

# Simulator Building and Parameter Optimization of an Autonomous Robotic Fish

Jindong Liu, Huosheng Hu

Department of Computer Science, University of Essex  
Wivenhoe Park, Colchester CO4 3SQ

Email: [jliua@essex.ac.uk](mailto:jliua@essex.ac.uk)

**Abstract--** This paper presents a short review on the research of robotic fish. A simulation environment for robotic fish is built and the experiment shows that it is a convenient way to make research on the robotic fish's motion control algorithm. Furthermore, a parameter optimizing method for fish's traveling wave approximating is proposed. A special error function is selected depending on the hydrodynamics theory of fish.

## I. INTRODUCTION

In nature, fish has astonishing swimming ability after thousands years evolution. It is well known that the tuna swims with high speed and high efficiency, the pike accelerates in a flash and the eel could swims skillfully into narrow holes. Such astonishing swimming ability inspires the researchers to improve the performance of aquatic man-made systems. An example application is robotic fish.

Instead of the conventional rotary propeller used in ship or underwater vehicles, the undulation movement like fish provides the main energy of the robotic fish. The observation on the real fish shows that this kind of propulsion is more noiseless, effective, and maneuverable than the propeller-based propulsion. So, the robotic fish could be used in many marine and military fields such as exploring the fish behaviors, detecting the leakage of oil pipeline, robotics education, mine countermeasures, etc.

In 1994, the first robot fish named robotuna was developed by MIT[2]. In 1998, the Draper Lab realized Vorticity Control Unmanned Undersea Vehicle (VCUUV)[3] on the base of robotuna. VCUUV could avoid obstacle and realize the up-down motion. It is the most widely known robotic fish developed up to now. After that, many researchers put forward several kinds of robot fish. The Northwestern University applied Shape Memory Alloy(SMA) on the robotic lamprey[6] which make aim to realize mine countermeasures. In Japan, Nagoya University developed a kind of micro robotic fish using ICPF Actuator[7] and Tokai University realized a robotic Blackbass[5] to research the propulsion of pectoral fins. National Maritime Research Institute of Japan developed many kinds of robotic fish prototypes from PF300 to PPF-09[8] to exploit the up-down method and effective swimming. In China, the Beihang University (BUAA) developed three kinds of robotic fish[9] and the Institute of Automation Chinese Academy of Sciences (CASIA)[10] also made some progress on four-joint robotic fish. Table 1 lists the most of researches on robotic fish in the world.

**Table 1: The robotic fish research in the world**

| Nation | Researcher                           | Project  |
|--------|--------------------------------------|--|
| U.S.   | MIT                                  | Robotuna(1994) [2], VCUUV(1998) [3]                                  |
|        | Northeastern University              | robotic lamperry [6]   |
|        | California Technology Institute      | Sensors and Control of Robot fish[4]                                 |
|        | University of New Mexico             | The application of IEM and artificial muscle on the robotic fish[13] |
| U.K    | Heriot-Watt University               | FLAPS project[1]   |
| Japan  | National Maritime Research Institute | PF-300, PF-600, PPF-09, PF-700, UPF-2001[8]                          |
|        | Tokai University                     | Robot Blackbass[5]   |
|        | Nagoya University                    | Microroobt using ICPF Actuator[7]                                    |
| China  | BUAA                                 | mini-robotfish and SPC[9]  |
|        | CASIA                                | Robotfish[10]  |
| Other  | Professor Dan Mssie                  | Robotic fish named Dongle[14]  |

Most of previous research focused on the hydrodynamics mechanism of fishlike swimming, the special skin material and mechanical structure of robotic fish models. Although, for future application, autonomous motion control is definitely necessary for robotic fish, there is almost no person to do research on it.

The aim of the project described in this paper is to design and build an autonomous navigation robotic fish. The robotic fish would have two main features: to swim like real fish and to realize autonomous motion control. Two of optimal elements are intelligent motion control algorithm and fish-like motion parameters. Here, a simulation environment is built up in Part II to explore an efficient autonomous navigation algorithm based on the four-joint robotic fish of CASIA (Figure 1). Furthermore, a parameter optimization method for fish's traveling wave approximating is proposed in Part III. A special error function is selected depending on the

hydrodynamics theory of fish. Two experiments are given in Part IV. Finally, section V summarizes the research progress and potential future development.

## II. SIMULATION ENVIRONMENT

To study the kinematic and hydrodynamic theory, a virtual swimming pool and robotic fish have been built based on a robotic fish in our lab. The main targets of the simulation work are:

- To simulate the hydrodynamic model of a robotic fish, to understand the relationship between the fish tail movement and the forces acting on fish.
- To develop fish-like motion control algorithms for the robotic fish, to realize or mimic the real fish behaviors such as decelerating/accelerating swim, constant swim, turning and hover.
- In large noise situation (the wave effect), to test the algorithms of artificial intelligence in robotic fish such as to avoid obstacle, to pursue a moving target, to swim in an appointed trace, etc.



Figure 1 The Robotic Fish

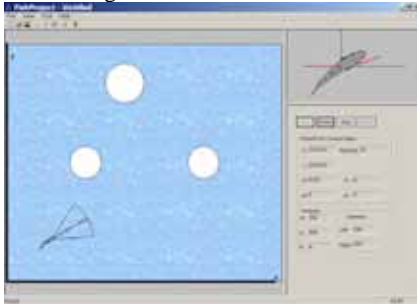


Figure 2. The simulation Environment

### A. Components of Simulation Environment

Figure 3 shows the main frame of the simulator that consists of: Object Layer, Data Layer and User Interface Layer.

The Object layer consists of all virtual objects in the simulator such as robotic fish, sonar, fish joints, ambient, obstacles, task and noise. The virtual robotic fish includes sonars and joints. The ambient describes the virtual swimming pool in which the robotic fish swims. It includes several obstacles which are simplified as round shape. The task records the appointed trace which the robotic fish is expected to follow. The noise simulates the disturbance of water when the robotic fish swims.

The Data Layer is a data exchange center. All virtual environment data is stored and exchanged by it. It is also a connector between User Interface Layer and Object layer.

The User Interface Layer is an interactive layer between users and the simulation environment. Its responsibility is to transfer the command and initial object data from users to Data Layer. It also could display the real-time update of objects on the screen. Figure 2 shows an example of user interface.

The whole simulation environment is built by “Object Oriented” programming method. C++ is selected as the program language.

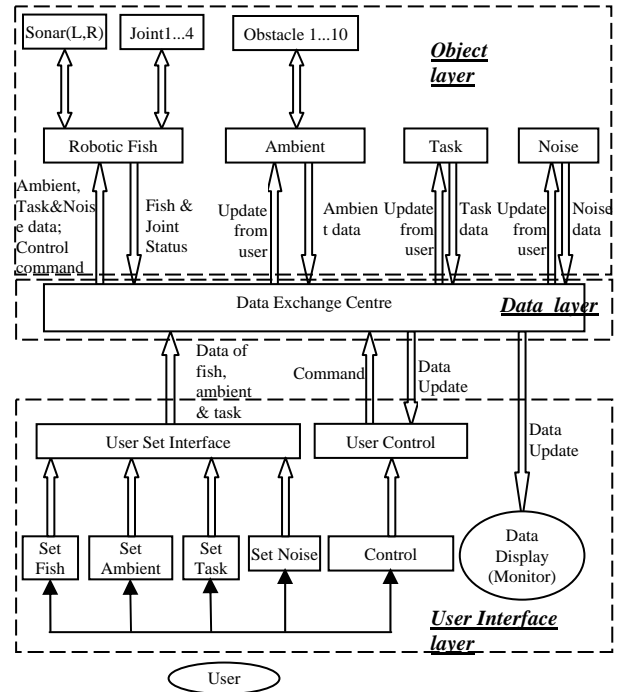


Figure 3. The simulation environment

The virtual robotic fish has three main components: head, 4 joints and 2 sensors. Each fish joint is viewed as a quadrilateral which is defined by four vertexes. It could rotate at the base point by a relative angle related to the anterior joint. Where, the base point is the center of the conjunct line of two adjacent joints. In the real world, the base-point is the position to fix the joint motor. Figure 4 shows a fish joint transformation when a robotic fish swims. The joint will first rotate  $\theta$  following the fish head and then rotate  $\varphi$  at its axle. Figure 5 gives a posture of robotic fish by appointing the heading, the relative angle between Joint\_I vs. Head, Joint\_II vs. Joint\_I, Joint\_III vs. Joint\_II, Joint\_IV vs. Joint\_III. An angle set  $\Phi$  denotes the above four relative angles. When the  $\Phi$  changes as some wave function, the virtual robotic fish will swim like real fish. See the Part III for details.

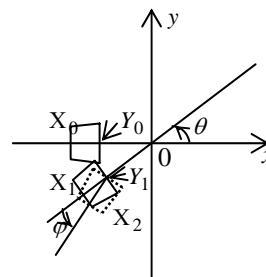


Figure 4 Joint Transformation

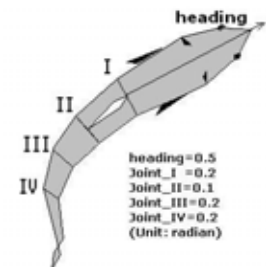


Figure 5. A virtual robotic fish

Two sonar sensors are fixed on the “eyes” position of the virtual fish. The virtual sonar sensor (Figure 6-B) is built up on the base of real sonar sensors (Figure 6-A[15]). The transducer and the receiver are viewed as a same point. A bunch of rays is created at the front of the T/R point to simulate the ultrasonic wave. When the robotic fish try to detect obstacle by sonar, just to compute the crossing point of each ray with the obstacle in ambient.

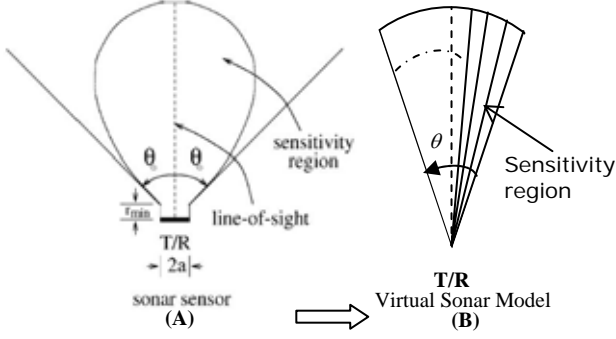


Figure 6. The sonar model in our robotic fish

There are three types of task: free swimming, static trace and dynamic trace. In this paper, the experiment is limited in the free swimming task. The robotic fish wanders in a swimming pool without a goal. In every data update cycle of robotic fish, a random command is sent out. The aim of such task is to test the ability of avoiding obstacle and detecting pool border.

### B. Computation in Virtual Robotic Fish

In simulation, an “Update Cycle” is predefined to control the update of the status of robotic fish and fish joints in real time, where, the status means kinemics information such as position and velocity. Figure 7 shows the main components in one of update cycles. There are six processing models and one fish behaviors library. The detailed process can be described as follows:

First, the robotic fish gets itself current status which includes position, fish heading, linear velocity, angular velocity, linear acceleration, angular acceleration and the time of last updating. The “Task Compare” model is called to compare the current fish position with the expected trace, i.e. task, and outputs the  $\Delta$  Task for “Make Decision” model.

Then the “Detect Obstacle” model and “Pre-proccession” model are called to get the obstacle direction and distance.

Third, in “Make Decision Model” the robotic fish makes decision depending on the obstacle information and the result from task compare model. The decision is limited in the scope of “Fish Behaviors Library” which stores all possible fish-like behaviors.

Fourth, the robotic fish transforms the decision into the expected linear acceleration and angular acceleration for the next time.

Finally, in “Kinematic and Hydrodynamic Model (K&D model)”, the robotic fish computes the hydrodynamic parameter to get the “real” linear acceleration and angular acceleration from the expected. Then the status of robotic fish and joint for next time are got from kinematic model. The

reason of why the expected  $\hat{A}_r^{i+1}, \hat{A}_r^{i+1}$  may be different from the real  $A_r^{i+1}, A_r^{i+1}$  is caused by the mechanical limitation of real robotic fish and its special hydrodynamic feature. Lighthill’s “Large-amplitude elongated-body theory”[12] is selected as the basic theory of hydrodynamic model. The detail of (K&D model) is shown in Figure 8.

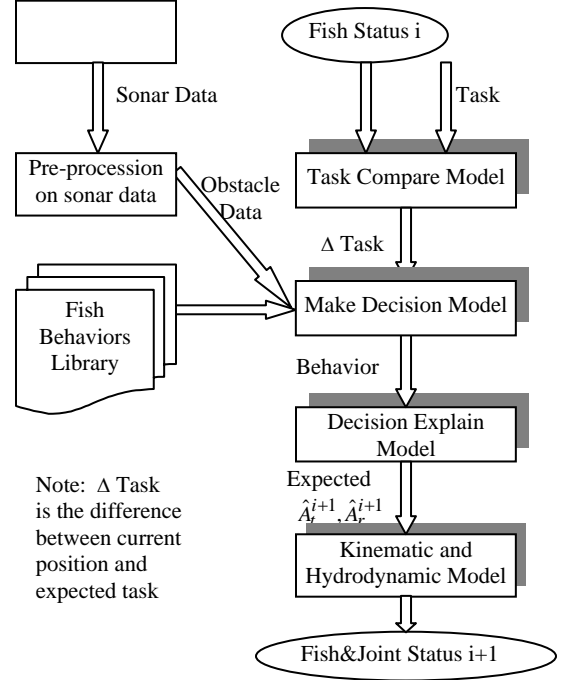
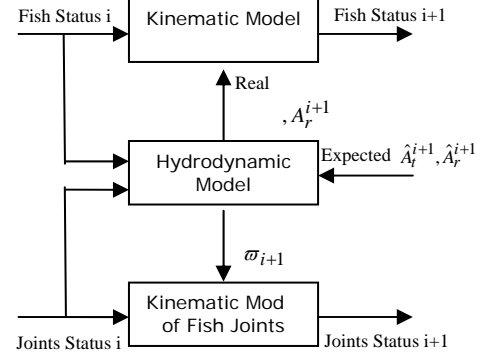


Figure 7. Update Cycle for fish and joints



Note:  $A_r^{i+1}, A_r^{i+1}$  are the linear speed and rotational speed of robotic fish respectively.  $\omega_{i+1}$  is the phase-angle between Joint Status  $i$  and Joint Status  $i+1$ . The expected  $\hat{A}_r^{i+1}, \hat{A}_r^{i+1}$  are the result of Decision Model.

drodynamics Model of our robotic fish

### III. PARAMETER OPTIMIZATION

The motion of fish tail could be described by a traveling wave (1) which was originally suggested by Lighthill[11]. The original point of (1) is set at the conjunction point between

fish head and tail. The parameter of traveling wave changes depending on the kind of fish and the fish kinetics status in water. So, the swimming of fish could be viewed as making the fish body approximate the traveling wave. For real fish, it has tens of vertebrae that could be viewed as tens of mini joints to approximate the wave. So the approximate result is very smooth. But for our robotic fish, it only has four joints, which is impossible to generate smooth wave. How to use limited joints to approximate the traveling wave in minimal error is the topic discussed in this paragraph.

$$y_{body}(x, t) = (c_1x + c_2x^2) \sin(kx + \omega t) \quad (1)$$

where  $y_{body}$  transverse displacement of tail unit;  $x$  displacement along main axis;  $k = \frac{2\pi}{\lambda}$  wave number;  $\lambda$  wave length;  $c_1$  linear wave amplitude envelope;  $c_2$  quadratic wave amplitude envelope;  $\omega = 2\pi f$  wave frequency.  $t$  time.

In [16], a discrete traveling wave is considered and time  $t$  is separated from (1) in order to simplify the control method on joint motors of robotic fish. The rewritten equation is (2) in which the original traveling wave is decomposed into two parts: the time-independent wave sequence  $y_{body}(x, i) (i = 0, 1, \dots, M-1)$  in an oscillation cycle and the time-dependent oscillation frequency  $f$ .

$$y_{body}(x, i) = (c_1x + c_2x^2) \sin(kx - \frac{2\pi}{M}i) \quad i \in [0, M-1] \quad (2)$$

Where,  $i$  is serial number in an oscillation cycle.  $M$  is the resolution of discrete traveling wave. Figure 9 is an example for one oscillation cycle at  $M = 18$ . Figure 10 shows an approximating result by four joints. The "I...IV" is the joint number. It is clear that the end point of each joint is the key element for traveling wave approximating. In[16], the method to find end point assumed each point falls onto the traveling curve. But we found the error of it is large. Here, a special error function is proposed considering the hydrodynamic theory of fish and a simple computation method is applied to get end points.

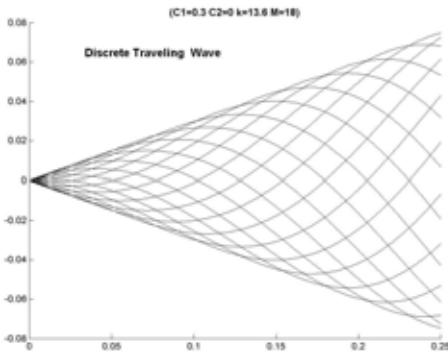


Figure 9 the discrete traveling wave

Before discussion, it is necessary to define some special terms. See Figure 11. When a length-fix line  $L$  which starts from "Base\_Point" tries to approximate a curve defined as  $y = f(x)$ , assume the first crossing point from "Base\_Point" between  $L$  and  $y = f(x)$  as "Cross\_Point". Another end of  $L$  is defined as "End\_Point". The function of line  $L$  is defined as  $y = g(x)$ . So we get following two terms:

$$Positive\_Reg = \int_{Base\_x}^{Cross\_x} [g(x) - f(x)] dx \quad (3)$$

$$Negative\_Reg = \int_{Cross\_x}^{End\_x} [g(x) - f(x)] dx \quad (4)$$

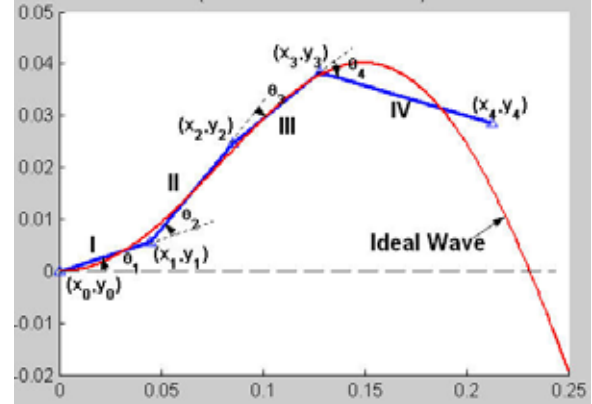


Figure 10 a Traveling wave approximating example

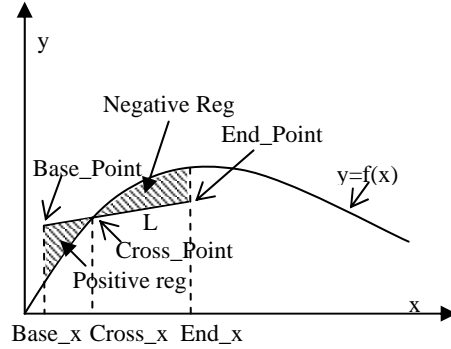


Figure 11 Term Definition

There are two main methods to explain how thrust force is generated for fish: an added-mass method and a lift-based (vorticity) method[1]. Carangiform mode that our robotic fish belongs to is associated with the added-mass method. As the propulsive wave passes backward along the fish, the momentum of the water passing backward is changed by the movement of the fish tail, which causes a reaction force  $F_R$  from water to fish.  $F_R$  is analyzed into a lateral  $F_L$  and a thrust  $F_T$  component which contributes to forward propulsion for fish. The mass of the water passing backward is called added-mass. Its magnitude is partly depending on the wave of the fish tail.

When using a length-fix joint to approximate a wave of fish tail, the ideal situation is to make the added-mass pushed away by joint equal to one by wave. i.e.  $Positive\_Reg + Negative\_Reg = 0$ . So the following error function is selected for traveling wave approximation instead of common quadratic error function.

$$e(x) = |Positive\_Reg + Negative\_Reg| = \left| \int_{Base\_x}^{End\_x} [g(x) - f(x)] dx \right| \quad (5)$$

When applying four joints to approximate the traveling curve, the task is to find a series of “End\_Point” to minimize the

$$e_{i,j}(x) = \left| \frac{End\_x_{i,j} - \int_{Base\_x_{i,j}}^{End\_x_{i,j}} [g_{i,j}(x) - f_i(x)] dx}{Base\_x_{i,j}} \right| (i=0,2\dots M-1, j=1,2,3,4) \quad (6)$$

where  $f_i(x) = y_{body}(x, i)$

$$g_{i,j}(x) = k_{i,j}x + b_{i,j}, k_{i,j} = \frac{End\_y_{i,j} - Base\_y_{i,j}}{End\_x_{i,j} - Base\_x_{i,j}}$$

$$b_{i,j} = Base\_y_{i,j} - k_{i,j}Base\_x_{i,j}$$

The constraint function is:

$$\begin{cases} Base\_x_{i,j} = Base\_y_{i,j} = 0 & j = 1 \\ Base\_x_{i,j} = End\_x_{i,j-1} & j = 2,3,4 \end{cases}$$

$$(End\_x_{i,j} - Base\_x_{i,j})^2 + (End\_y_{i,j} - Base\_y_{i,j})^2 = l_j^2$$

where  $l_j$  is the length of joint  $j$ .

It is difficult to get analytic solutions from (5). So we seek for the numerical solution by MATLAB.

A crossing ratio is defined as  $R_c = \frac{Cross\_x_{i,j} - Base\_x_{i,j}}{End\_x_{i,j} - Base\_x_{i,j}}, R_c \in [0,1]$ , an equations sets is defined as:

$$\begin{cases} (x - Base\_x_{i,j})^2 + (y - Base\_y_{i,j})^2 = (R_c l_j)^2 \\ y = (c_1 x + c_2 x^2) \sin(kx - \frac{2\pi}{M} i) \end{cases} \quad (7)$$

$$i = 1,2\dots M-1, j = 1,2,3,4$$

For a given  $R_c = R_{c0}$ , the equations sets 7 could be solved by iterative method. i.e. a Cross\_Point position for  $R_{c0}$ . Then an End\_Point position  $(End\_x_{i,j}, End\_y_{i,j})_{R_{c0}}$  is easy to get and the parameter in (6) is determined for  $R_{c0}$ . Finally, a  $e_{i,j}(x)|_{R_{c0}}$  could be obtained.

If  $R_c = R_{c0}, R_{c1}, R_{c2}, \dots, R_{cn}$  is set by step  $Step_R = \frac{1}{n}$ , an approximate minimal  $e_{i,j \min}$  could be compute as:

$$e_{i,j \min} = \text{Min}_{k=0\dots n} (e_{i,j}(x)|_{R_{ck}})$$

The corresponding  $(End\_x_{i,j}, End\_y_{i,j})_{R_{ck}}$  is set as the final result for joint  $j$  & serial number  $i$ .

By using same method, a 2D array  $End\_Point[M][4]$  will be achieved and a corresponding  $Joint\_Angle[M][4]$  which stores the joint relative angle  $\theta_{i,j}$  (see Figure 10 for an example) could be obtained easily. The  $Joint\_Angle[M][4]$  will be used for joint motor control in a robotic fish directly.

## IV. EXPERIMENT

### A. Simulation Experiment

Figure 12 is a video sequence for the simulation experiment being built. The robotic fish is trying to finish a “Free Swimming” task to test a simple motion control algorithm. When the virtual fish swims, its joint data is recorded for future analysis.

### B. Parameter Optimization Result

Figure 13 is an example that applies four joints to approximate some traveling wave when  $M = 18$  and  $Step_R = \frac{1}{10}$ . The signal “ $\Delta$ ”s show the End\_Point of each joint. The 8-shape curves are the trace of End\_Point in one cycle.

Figure 14 is two curves of error sum .vs. serial num. i.e.:

$$Sum_{error}(i) = \sum_{j=1}^4 e_{i,j \min}, i = 0,2\dots 17 \quad (8)$$

$Sum_{error}(18) = Sum_{error}(0)$ , the error curve A is the result of the method in[16] and the error curve B is the result of the new method in this paper.

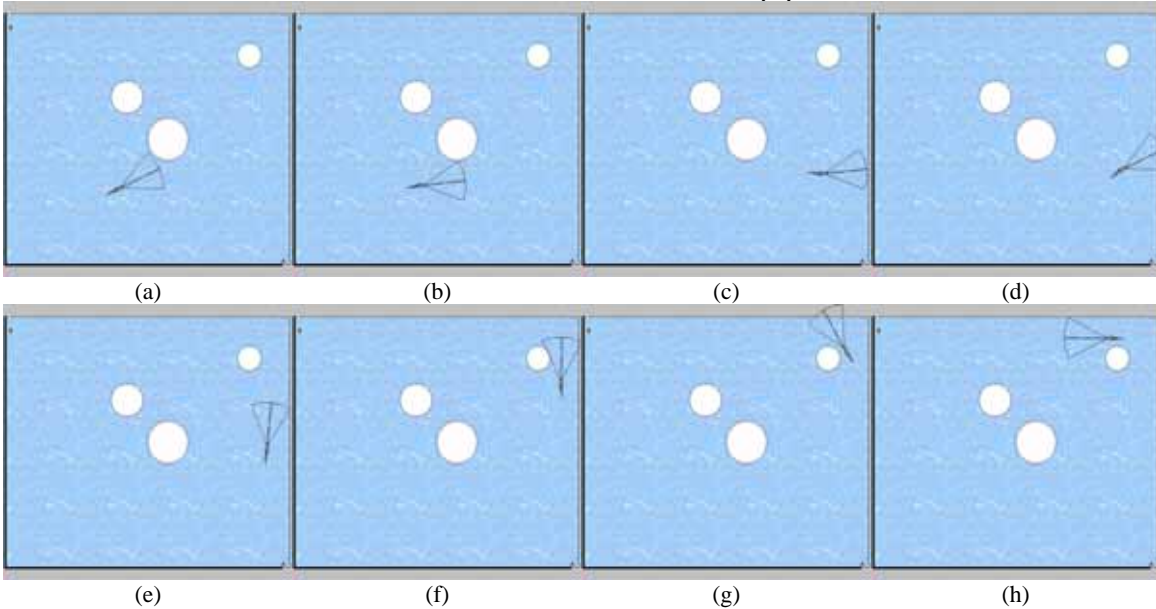


Figure 12. An Experiment of Free Swimming

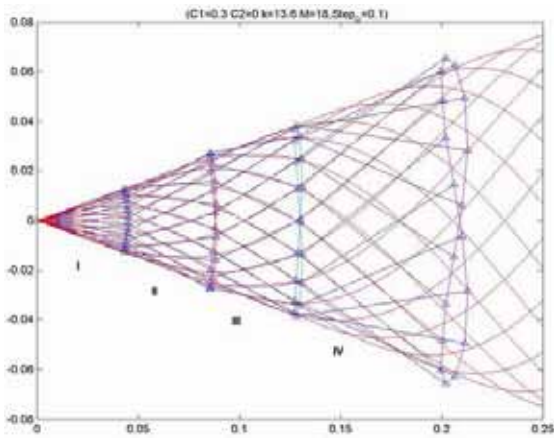


Figure 13 an example of traveling wave approximation

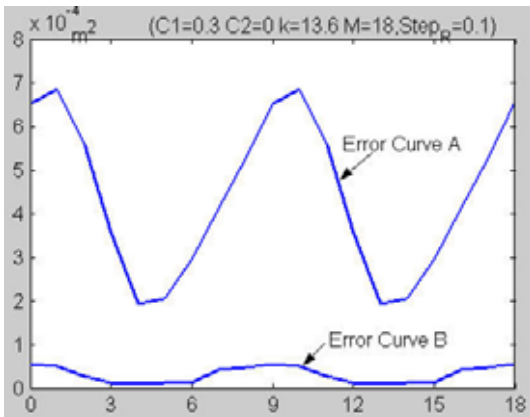


Figure 14 Error Sum Curve

## V. CONCLUSION AND FUTURE WORK

The experimental results have shown that that the simulation environment is a convenient method to do research on robotic fish motion control. The new method to compute End\_Point is with less error than the old method.

In future, we would optimize the parameters:  $C_1, C_2, k$  for the robotic fish to achieve high efficiency of thrust. Furthermore, the motion control method for other kinds of task will also be investigated.

## VI. ACKNOWLEDGEMENT

Our thanks go to Dr. Wang Shuo and Dr. Yu Junzhi for their robotic fish platform.

## REFERENCES

- [1] M. Sfakiotakis, etc. "Review of Fish Swimming Modes for Aquatic Locomotion. IEEE Journal of Oceanic Engineering". 1999,24(2): 237-252
- [2] K. Streitlien, G. S. Triantafyllou, and M. S. Triantafyllou, "Efficient foil propulsion through vortex control," AIAA J., vol. 34, pp. 2315–2319, 1996.

- [3] J.M.Anderson, "Vorticity control for efficient propulsion," Ph.D. dissertation, Massachusetts Inst. Technol./Woods Hole Oceanographic Inst. Joint Program, Woods Hole, MA, 1996.
- [4] K.A. Morgansen, Vincent Duindam, Richard J. Mason. "Nonlinear Control Methods for Planar Carangiform Robot Fish Locomotion". Proceeding of the 2001 IEEE International Conference on Robotics and Automation. pp 427-434.
- [5] N. Kato. "Control performance in the horizontal plane of a fish robot with mechanical pectoral fins". IEEE Journal of Oceanic Engineering 25 1 2000 IEEE p 121-129 0364-9059
- [6] <http://www.dac.neu.edu/msc/burp.html>
- [7] S. Guo, T. Fukuda, Norihiko KATO, Keisuke OGURO. "Development of Underwater Microrobot Using ICPF Actuator". Proceedings of the 1998 IEEE International Conference on Robotics & Automation. pp1829-1834.
- [8] <http://www.nmri.go.jp/eng/khirata/fish>
- [9] J. Liang, etc. "Researchful Development of Underwater Robofish II- Development of a Small Experimental Robofish". Robot .2002, 24(3)
- [10]J.Z. Yu, E.K. Chen, S. Wang, M. TAN, "Research Evolution and Analysis of Biomimetic Robot Fish". Control Theory and Application. Accepted in Chinese.2002
- [11]M. J. Lighthill, "Note on the swimming of slender fish," J. Fluid Mech., vol. 9, pp. 305–317, 1960.
- [12]M. J. Lighthill, "Large-amplitude elongated-body theory of fish locomotion," in Proc. R. Soc. Lond. B, 1971, vol. 179, pp. 125–138.
- [13]M. Mojarrad; M. Shahinpoor. "Noiseless propulsion for swimming robotic structures using polyelectrolyte ion-exchange membrane". The International Society for Optical Engineering v 2716 Feb 26-27 96 Sponsored by: SPIE - Int Soc for Opt Engineering, Bellingham, WA USA Society of Photo-Optical Instrumentation Engineers p 183-192
- [14][http://www.seattlerobotics.org/encoder/200211/autonomous\\_robotic\\_fish.html](http://www.seattlerobotics.org/encoder/200211/autonomous_robotic_fish.html)
- [15]B. Ayru lu and B. Barshan. "Reliability measure assignment to sonar forro bust target differentiation". Pattern Recognition 35 (2002) 1403–1419.
- [16]J.Z. Yu, etc. A simplified Propulsive Model of Biomimetic Robot Fish and Its Realization. Submitted.
- [17]C.C. Lindsey, "Form, function and locomotory habits in fish". Fish Physiology Vol. VII Locomotion. New York: Academic. 1978. pp.1-100.