}$ term is called the foreshortening term.

One common model we used for this is the following:

$$C_{reflection}\left(\overrightarrow{d_e}\right) = r_s \times I \times \max\left(0, \overrightarrow{m} \cdot \overrightarrow{d_e}\right)^\alpha \tag{7}$$

where r_s is called the specular reflection coefficient (often equal to $1-r_d$), I is the incident power from the point source, and $\alpha \geq 0$ is a constant that determines the width of the specular highlights. As α increases, the effective width of the specular reflection decreases. In the limit as α increases, this becomes a mirror.

As light passes from air to sea water, its speed slows down and the light rays are bent. This is called *refraction* as shown in Figure 2, which affects the appearance of fish

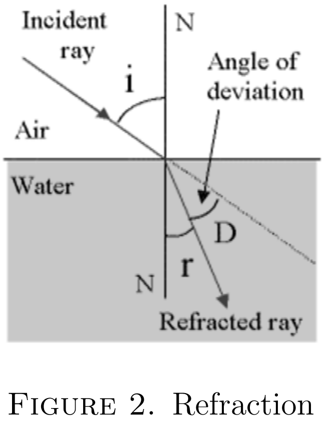

FIGURE 2. Refraction

schooling you see under water. In Figure 2, i is angle of incidence. r is angle of refraction. N is the normal line.

According to Snell's law, it simply relates angles i and r to the refraction indices of the two medias $n1$ and $n2$. The ratio of the sines of incidence and refraction angles will be equal to the inverted ratio of the indices of refraction as follows.

$$n1 \times \sin(i) = n2 \times \sin(r) \tag{8}$$

4. **Results.** We have implemented simulation of fish schooling under realistic sea water and give a fast animation in real time. Our final rendering of fish schooling clearly shows the effective implementation of the proposed technique. With AMD Athlon II X4 Four Cores and NVIDIA GeForce GT430 @ 2G, we have achieved frame rates of approximately 48 frames per second (FPS) when rendering 30 fishes once, which is a relatively high speed.

We need simulate fish swimming behaviors by solving the schooling-style for short time intervals, and re-meshing to track the long swimming path we aim at simulating. By using solution of spring-mass system (4), we obtain a carangiform schooling-style to simulate fish swimming behavior as shown in Figure 3.

From Figure 3, sequence of fish swimming shapes as consequence of muscles activation is effectively simulated via spring-mass system. Color map denotes two different muscle statuses including contracted and elongated. This undulatory motion gives the thrust required to swim rightward. So we can give lifelike behaviors when fish is swimming.

Then in order to simulate fish schooling and emphasize the optical effects under sea water, we implement a real-time animation of thirty fishes schooling under reflection and refraction of light, and acquire realistic swimming effects as shown in Figure 4.

From Figure 4, as a result of considering optical effects that appear in the sea water, we could acquire more realistic surroundings of fish schooling under different illuminations, which is naturally integrated into the sea water and greatly enhances the sense of reality.


T=0.0s T=0.7s T=1.3s

Figure 3. Carangiform schooling-style of fish in short time intervals

Figure 4. Partial screenshot of fish schooling under sea water

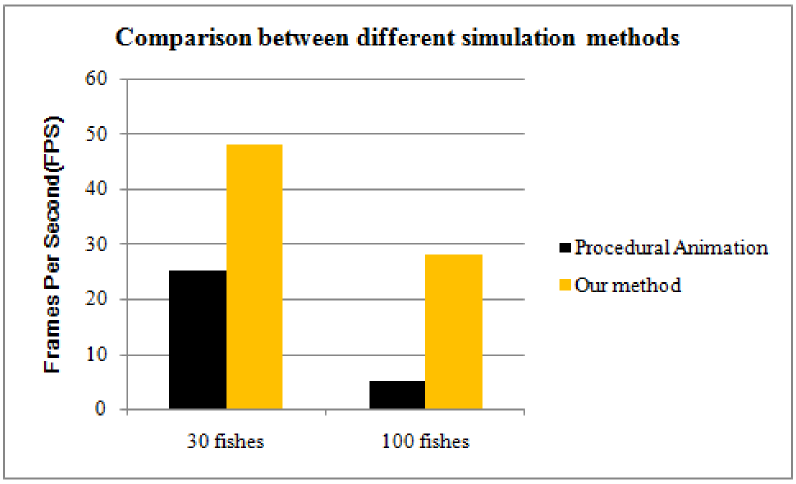

Figure 5. FPS comparison between different simulation methods

In the performance of rendering speed of animation, using our method on the GPU, we have obtained about 48 frames per second (FPS) when rendering 30 fishes; however, the procedural animation of fish [5] only acquires 25 FPS. The difference of rendering time between our method and the method [5] is more obvious when increasing numbers of fishes greatly. The difference of rendering speed is between two methods as shown in Figure 5. From Figure 5, our method is about 1.92-5.6 times faster than procedural animation of fish [5].

5. Conclusions.

We have implemented simulation of fish muscles locomotion and acquired realistic rendering of fish schooling under sea water. The fish muscle activation via spring-mass system can give lifelike sequence of fish swimming shapes and prove our method is effective. Also we simulate the sea water by rendering optical effects and give a real-time rendering based on GPU pipeline. The rendering speed of our method is faster over procedural animation method, and the difference between two methods is more obvious when increasing fish numbers greatly.

Although we have made some practical progress in rendering of fish schooling under sea water, how the fish living habits such as predator stalking and mating behavior are simulated is an interesting direction which we are currently working on.

REFERENCES

[1] M. Bergmann and A. Iollo, Modeling, simulation and control of fish-like swimming, *Computational Fluid Dynamics*, pp.689-694, 2011.
[2] C. R. Ward, F. Gobet and G. Kendall, Evolving collective behavior in an artificial ecology, *Artificial Life*, vol.7, no.2, pp.191-209, 2001.
[3] M. Bedau, J. Mccaskill, N. Packard and S. Rasmussen, Studying attention dynamics of a predator in a prey-predator system, *Artificial Life*, pp.337-342, 2000.
[4] *Dynamic Motion Synthesis*, http://www.naturalmotion.com/files/white_paper_dms.pdf, 2005.
[5] A. R. C. Gritt, *Procedural Fish Animation*, Master Thesis, Bournemouth University, 2010.
[6] B. Iman and S. Fotis, Numerical investigation of the hydrodynamics of carangiform swimming in the transitional and inertial flow regimes, *Journal of Experimental Biology*, vol.211, pp.1541-1558, 2008.
[7] P. Nardinocchi and L. Teresi, On the active response of soft living tissues, *Journal of Elasticity*, vol.88, no.1, pp.27-39, 2007.
[8] U. Erra, B. Frola, V. Scarano and I. Couzin, An efficient GPU implementation for large scale individual-based simulation of collective behavior, *IEEE International Workshop on High Performance Computational Systems Biology*, pp.51-58, 2009.
[9] H. Li, A. Kolpas, L. Petzold and J. Moehlis, Parallel simulation for a fish schooling model on a general-purpose graphics processing unit, *Concurrency & Computation Practice & Experience*, vol.21, no.6, pp.725-737, 2009.

[10] F. Borges, A. Gutierrez-Milla, R. Suppi et al., A hybrid MPI+OpenMP solution of the distributed cluster-based fish schooling simulator, *Procedia Computer Science*, vol.29, pp.2111-2120, 2014.

[11] W. P. Lau, R. Du and Y. Zhong, Swaying wire-driven robot fish design and prototyping, *Computer-Aided Design and Applications*, vol.12, no.4, pp.492-496, 2015.

[12] R. Ito, Y. Sawahara, T. Ishizaki and I. Awai, Construction of a secure wireless power transfer system for robot fish, *Proc. of IEEE Wireless Power Transfer Conference*, 2015.

[13] K. Rivanek, J. Konttinen and S. Pattanaik, Fast approximation to spherical harmonics rotation, *Press of ACM*, Boston, pp.10-24, 2006.