Development of a Biomimetic Robotic Fish and Its Control Algorithm

Junzhi Yu, Min Tan, Shuo Wang, and Erkui Chen

***Abstract*—This paper is concerned with the design of a robotic fish and its motion control algorithms. A radio-controlled, four-link biomimetic robotic fish is developed using a flexible posterior body and an oscillating foil as a propeller. The swimming speed of the robotic fish is adjusted by modulating joint's oscillating frequency, and its orientation is tuned by different joint's deflections. Since the motion control of a robotic fish involves both hydrodynamics of the fluid environment and dynamics of the robot, it is very difficult to establish a precise mathematical model employing purely analytical methods. Therefore, the fish's motion control task is decomposed into two control systems. The online speed control implements a hybrid control strategy and a proportional-integral-derivative (PID) control algorithm. The orientation control system is based on a fuzzy logic controller. In our experiments, a point-to-point (PTP) control algorithm is implemented and an overhead vision system is adopted to provide real-time visual feedback. The experimental results confirm the effectiveness of the proposed algorithms.**

***Index Terms*—Biomimetic robotic fish, orientation control, point-to-point (PTP) control, speed control.**

I. Introduction

ROBOTICS research is driven by the challenge of extending robot technology to complex and dynamic environments to some extent, especially inaccessible ones to human. Inspired by biomimetics, robotic devices are being developed to investigate and assess aquatic biological systems and their locomotion mechanisms for better performance. It is well-known that a fish in nature propels itself by the coordinate motion of its body, fins, and tail, achieving tremendous propulsive efficiency and excellent maneuverability that has the advantage over conventional marine vehicles powered by rotary propellers with the same power consumption. Nature selection has ensured that the mechanical systems evolved in fish are very efficient and adapted to their living environments. The fish, in a sense of engineering, is a distinguished autonomous underwater vehicle (AUV) prototype. In recent years, growing research in propulsion and maneuvering mechanisms used by fish has demonstrated a variety of prospective utilities in undersea vehicles [1]–[3], and some reviews concerning fish swimming and the analytical methods that had been applied to some of their propulsive mechanisms have appeared [4], [5]. In 1994, MIT successfully developed an eight-link, fish-like machine—RoboTuna, which may be the first free-swimming robotic fish in the world. RoboTuna and subsequent RoboPike projects attempt to create AUVs with increased energy savings and longer mission duration by utilizing a flexible posterior body and a flapping foil (tail fin) that exploits external fluid forces to produce thrust. Meanwhile, another motivation is to answer Gray's paradox, which is that a fish does not seem to have enough power to propel itself at the speed it does [6]. Since then, based on recent progress in robotics, hydrodynamics of fish-like swimming, new materials, actuators, and control technology, more and more research has focused on the development of novel fish-like vehicles with the advantages in efficiency, maneuverability and noise. As a matter of significance in practical applications, a robotic fish can be applied to military detection, undersea operation, oceanic supervision, aquatic life-form observation, pollution search, and so on.

For the convenience of description we define a robotic fish as a fish-like aquatic vehicle that is based on the swimming skills and anatomic structure of a fish: primarily the undulatory/oscillatory body motions, the highly controllable fins and the large aspect ratio lunate tail. As a combination of biomechanism and engineering technology, the robotic fish is a multidisciplinary study that mainly involves hydrodynamics based control and actuation technology. In this paper, the major objective is to design a radio-controlled, four-link, and free-swimming biomimetic robotic fish that has a flexible posterior body and an oscillating foil as a propulsor, and to develop preliminary motion control strategy of robotic fish systems using visual feedback for the robot's position. The point-to-point (PTP) control, which means how to make a robot move continuously and steadily from an initial point to a final one, is one of the basic problems concerning the robot's controllability. Many complex motions of the fish such as obstacle avoidance and formation control can be reduced to a series of PTP controls.

The rest of the paper is organized as follows. A brief review of previous related work on robotic fish control is introduced in Section II. The overall experimental system and the control performance are described in Section III. A speed control algorithm is presented in Section IV. Then a fuzzy controller for orientation control is designed in Section V. Based upon speed control and orientation control, a PTP control algorithm and corresponding experimental results are addressed in Section VI and

Manuscript received May 20, 2003; revised March 25, 2004. This work was supported by the Robotics Subject of 863 Program 2001AA422370 and by the 973 Program of China 2002CB312200. This paper was recommended by Associate Editor M.S. de Queiroz.

J. Yu was with the Laboratory of Complex Systems and Intelligence Sciences, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China. He is now with the Center for Systems and Control Peking University, Beijing 100871, China (e-mail: jzyu@compsys.ia.ac.cn; junzhiyu@yahoo.com).

M. Tan and S. Wang are with the Laboratory of Complex Systems and Intelligence Sciences, Institute of Automation, Chinese Academy of Sciences, Beijing 100080, China (e-mail: tan@compsys.ia.ac.cn; swang@compsys.ia.ac.cn).

E. Chen is with the School of Communication and Control Engineering, Southern Yangtze University, Wuxi 214036, China (e-mail: cek_chen@sohu.com).

Digital Object Identifier 10.1109/TSMCB.2004.831151

Section VII, respectively. The concluding remarks are presented in Section VIII.

II. Previous Related Work

Body and/or caudal fin (BCF) swimming movements are usually categorized into anguilliform, subcarangiform, carangiform, and thunniform mode basically according to the wavelength and the amplitude envelope of the propulsive wave underlying fish's behavior [4], [7]. In a broader context, recent studies on the robotic fish primarily concentrate on the anguilliform swimming mode and the caragiform swimming mode. During the anguilliform locomotion, the whole body participates in large amplitude undulations, which is common in eel and lamprey. For the carangiform locomotion, the body's undulations are entirely confined to the last 1/3 part of the body, and thrust is produced by means of a rather stiff caudal fin. Compared to anguilliform swimmers, carangiform swimmers are generally faster, but with less agility due to the relative rigidity of their bodies. Also, Carangiform propulsion is more convenient for engineering realization.

Some theoretical and experimental studies have explored the possibility of applying the carangiform propulsive mechanism for aquatic vehicles. Early resistive hydrodynamic models [8] were based on a quasistatic approach that uses steady-state flow theory to calculate the fluid forces. Later models dealt with more realistic fish-type motions, e.g., Wu [9] originally developed a two-dimensional (2-D) waving plate theory, treating fish as an elastic plate. Thereafter, elongated-body theory [10], [11] and large-amplitude elongated-body theory [12], [13] suited to carangiform swimming were formed. These linear or nonlinear extensions of the waving plate theory allow the analysis of fast acceleration and steady swimming.

At present, some artificial systems are developed to investigate fish-like locomotion mechanism. In particular, oscillating foil has been proposed as an alternative propeller to the conventional screw propeller [1], [14], [15]. The development of eight-link, foil-flapping robotic mechanism (RoboTuna) [1], [16] acquired detailed measurements of the forces on an actively controlled body, thus, it demonstrated that the power required to propel an actively swimming, streamed, fish-like body was significantly smaller than the power needed to tow the body straight and rigid at the same speed U. At the same time, a genetic algorithm was employed to optimize RotoTuna's swimming performance [17]. Harper *et al.* proposed the design of an optimal spring constant to actuate the oscillating foil [18]. Kelly *et al.* proposed a model for planar carangiform swimming based on reduced Euler–Lagrange equations for the interaction of a rigid body and an incompressible fluid [19]. Mason *et al.* built a three-link robot system to study carangiform-like swimming. They experimentally verified a quasisteady fluid flow model for predicting the thrust generated by a flapping tail [20]. Morgansen *et al.* used methods from nonlinear control theory to generate system inputs and achieved trajectory tracking for a planar carangiform robotic fish [21]. Kato *et al.* considered the control of pectoral fin like mechanism as a propulsor and built a Blackbass Robot prototype [22]. Hirata *et al.* developed a fish robot prototype and measured its turning performance [14], [23]. Using discrete-time continuous feedback and iteration of motion planning step, Bullo *et al.* presented the motion control algorithms for an underactuated mechanical control system to solve the PTP reconfiguration, static interpolation, and exponential stabilization, which can typically be applied to the model of underwater vehicles [24]. Saimek *et al.* proposed a maneuvering control strategy for a swimming machine, whose control task was decomposed into the offline step of motion planning and the online step of feedback tracking [25].

These interesting investigations, as a matter of fact, do make good progress in carangiform propulsion. However, there is a tremendous amount of research needed to conduct in both integration of the theories and applicability of the real aquatic mechanisms. The contribution of this paper lies in 1) an improved approach to design a robotic fish based on a simplified kinematics model that relates frequency to speed and joint angle bias to turns, where geometric reduction is employed and complex hydrodynamic analysis is avoided and 2) implementation of the different modes of turning, where fuzzy logic is used to directly control the actuators. Contrary to nonlinear control methods for fish's motion described in the above literature, our scheme is easily designed and implemented online. It is also not necessary to find the rigorous mathematical model of the system to design the proposed controller. A drawback is that the system dynamics is not tackled. Therefore, it is quite clear that a need exists for more research about integrating dynamics and kinematics of fish swimming into an actual fish-like system.

III. Robotic Fish Prototype Based on a Simplified Propulsive Model

In 2001, a radio-controlled, four-link biomimetic robotic fish was developed by the Laboratory of Complex Systems and Intelligence Science in the Chinese Academy of Sciences and the Robotic Institute in the Beijing University of Aeronautics and Astronautics, which is 400 mm in length, 40 mm in width, and 78 mm in thickness.

A. Simplified Carangiform Propulsive Model

As described in the last section, carangiform motion involves the undulation of the entire body, whose large amplitude undulation is mainly confined to the last 1/3 part of the body, and thrust is produced by a rather stiff caudal fin. The amplitude of this undulation, however, is small, or zero, in the anterior portion of the fish, increasing drastically in the immediate vicinity of the trailing edge [26]. Based on this information, as shown in Fig. 1, a physical model of the carangiform motion can be divided into two parts: flexible body and oscillatory lunate caudal fin, where the flexible body is represented by a series of oscillatory hinge joints and the caudal fin by an oscillating foil. A relative swimming model for RoboTuna (carangiform) has been presented by Barrett *et al.* [17], whose undulatory motion is assumed to take the form of a traveling wave (1) originally suggested by Lighthill [10]

$$y_{\text{body}}(x,t) = \left[(c_1 x + c_2 x^2)\right]\left[\sin(kx + \omega t)\right] \tag{1}$$

where y_{body} represents the transverse displacement of the fish body, x denotes the displacement along main axis, k indicates

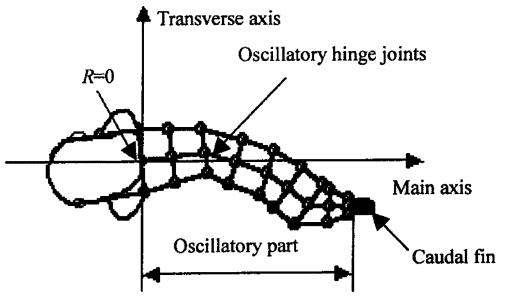


Fig. 1. Physical model of fish swimming.

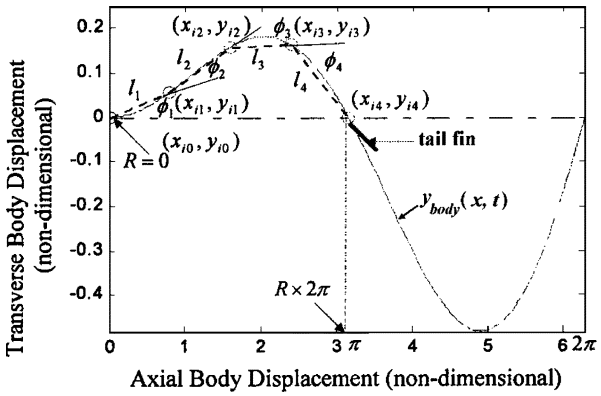


Fig. 2. Link based body-wave fitting.

the body wave number ($k = 2\pi/\lambda$), λ is the body wave length, c_1 is the linear wave amplitude envelope, c_2 is the quadratic wave amplitude envelope, and ω is the body wave frequency ($\omega = 2\pi f = 2\pi/T$).

There is no standard technique for the quantitative analysis of the carangiform motion that encompasses various external forces and torques due to hydrodynamic forces, gravitational forces, buoyant forces, etc. In this section, the kinematics of the carangiform motion will only be discussed depending on the specified propulsive wave, i.e., the above body-wave equation. Given the equation, the following task is to determine the proper body-wave parameters (i.e., c_1, c_2, k, ω, etc.) for a desired swimming motion in terms of some criteria. In [17], a set of seven key parameters for the kinematics model of RoboTuna was captured and a genetic algorithm was used to guide the search for an optimal swimming efficiency. But for various species, dimensions, and shapes of fish, there are different parameter sets to adapt themselves to the surroundings. Consequently, it is an exceedingly tough task to optimize the fish's swimming efficiency and/or maneuverability.

For simplicity, a discrete planar spline curve parameterized as "sinusoid" is taken into account, i.e., time variable t is separated from the body-wave function $y_{\text{body}}(x,t)$. That is to say, the traveling body-wave is decomposed into two parts: the time-independent spline curve sequences $y_{\text{body}}(x,i)$ $(i = 0,1,\ldots,M-1)$ in an oscillation period, which is described by (2), and the time-dependent oscillating frequency f, which is described as the times of recurring oscillation at the unit time interval.

$$y_{\text{body}}(x,i) = \left[(c_1 x + c_2 x^2)\right]\left[\sin\left(kx \pm \frac{2\pi}{M} i\right)\right] \qquad (2)$$

where i denotes the ith variable of the spline curve sequence $y_{\text{body}}(x,i)$, M is called body-wave resolution that represents the discrete degree of the overall traveling wave, which is restricted by the maximum oscillating frequency of actuators. It should be noticed that the "+" sign or the "−" one has the same effect on the sequence $y_{\text{body}}(x,i)$ in an overall oscillation period, but their initial moving direction is different due to different initial values. Notice that the "−" sign is used in this paper.

Considered that the oscillatory part of a fish consists of many rotating hinge joints, as shown in Fig. 2, it can be modeled as a planar serial chain of links along the axial body displacement. The position of each link in the moving chain can then be achieved by numerical fitting. Before fitting the body-wave curve, we define relative wavelength R as the length ratio of wavelength exhibited by the fish's oscillatory part to that of a whole sine wave. If assuming the full fishbody exhibits a whole propulsive wave at $k = 1$, R will be identical with the length ratio of the fish's oscillatory part to that of the fishbody. When R approaches zero, to some extent, the fish's oscillatory part may be viewed as a rigid rod that hardly produces thrust; when R nears to 0.5, half a sine wave reveals in the oscillatory part during locomotion. With the purpose of mimicking carangiform motion, R is empirically around 1/3. To expand the maneuverability and facilitate the realization of the fish-like mechanical system, in actual implementation, a larger value, e.g., 0.5, can be chosen.

As mentioned above, since the wavelength of a whole propulsive wave is viewed as 2π at $k = 1$, the wavelength of the oscillatory part at R is then $R \times 2\pi$. On the assumption that the fish-like mechanical system is to be made up of N joints and body-wave resolution is M, the body wave at an interval of 0 to $R \times 2\pi$ along the axial body displacement can be fitted with a N-link mechanism. Notice that $R \times 2\pi$ is dimensionless. Let the length of each link be l_j $(j = 1,2,\ldots,N)$, to keep nondimensional, the ratio of the links must be normalized to be independent of its actual size, i.e., $l_1 : l_2 : \ldots : l_N = m[l'_1 : l'_2 : \ldots : l'_N]$, where m denotes the length factor, l'_j indicates the normalized length of the j-th link, and especially l'_1 equals 1.0. Also let two end-point coordinate pairs of each link l'_j be (x_{j-1}, y_{j-1}) and (x_j, y_j), respectively, and the joint angle between l'_{j-1} and l'_j be ϕ_j. Then once the amplitude coefficients (i.e., c_1 and c_2) and k are determined, the shape of the propulsive wave at some time will be ensured. Mathematically, the jth link's joint angle ϕ_{ij} at the time of i-th $(i = 0,1,\ldots,M-1)$ can be calculated by fitting the current wave. The following question is to search appropriate joint angle ϕ_{ij} to meet the condition that the end-point of the link l'_j falls into the wavy curve and the x-coordinate of the last link's endpoint (x_{iN}, y_{iN}) just equals $R \times 2\pi$. That is to say, it must satisfy the constraint condition given by

$$\begin{cases} (x_{i,j} - x_{i,j-1})^2 + (y_{i,j} - y_{i,j-1})^2 = {l'_j}^2 \\ y_{i,j} = \left(c_1 x_{ij} + c_2 x_{ij}^2\right) \sin\left(k x_{ij} - \frac{2\pi}{M} i\right) \end{cases} \qquad (3)$$

where the subscript i indicates the ith time of the oscillating sequence, and j denotes the jth link. Through a series of analytical iterative operations, the end-point coordinate pair $(x_{i,j}, y_{i,j})$

can be calculated. Then the slope of each link l'_j at the arbitrary ith time can be computed. Finally, as illustrated in (4), a two-dimensional rectangular array $\mathrm{OscData}[M][N]$ for the joint angle ϕ_{ij} is obtained, which will be used as the primitive oscillating data of the robotic fish. Based on this oscillatory array, the fish body's shape can geometrically be changed by adding different deflections $\Delta\phi_i$ to each joint, the corresponding oscillatory array $\mathrm{OscData}'[M][N]$ is shown in (5). Some deflections may be zero as necessary in practice, shown in (4) and (5) at the bottom of the page

We next assume that the controllability of the fish relies on the internal shape (the joint angle ϕ_{ij}) for maneuverability and the oscillating frequency f of the tail for speed. There are some reasons for this choice, although fish in nature is observed to use a combination of amplitude and frequency for speed control. First, in essence, fish's motion is governed by the body's velocity whose magnitude is its speed and whose orientation is its direction of motion. The separation of speed and direction will lend it self to actual motion control with different methods. Secondly, only the relating frequency to speed is easily realized in the control. We also attempt to increase amplitude during changing frequency, but an acute increase in amplitude often lead to malfunction of running servomotors with high frequency, which is probably due to poor quantitative understanding of the role of amplitude playing in the speed control as well as the limited rotation range of the servomotor. Finally, as will be shown in the following section, a robotic fish controlled in this method can adequately reproduce a carangiform-like motion and obtain a certain level of maneuverability. Based on the success and failure of this design, it can be concluded that the robotic fish actuated by servomotors is feasible but not ideal, and that a more flexible actuator will perhaps need to mimic the full motion of fish in nature.

Since the mechanical robotic fish is equipped with four links, all calculations and experiments in this paper are implemented on a four-link model. The schematic of the link-based body-wave fitting has been shown in Fig. 2. The body-wave parameters we chosen are as follows: $c_1 = 0.1$, $c_2 = 0.05$, $k = 1$, $R = 0.5$, $l'_1 : l'_2 : l'_3 : l'_4 = 1 : 1 : 0.8 : 0.6$, $M = 18$, and $N = 4$. Regarding the selection of amplitude coefficients c_1 and c_2, it must be noted that the maximum sideways tail motion for elongate is approximately 20% of the fish length [27], i.e., satisfying the constraint given by

$$\max\left(2|c_1 x + c_2 x^2|\right) < 0.2BL \tag{6}$$

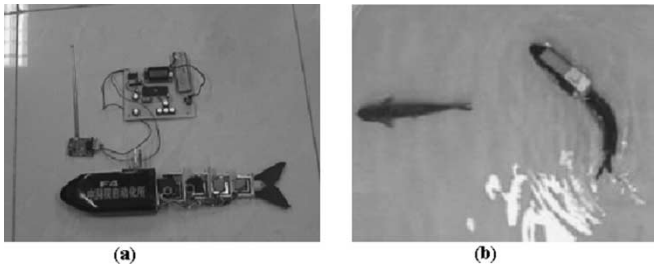


Fig. 3. Top view of a robotic fish. (a) Prototype of a robotic fish and its remote controller. (b) Swimming robotic fish versus real fish (*carp*).

where BL denotes the body length of the robotic fish. In general, the more the number of the links, the better the mechanism's maneuverability and redundancy, but the harder the control and construction of the robot. In a synthetic way, 2–6 links are perhaps appropriate for the robotic fish design. In a strict way, such a robotic fish design based on the above parameters may not be carangiform, since both theoretical and practical factors have to be synthesized during realization.

As mentioned before, the eventual results for the propulsive wave fitting according to the given parameters are a 2-D 18×4 rectangular array of joint angles and the oscillating frequency, which are independent of the fish's dimensions and shapes. Therefore, a parameters set $\{\phi_{i1}, \phi_{i2}, \phi_{i3}, \phi_{i4}, f\}$ is chosen to control the fish's motion. The details of the control method of the speed and the turning will be presented in Section III-B.

B. Basic Configuration of the Robotic Fish Prototype

The robotic fish prototype that we are currently developing is presented in Fig. 3. It is controlled by a remote controller shown in Fig. 3(a). Fig. 3(b) shows that it swims with a real carp fish in a water pond. The robotic fish, as illustrated in Fig. 4, primarily consists of

- control unit (onboard microprocessor + peripherals);
- communication unit (wireless receiver);
- support (aluminum exoskeleton + head + forebody);
- actuation unit (4 dc servomotors);
- accessories (battery, waterproofed skin, tail fin, etc.).

C. Control System and Control Performance

In the fish's control unit, four servomotors are controlled by an onboard microprocessor and a CPLD (complex programmable logic device). The speed of fish's straight motion is adjusted by modulating the joint's oscillating frequency, and its orientation is tuned by different joint's deflection. Adding

$$\mathrm{OscData}[M][N] = \begin{pmatrix} \phi_{01} & \phi_{02} & \cdots & \phi_{0N} \\ \phi_{11} & \phi_{12} & \cdots & \phi_{1N} \\ \cdots & \cdots & \cdots & \cdots \\ \phi_{M-1,1} & \phi_{M-1,2} & \cdots & \phi_{M-1,N} \end{pmatrix} \tag{4}$$

$$\mathrm{OscData}'[M][N] = \begin{pmatrix} \phi_{01} + \Delta\phi_1 & \phi_{02} + \Delta\phi_2 & \cdots & \phi_{0,N} + \Delta\phi_N \\ \phi_{11} + \Delta\phi_1 & \phi_{12} + \Delta\phi_2 & \cdots & \phi_{1,N} + \Delta\phi_N \\ \cdots & \cdots & \cdots & \cdots \\ \phi_{M-1,1} + \Delta\phi_1 & \phi_{M-1,2} + \Delta\phi_2 & \cdots & \phi_{M-1,N} + \Delta\phi_N \end{pmatrix} \tag{5}$$

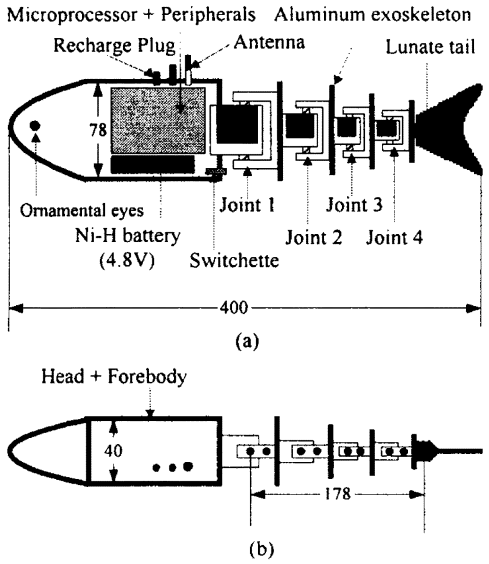


Fig. 4. Mechanical configuration of the robotic fish. (a) Front view. (b) Top view.

TABLE I
TECHNICAL PARAMETERS OF THE ROBOTIC FISH PROTOTYPE

Size (L×W×H)	~400×40×78 mm^3
Weight	~0.5 kg
Number of joints	4
Max oscillating frequency	2 Hz (in water)
Length of oscillatory part	178 mm
Maximum speed	~0.32 m/s
Minimum turning radius	~200 mm
Maximum torque	3.2 kgf×cm
Working voltage	4.8 V
Drive mode	DC servomotor
Control mode	Radio control (315M)

various deflections, plus or minus, to the joint angles ϕ_{i1} and ϕ_{i2} in each oscillation period, different motional direction is achieved. The basic technical parameters of the robotic fish prototype are described in Table I.

Through a wireless receiver, the commands from the upper level are transformed to an oscillating frequency and rotation angles of the servomotors that are adjusted by PWM (pulse width modulation) signals generated from CPLD. Recently, two control modes have been developed for the prototype: the manual mode and the automatic control mode.

- For the manual mode, a remote controller imitating a game joystick is developed, as shown in Fig. 3(a). The fish is able to accelerate, decelerate, turn right and turn left by pressing keys: UP/DOWN (for speed control) and LEFT/RIGHT (for orientation control).
- In the automatic control mode, an overhead vision system is adopted to control the robotic fish in a closed loop. A CCD camera hung over the swimming pond acts as a sensor to capture the fish's motion and surrounding information, which provide a real-time visual feedback to be detailed in Section VII. The speed of straight swimming and turning performance, by means of online visual feedback of the fishes' position and orientation, are then evaluated.

TABLE II
OSCILLATING FREQUENCY (Hz) VERSUS SPEED (m/s) OF STRAIGHT SWIMMING

Frequency	0.5	0.57	0.67	0.8	1.0	1.34	2.0
Speed	0.12	0.16	0.18	0.20	0.23	0.26	0.32

TABLE III
ANGULAR SPEED (rad/s) AT $f = 2$ Hz

Deflections	−30	−22.5	−15	−7.5	0	7.5	15	22.5	30
Angular speed	−1.2	−0.92	−0.8	−0.5	−0.1~0.1	0.4	0.75	0.95	1.1

Based upon the experimental data, the relationship between the oscillating frequency f and the straight swimming speed V is shown in Table II. It is observed that the maximum swimming speed nears to 0.32 m/s, i.e., about 0.8 times of body length per second, at the frequency of 2 Hz with a lunate tail fin. In contrast to a real fish, such a swimming efficiency is not high due to large drag between the oscillatory part and the water. However, a general tendency is that the swimming speed increases with the oscillating frequency. For a practical reason, the speed cannot be infinitely expanded since the servomotors can hardly follow sufficiently high speeds in high oscillating frequency areas.

Further more, as an essential element of maneuverability, the turning performance of the fish is measured. Table III shows the experimental results for an angular speed at $f = 2$ Hz. The fish is required to round a 2π circle with different turning radiuses in the experiments, and corresponding angular speeds are obtained. During the measurements, eight directional levels are sampled at intervals of 7.5°. The corresponding deflection in degrees is added to the first two joint angles (ϕ_{i1}, ϕ_{i2}) in each oscillation period, accordingly the fish body deflects to one side. Why not add deflections to all joint angles? A precondition is firstly imposed that a robotic fish moving in the form of body wave is efficient and agile. The fish is then required to motion in the form of body wave as much as possible. In the case of adding two deflections, the fish is more maneuverable compared to the case of adding only one deflection, but it makes no marked difference to the case of adding three deflections. So two defections are put in actuality. It is also attempted to add deflections to the last two joints (ϕ_{i3}, ϕ_{i4}), but some differences to turning radius are found. In this paper, only deflections added to the first two joints are discussed. As seen from Table III, the angular speed increases with augment of deflections, and the values of the angular speed are not quite symmetric with bias. This problem seems to indicate that the current system is lack of mechanical symmetry due to joint binding, motor installation and so on.

Three basic turning modes for a fish propelled only with oscillating tail fin, where initially discussed by Hirata *et al.* [23], are then redefined as follows.

- **Mode A**: turning during advancing [Fig. 5(a)]. In this mode, the robotic fish intentionally deflects its body only to one side by exerting on geometric bias during advance, where the head and forebody of the fish are acting as passive vehicle and the tail is acting both as the rudder and the propulsor. This is a fundamental turning mode in that the robotic fish can turn at various turning-radiuses and speeds in this manner.

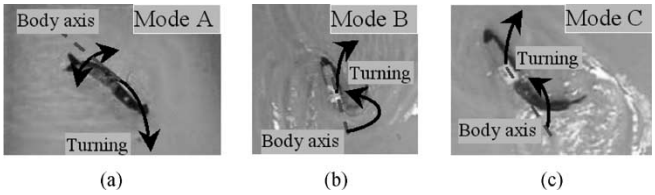


Fig. 5. Three basic turning modes of robotic fish. (a) Turning during advancing. (b) Snap turning. (c) Turning from rest.

- **Mode B**: snap turning [Fig. 5(b)]. The robotic fish, using this mode, suddenly bends its body to a "C" sharp and keeps the posture during motion, i.e., all the joints synchronously reach their left/right oscillatory limits. As is well known, C-shaped and S-shaped movements are important for most fishes when escaping predators and for some fish in achieving prey capture [28]. The remaining kinetic energy and hydrodynamic forces jointly act on the C-shaped fish body so that its motion direction changes drastically, and its turning radius is the smallest of three modes. Therefore, this turning mode is very effective in occasions needed high maneuverability or fast-turn. For our robotic fish prototype, its snap turning radius is about $0.5BL$ and the angular speed around 1.0 rad/s.
- **Mode C**: turning from rest [Fig. 5(c)]. The robotic fish, in this case, deflects its body only to one side swiftly from a stationary state. The inertia force and drag of the fish body and the tail fin are changed to the moment of rotation, and the fish turns quickly. This mode is also suitable for fast start and directional adjustment with large-angle, but its turning speed and turning angle are difficult to control accurately in practice.

After performing some simple movements such as going forward, turning left and turning right, a high quality control system can be developed by a combination of these basic moving patterns. Considered that a parameters set $\{\phi_{i1}, \phi_{i2}, \phi_{i3}, \phi_{i4}, f\}$ can be reduced to control fish's motion in the above propulsive model, the robotic fish's motion control is then decomposed into the speed control and the orientation control. The aim of the speed control is to search a certain oscillating frequency f so that the fish moves fast and steadily. While the orientation control is designed to choose appropriate joints angles $\{\phi_{i1}, \phi_{i2}, \phi_{i3}, \phi_{i4}\}$ to navigate the fish to the desired position with a certain speed. Since the shape of the fish body can be geometrically changed by joints' bias, the fish can be viewed as a deformable body and has nonlinear motion, i.e., the body oscillates. The fuzzy logic method resembles human decision-making ability to generate useful solutions based on approximate information, which may be an answer to the fish's orientation control.

IV. Speed Control Algorithm

When a robotic fish swims in water, the regulation of its body speed at the center of mass is realized by mainly changing the servomotors' oscillating frequency. There are some unfavorable factors against the robotic fish's speed control. On the one hand, the interactions between the fish and surrounding water will result in resonance at a certain frequency, accompanying with the robot's rolling along its body axis and yawing along the axis perpendicular to the water surface. On the other hand, the fish cannot stop immediately even if the speed of each joint drops to zero duo to drag, which allows momentum to be bled out of the system. The inertia forces and hydrodynamic forces, in this case, will jointly allow the fish with the stable shape to drift a short distance along the current direction. Without full understanding of hydrodynamic effects on swimming fish, how to find a tradeoff between the swimming speed and the hydrodynamic force then become a central issue for the fish's smooth motion.

As exploited in elevator control, an acceptable tradeoff between speed and stress (potential energy) can be achieved by carefully manipulating the moving speed. The fish's inertia force can be restricted by setting the maximum acceleration to a value of A_m when swimming from a stationary state to the maximum steady speed V_f, at which the rolling and yawing of the fish body are minimum. Steady speed V_f, can be determined through a lot of experiments. To ensure the fast and steady motion, the robotic fish should reach its best ability, i.e., achieving at $v = V_f$, where v denotes the body speed of a running robotic fish. Therefore, the fish should accelerate to $v = V_f$ as soon as possible by holding the acceleration A_m. When the distance between the fish and the goal is less than some threshold, it begins to decelerate with a maximum deceleration $-A_m$ by gradually decreasing oscillating frequency f. Finally, the fish straightens itself and drifts toward the goal with zero-joint-speed, where zero-joint-speed means that all joints stop oscillating and that the corresponding body speed is necessarily not in zero due to the inertial forces and hydrodynamic effects. Notice that the choice of "drift to goal" is a makeshift when the precise hydrodynamic model is not available now, and that if dynamics control is well adopted, it is possible to turn off controller because the error would be zero.

With the assumption that the centers of mass and buoyancy coincide with the origin of the fish coordinate system, the speed profile will be piecewise in terms of the distribution function given by

$$v(t)=\begin{cases} 0.5V_f\left(1-\cos\left(\frac{\pi t}{T}\right)\right) & (0<t\leq T) \\ V_f & (T<t\leq T_d) \\ 0.5V_f\left(1-\cos\left(\frac{\pi(t-T_d-T)}{T}\right)\right) & (T_d<t\leq T_d+T) \\ 0 & (t>T_d+T) \end{cases} \tag{7}$$

where $T = (\pi/2)(V_f/A_m)$, and A_m can be predefined experimentally. A S-shaped motion process, as shown in Fig. 6, can clearly be divided into four phases: the acceleration phase, the constant phase, the deceleration phase, and the drift phase.

Let l be the distance between the fish and the goal, L_d be decelerating threshold, L_s be stopping oscillation threshold, and $L_d > L_s$. Some specified cases in the speed control, using hybrid strategy, have to be further elaborated on below:

1) If $l > L_d$, the fish keeps accelerating ($v = v + \Delta v$) until $v = V_f$.
2) If $L_s < l < L_d$ and $v > V_s$, the fish keeps decelerating ($v = v - \Delta v$) until $v = V_s$, where V_s indicates a nonzero low speed, e.g., $V_s = V_f/3$.
3) If $l < L_s$, the fish stops oscillation and straightens itself, then drifts to its goal.
4) If V_f is not reached, the overall distance is not enough to complete the full acceleration, deceleration and drift

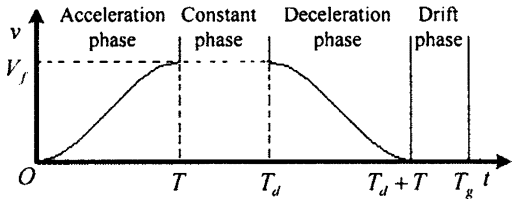


Fig. 6. Speed profile of the robotic fish.

phases, e.g., the initial l meets $l_{\text{initial}} < L_d$ or even $l_{\text{initial}} < L_s$, the fish will approach the destination directly at $v = V_s$, where the distance is too short to accomplish the effective control and the overshoot often occurs.

5) If the target point is overshot, i.e., the distant error oversteps the permitted bounds, turning mode B or C is used to change the heading of the fish and then the above strategy according to the current value of l is taken again. Meanwhile, when the overshoot occurs, the value of L_s should be changed according to

$$\text{If } |\delta_l| > \varepsilon |r_k|, \text{ then } L_s = L_s - \varepsilon \delta_l \tag{8}$$

where δ_l is assumed to be a typical overshoot/undershoot, r_k is the reference, i.e., the initial distance between the fish and the destination, and ε $(\varepsilon > 0)$ is the percentage allowed for the steady error. If an undershoot $(\delta_l < 0)$ occurs, L_s will increase by $-\varepsilon\delta_l$; if an overshoot $(\delta_l > 0)$ works, L_s will decrease by $\varepsilon\delta_l$. Strictly speaking, a precise goal-reaching is hard to achieve due to unpredictable hydrodynamic effects, and an undershoot or overshoot easily arises during approaching destination.

For a desired speed, a proportional-integral-derivative (PID) controller is designed, whose structure is illustrated in Fig. 7, where V_{set} denotes the desired speed, f_{set} represents the expected oscillating frequency derived from speed-frequency function $f_1(v)$, fe indicates the error of the oscillating frequency, v is the body speed of the fish, which is measured by an auxiliary visual subsystem, and f is the feedback oscillating frequency derived from the speed-frequency function $f_2(v)$. For simplicity, $f_1(v)$ and $f_2(v)$ take the same form, which can be fitted by a linear function using an abundance of experimental data given in Table II. It should be noticed that, in particular, when the error fe nears to zero, the PID controller will not work.

V. Orientation Control Using Fuzzy Logic

As discussed in Section III, the robotic fish can navigate to a desired position at a certain speed by choosing four proper joint angles $\{\phi_{i1}, \phi_{i2}, \phi_{i3}, \phi_{i4}\}$. The deflections of the joint angle are added to the first two joint angles $\{\phi_{i1}, \phi_{i2}\}$ so that the fish can turn with different turning radius. The key issue then becomes how to choose suitable deflections in response to environmental changes, which is clearly a hard nut to crack owing to un-modeled uncertainty in fish's motion. Luckily, there is an increasing tendency to use fuzzy logic controllers (FLC) to resolve an issue of un-modeled uncertainty. The mechanism of a FLC is that the uncertainty is represented by fuzzy sets and an action is generated cooperatively by several rules that are triggered to some

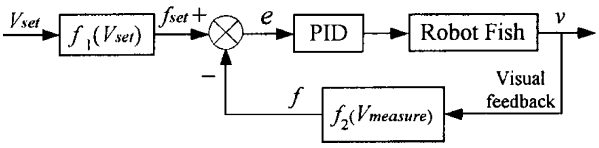


Fig. 7. Structure of PID controller for a desired speed.

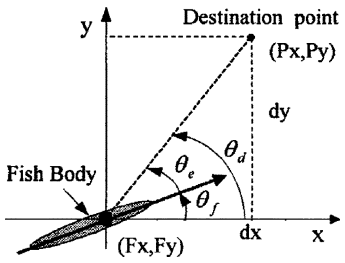


Fig. 8. Control inputs.

degree, and smooth and robust control outputs are finally produced. The difficulties in designing a FLC are the setting of parameters of membership functions and the composition of fuzzy rules. Since the fuzzy controllers are able to resemble human's decision-making to a certain extent, compared to the traditional control paradigm, the advantages of fuzzy control paradigms are twofold. A mathematical model of the system to be controlled is not required, and a satisfactory nonlinear controller can often be developed empirically without complicated mathematics. The core value of these advantages is the practicality, leading to less system development time and cost [29]. Attracted by the merit, the FLC is chosen here for the fish's orientation control. The objective is to build a FLC that generates the deflections of the first two joint angles when the fish moves from any initial position to its final position. Essentially, different joint deflections lead to different turning radiuses, and in turn lead to different angular speeds. With a proper angular speed, the desired heading of the fish can be achieved. In some sense, the FLC for the orientation control is reckoned as a quantitative use of turning mode A.

Initially introduced by Zadeh, fuzzy logic implements classes or groupings of data with boundaries that are not sharply defined (i.e., fuzzy). Any methodology or theory implementing "crisp" definitions such as classical set theory, arithmetic, and programming, may be "fuzzified" by generalizing the concept of a crisp set to a fuzzy set with blurred boundaries. A typical FLC works in a similar way to a conventional controller: it accepts an input value, performs some calculations, and generates an output value. This process is called the fuzzy inference process that primarily works in three stages:

1) fuzzification, where a crisp input is translated into a fuzzy value;
2) rule evaluation, where the fuzzy output truth values are computed;
3) defuzzification, where the fuzzy output is translated to a crisp value [30], [31].

The fuzzy orientation function control inputs are shown in Fig. 8. The three state variables Fx, Fy, and θ_f determine the current fish position, which are defined by the vision subsystem. θ_f specifies the angle of the fish with respect to the horizontal, and the coordinate pair (Px,Py) denotes the position of the destination point of the fish which may be the real position of a

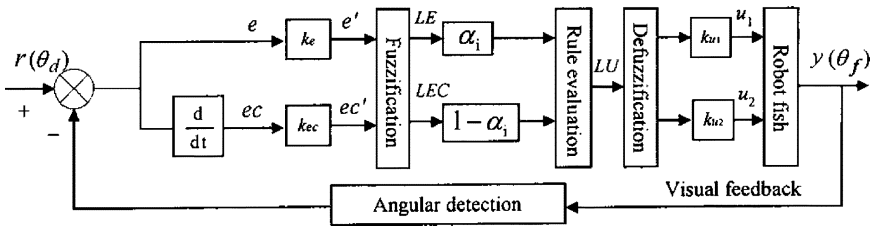


Fig. 9. Structure of FLC for orientation control.

ball or the location of a hole. The FLC will calculate the corresponding joint deflections in real-time during moving toward the destination.

First of all, to develop a FLC the input and output parameters must be defined. Fig. 9 shows the structure of FLC for orientation control, which takes two inputs and produces two outputs. Suppose that in a certain instant p, the values for error and its change are e and ec, respectively, which serve as the inputs described by $e_p = r_p - y_p$ and $ec_p = e_p - e_{p-1}$, where r indicates the desired input, namely the desired angle θ_d that the fish should face every instance when to approach the destination, y denotes the measured angle by visual subsystem, i.e., the fish's current angle θ_f, and especially e_p is equivalent to θ_e shown in Fig. 8. The outputs of the controller are the deflections of the first two joints: u_1 and u_2, which will be used for various orientation adjusting. During the fuzzification and rule evaluation, the same membership function and fuzzy rules are applied to u_1 and u_2, i.e., $u = [u_1, u_2]$. In the process of the defuzzification where they are multiplied by different scaling factors: k_{u1} and k_{u2}, respectively. The ranges of the input and output variable values determined by experiments are as follows: $-30° \le e \le 30°$, $-60° \le ec \le 60°$, $-240 \le u_1 \le 240$ and $-200 \le u_2 \le 200$. The value of e can be positive or negative, which a positive value signifies that the fish turns right otherwise the fish turns left. The universes of discourse of the controller variables are expressed as E, EC and U, respectively, which are all graded into 13 levels from -6 to 6.

The next step of developing a FLC is to represent the fuzzy set variables into linguistic terms, which are used to describe the control system's behavior. This means to name the linguistic labels covering that universe, and to specify the membership function associated to each label. The number of linguistic terms for each linguistic variable is 7, which can be labeled as NB (negative big), NM (negative medium), NS (negative small), ZE (zero), PS (positive small), PM (positive medium), and PB (positive big). That is

$$LE = LEC = LU = \{NB, NM, NS, ZE, PS, PM, PB\} \tag{9}$$

where LE, LEC, and LU are fuzzy variable sets associated with linguistic variables E, EC, and U, respectively. These labels are set of overlapping values represented by trapezoidal or triangular shaped that are called fuzzy membership functions. The range of values for each of these labels in the membership function can be determined by actual experiments. Through actual turning test, the membership functions for the inputs (e and ec) in FLC are determined, and a triangular shaped membership function for output variable U is also defined. It should be noted that the shape and the region of each membership function is able to alter by reassigning its grade distribution combined with several experiments.

The following step in FLC design is to specify the fuzzy rules that can be represented and stored by fuzzy associative memory (FAM) matrix that gives fuzzy rules of the inference engine. The size of a FAM matrix is completely dependent on the number of input fuzzy sets of the system. A 2-D (7×7) FAM matrix here is formulated, which can be explained as antecedent-consequent pairs or IF-THEN statements.

If e is LE_i and ec is LEC_j, then u is c_{ij} ($i = 1, 2, \ldots, 7$ and $j = 1, 2, \ldots, 7$) where the subscript i and j denotes the $i - th$ rule of fuzzy set LE and the $j - th$ rule of fuzzy set LEC respectively, c_{ij} represent the point of minimum fuzziness in the consequent part of the rules, i.e., the membership function centers.

To make all this work together, an inference mechanism that generates the output signal is necessary. The activation of the $i \times j$-th rule triggered by an input containing the error e and its change ec, is then calculated by Mamdani inference with min for intersection and max for union. Suppose that in a certain instant p, the error and its change are e_p and ec_p, respectively, the firing strength of a fuzzy control rule will be given by

$$f_{ij} = \min\left(\mu_{LE_i}(e_p), \mu_{LEC_j}(ec_p)\right) \tag{10}$$

where μ_{LE_i} and μ_{LEC_j} represent the membership functions of the linguistic values LE_i and LEC_j, respectively.

At the defuzzificationstep, as in (11), a calculation method called the center of area (COA) is used in order to produce the crisp output value of u. In reality, after the crisp value is multiplied by scaling factor (ku_1 ku_2), the outputting angular variation of the first two servomotors is obtained

$$u_{ij} = \frac{\sum\limits_{i,j} f_{ij} c_{ij}}{\sum\limits_{i,j} f_{ij}} \tag{11}$$

In addition, seen from the Fig. 9 again, scaling factors, k_e, k_{ec}, k_{u1}, and k_{u2}, which are associated to linguistic variables, are contained in the detailed structure of FLC. Hence, the input and output of the controller are changed proportionally. Their role is to tune FLC to obtain the desired dynamic properties of the process loop. The scaling factors here determined by experiments are as follows: $k_e = 6/30 = 0.2$, $k_{ec} = 6/60 = 0.1$, $k_{u1} = 240/6 = 40$ and $k_{u2} = 200/6 = 33.3$. Notice that these parameters will have to be returned for each system to which the algorithm is applied.

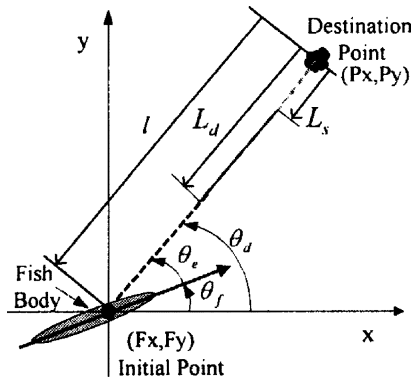


Fig. 10. Decomposition of the PTP control.

VI. PTP Control Algorithm

In the former section, we have discussed some algorithms for the speed and orientation control of our robotic fish prototype. In this section, we will explore the implementation of steering the fish from an arbitrary initial position to a destination point in 2-D Euclidean space. Notice that the idea of the orientation control is achieved by continuously reducing the angular error in an average sense at the assumption that a possible path connecting the initial point and destination point is a straight line. Also, the fish can only averagely be driven in a straight line by changing the oscillating frequency. Combined the speed control and orientation control, a straight-line-based motion will be consequently achieved.

In order to realize the PTP control of the robotic fish, the strategy we choose is to get rid of the error of the orientation between the fish body and the line from the initial point (Fx,Fy) to the destination point (Px,Py) while advancing along the specified line. An ideal PTP unit position vector (V_{PTP}) is given by

$$V_{PTP} = \frac{1}{\sqrt{(\mathrm{Px}-\mathrm{Fx})^2+(\mathrm{Py}-\mathrm{Fy})^2}}\begin{bmatrix}\mathrm{Px}-\mathrm{Fx}\\ \mathrm{Py}-\mathrm{Fy}\end{bmatrix}. \quad (12)$$

As specified in Fig. 8, the coordinate pair (Px,Py) specifies the destination of the fish, and the coordinate pair (Fx,Fy) denotes the current position of the fish. These position and orientation variables are all provided by the vision subsystem.

On the basis of hybrid control, as shown in Fig. 10, different strategies are chosen according to different distance (l) between the fish body and the destination point. The measure being taken is from crude to fine. If $l > L_d$, the fish speeds up to approach the destination; If $L_s < l < L_d$, when the fish is in motion, accurate control is employed, that is, it slows down and approaches within a certain orientation error; otherwise, it approaches with a nonzero low speed V_s; If $l < L_s$, when the fish is in motion, it stops oscillation and straightens itself, then drifts onwards with zero-joint-speed; otherwise, it approaches at $v = V_s$. Once overshot, special measures described in Section IV will be taken.

To facilitate the subsequent description, three basic actions of the fish will first be defined as follows.

1) SET-STRAIGHT—all links move to their initial zero-positions which are in a line with the forebody, namely, the fish straighten itself to make its whole body be a line.
2) SNAP-TURNING—it is used to achieve large-amplitude fast-turn.
3) REST-TURNING—turning-from-rest mode is applied to the overshoot cases.

At the same time, to make a good use of fish's excellent maneuverability exhibited by snap turning, an absolute angular threshold θ_t is specified. When the absolute value of θ_e oversteps the bounds of angular threshold, SNAP-TURNING works promptly. The detailed algorithm is then illustrated as follows.

PTP CONTROL ALGORITHM

Step 0) Initialize the environment and the destination point (Px,Py), and let fish SET-STRAIGHT.

Step 1) Update position & orientation information of both the fish and its environment obtained from the overhead camera, and calculate the orientation error θ_e and the distance error l relative to the destination point.
If the terminating condition is met, i.e., $fabs(\theta_e) < \delta e$ AND $fabs(l) < \delta l$, where $fabs()$ represents absolute value of a function, δe and δl are absolute error for θe and l respectively, then the algorithm exits and the fish is to be SET-STRAIGHT.
Otherwise, go to Step 2.

Step 2) Combine FLC for orientation control and SNAP TURNING to plan the fish's orientation strategy according to the value of θ_e.

I. Get current θ_e, if $fabs(\theta_e) < \theta_t$, go to II; otherwise, go to III.

II. Compute error of change ec, and determine the angular bias of the first two servomotors in terms of θ_e and ec, then go to Step 3.

III. Determine the use of SNAP-TURNING.
1) If $\theta_e < -\theta_t$, perform right SNAP-TURNING. Go to Step 4.
2) If $\theta_e > -\theta_t$, perform left SNAP-TURNING. Go to Step 4.

Step 3) Call the speed control algorithm to plan the fish's speed strategy according to the value of l.

I. Get the fish's current distance error l and the current speed v,

judge whether the fish is in overshoot. If no, go to II; otherwise, go to IV.

II. Decide the desired speed V_{set} according to l and v.

1) If $l > L_d$, the fish keeps accelerating ($V_{\text{set}} = v + \Delta v$) until $V_{\text{set}} = V_f$. Go to III.

2) If $L_s < l < L_d$, check if $v > V_s$.

1) If true, let $V_{\text{set}} = v - \Delta v$ until $V_{\text{set}} = V_s$. Go to III.

2) If false, let $V_{\text{set}} = V_s$. Go to III.

If $l < L_s$, check if $v = 0$.

1) If true, i.e., the fish is in start state, let $V_{\text{set}} = V_s$. Go to III.

2) If false, the fish stops oscillation and straightens itself (to be SET-STRAIGHT), then drifts to goal. Go to step 4.

III) For a given V_{set}, the PID controller is used to derive the desired oscillating frequency f. Then, go to step 4.

IV) If overshot, REST-TURNING is used to change the heading of the fish. Meanwhile, L_s is adjusted according with (8). Go to step 4.

Step 4) Translate the above results into fish's control parameter set $\{\phi_1, \phi_2, \phi_3, \phi_4, f\}$, then send them to the fish through the radio control module, and go to Step 1.

Based on the above algorithm, a steering function *MoveToGoal* $(destpt)$ is developed and applied to the following tests, where $destpt$ denotes the destination point (Px,Py).

VII. Experimental System and Results

To verify the feasibility and reliability of the proposed algorithms, an experimental robotic fish system has been constructed. The system, as depicted in Fig. 11, consists of four subsystems: the robotic fish subsystem, the vision subsystem, the decisions-making subsystem, and the communication subsystem. All aquatic experiments presented in this paper were carried out in a 2000 mm × 1150 mm pond with still water. The information of the fishes and their surroundings captured by overhead CCD camera is effectively processed and sent to the decision-making module as an input, and then the output of the decision-making subsystem is transmitted to the robotic fish through the communication subsystem with a baud rate of 9600 bits/s. However, effective radio communication is difficult

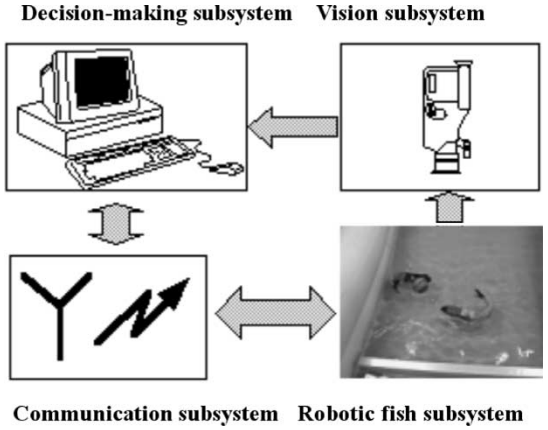


Fig. 11. Experimental system configuration.

to pursue due to unfavorable effects from aquatic surroundings, e.g., splashed waves.

In the experiment, robotic fishes, a ball, and obstacles are marked with specified colors. To locate the robotic fish and other objects quickly and accurately, a parallel algorithm for visual tracking based on color information has been developed [32], mainly by adaptive segmentation and a closure operation. Using the vision-based tracking system to provide real-time feedback, two experiments with a robotic fish were designed to test the proposed control strategies.

Experiment A: Playing Ball: In a pond with still water, a floating ball, 45 mm in radius, was used as a target. The robotic fish was controlled to approach the ball from an arbitrary initial position and orientation. The fish status $(\text{Fx}, \text{Fy}, \theta_f)$ and the ball position (Px,Py) were located by the overhead camera. By calling the steering function *MoveToGoal* $(destpt)$ continuously, where $destpt = (\text{Px}, \text{Py})$, the fish intentionally swam toward the ball, and sometime pushed it. Because the ball was too light to remain stationary, the fish lost it and pushed it again just like playing a game. This can be considered that the fish tracked the floating ball continuously. Fig. 12(a) shows a photo of an experimental scenario during playing-ball, Fig. 12(b) shows a moving trajectory of the fish swimming toward a ball, where the positions of the fish and the ball are denoted in image plane coordinates in which the whole view field is regarded as a plane with 320 × 240 pixels. Fig. 12(c) shows the corresponding orientation error θ_e. Notice that the pond is not large enough to remove the effects of the reflective waves at present, so the positions of the ball and the fish will be slightly varied with the disturbances. Notice also that the range of the orientation error toward the end of the experiment seems much larger than the induced body oscillation resulting from the tail motion. Since as mentioned before, the body may not be moving in the direction that the head is pointing, when the fish approaches the ball near, it has to decelerate and move at a low speed, so the sampled direction of the fish may not be the true direction of moving. In addition, lacking of mechanical symmetry during the orientation control, the fish has a large roll motion as it swims at a low speed. Therefore, the fish seems to be un-steadier than it does at a steady speed V_f. For these reasons, the actual orientation error is larger than the expected value. If the experiment is done with a larger pool, more confident results will be achieved.

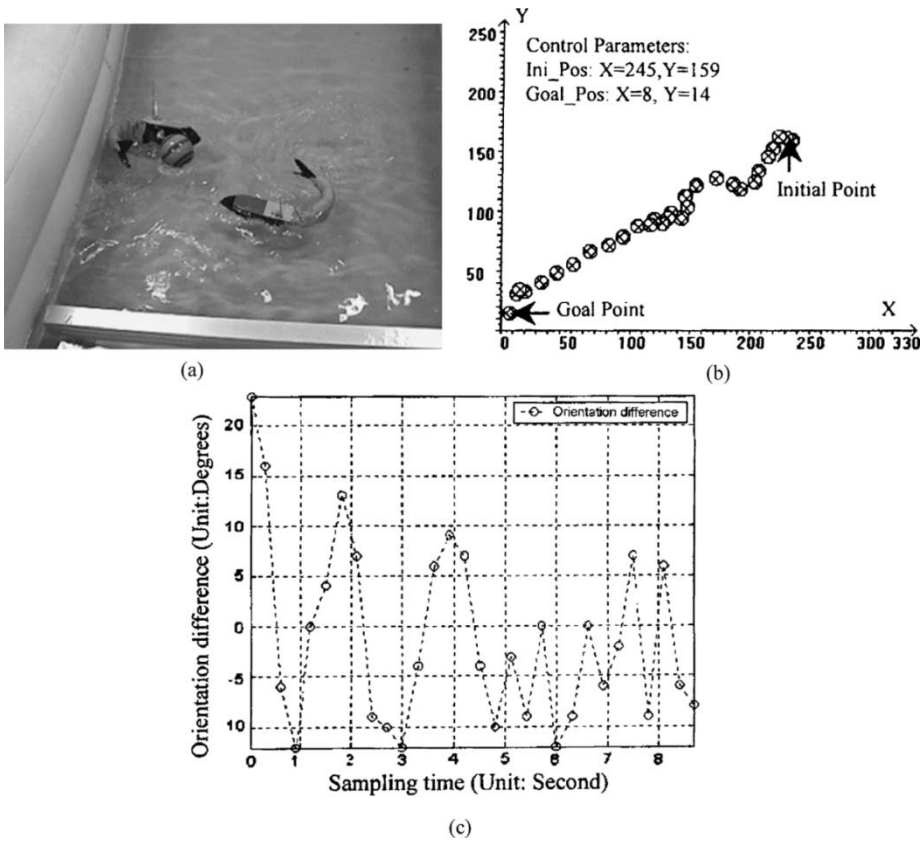


Fig. 12. (a) Scenario of playing ball. (b) Moving trajectory. (c) Orientation error θ_e.

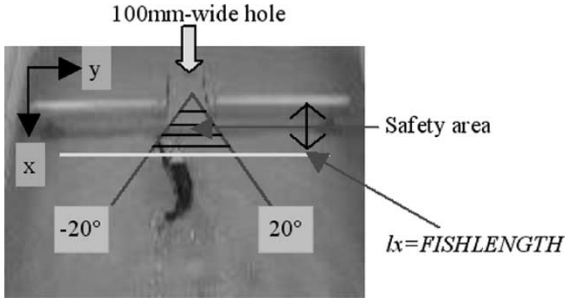


Fig. 13. Scenario of passing the hole.

Experiment B: Passing a Hole: To test controllability of the robotic fish in a narrow space, as shown in Fig. 13, two bars marked with the predefined color are aligned in a line to form a "HOLE" with a clearance of 100 mm. For the robotic fish, its task was to pass through the hole from an arbitrary initial position and orientation. The fish status $(\mathrm{Fx}, \mathrm{Fy}, \theta_f)$ and the hole position (Hx,Hy) were located by the overhead camera.

Before navigating the fish through the hole, a distant variable lx was defined as $lx = \mathrm{Fx} - \mathrm{Hx}$. When the fish was far from the hole, i.e., $lx > FISHLEGTH$, the steering function *MoveToGoal* $(destpt)$ was called continuously, where $destpt = (\mathrm{Hx}, \mathrm{Hy})$, so that the fish gradually swam toward the hole. When the fish was near the hole, that is, $FISHFOREBODY < lx < FISHLEGTH$, the orientation error θ_e in degrees was checked to see whether it lies between -20 and 20. If yes, let the fish straighten itself and move forward; otherwise, called the steering function to adjust its orientation till it met the requirement. When the fish was in the hole, namely, $-FISHFOREBODY < lx < FISHFOREBODY$, let the fish swim with full-speed V_f. After the fish passed through the hole, i.e., $lx < -FISHFOREBODY$, the above mentioned algorithm was repeated to make the fish pass from the other side of the bar again. Here, the constant $FISHLEGTH$ denotes the length of the fish, and $FISHFOREBODY$ indicates the length of the forebody. To make it more intelligible, as shown in Fig. 13, a triangular safety area satisfying the requirements of lx and θ_e is defined. In this area, the fish moves straight and is in a stage of "on;" otherwise, the fish is in a stage of "off" and has to call the steering function to adjust itself till it enters into the "on" stage. The slim robotic fish, by means of simple continual "ON-OFF" control in obstacle avoidance, can successfully get across a narrow gap. An image sequence of passing the hole is demonstrated in Fig. 14. Compared to other obstacle-avoidance methods such as the *potential field* technology [33] and the *distance transform method* [34], where the robot is often represented as a point in configuration, our proposed triangular "on-off" control can be especially applied to the slim-shaped robot's collision-free. Of course, once the fish integrated multiple various sensors is put into practice in future, more advanced local path-planning method considered the fish's shape will be further investigated.

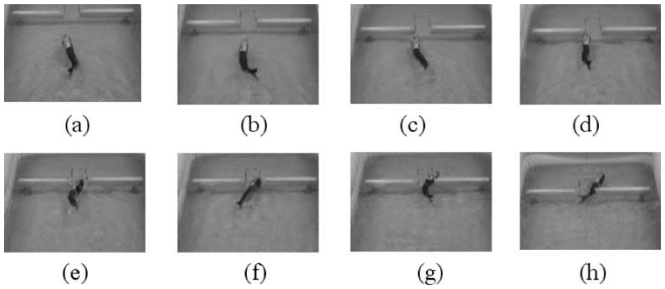


Fig. 14. Image sequence of passing hole [from (a) to (h)].

Discussions: By calling PTP control algorithm, at present, the fish can play with a ball and pass a hole in the pond. But, it is observed from Fig. 12(c) that the orientation error, between -25° and 25°, does not appear to limit to zero. This implies that it has nonlinear motion, that is, the fish body oscillates continuously during moving. To some extent, the dynamic imbalance of gravitation and buoyancy at the flapping tail affects propulsive performance and steadiness, which is limited by its intuitive oscillation propulsion mode. An additional pectoral mechanism may help the fish body acquire steadiness.

In this paper, both the hybrid speed strategy and the FLC for orientation control captured from the empirical models are nonlinear and partly effective. The FLC can be viewed as a practical, simple and intuitive way to incorporate nonlinear characteristics to the system, but the suitable membership functions are hard to determine empirically in the experiment because of the effects of both added mass and surface waves. For the robotic fish, a wide and large experimental locations as well as self-positioning ability are preferable to robust motion control.

VIII. Conclusions and Future Work

An experimental closed-loop control system for a 4-link and free-swimming biomimetic robotic fish has been developed based on the proposed simplified propulsive mode for carangiform swimming. The fish's motion control task is decomposed into online speed control and orientation control, and corresponding algorithms are implemented on the actual system. The experimental results have demonstrated the good performance of the robotic fish using vision-based positional feedback. However, our proposed motion control algorithms did not take fish dynamics into account, only the fish's motion kinematics was considered, which has to be improved in the future.

Further research will focus on the development of expanded closed-loop control, and a full planar motion-planning algorithm for complex and cluttered environments based on visual feedback. At the same time, some sensors (visual, ultrasonic and infrared detectors) are planning to be embedded into the fish body so that the robotic fish is able to react to the change in the environment and have certain local autonomy. A new degree of freedom (up/down) is also planned to add into the robotic fish by using pectoral-fin mechanism, which enables the robotic fish navigate in a 3-D workspace. Eventually, an autonomous robotic fish that can swim skillfully (high efficiency) and intellectually (autonomous obstacle avoidance with on-board sensors) will be realized.

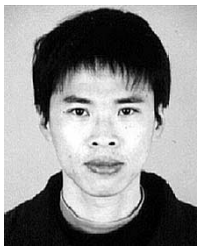
Acknowledgment

The authors would like to thank the referees for their careful reading of the manuscript and helpful comments.

References

[1] M. S. Triantafyllou and G. S. Triantafyllou, "An efficient swimming machine," *Sci. Amer.*, vol. 272, pp. 64–70, Mar. 1995.
[2] J. M. Anderson, M. S. Triantafyllou, and P. A. Kerrebrock, "Concept design of a flexible-hull unmanned undersea vehicle," in *Proc. 7th Int. Offshore Polar Engineering Conf.*, May 1997, pp. 82–88.
[3] M. Mojarrad, "AUV biomimetic propulsion," in *Oceans Conf. Rec.*, Sept. 2000, pp. 2141–2146.
[4] M. Sfakiotakis, D. M. Lane, and J. B. C. Davies, "Review of fish swimming modes for aquatic locomotion," *IEEE J. Oceanic Eng.*, vol. 24, pp. 237–252, Apr. 1999.
[5] B. G. Tong, "Propulsive mechanism of fish's undulatory motion," *Mech. Eng.*, vol. 22, no. 3, pp. 69–74, 2000.
[6] J. Gray, "Studies in animal locomotion. VI. The propulsive powers of the dolphin," *J. Exp. Biol.*, vol. 13, pp. 192–199, 1936.
[7] C. M. Breder, "The locomotion of fishes," *Zoologica*, vol. 4, pp. 159–256, 1926.
[8] G. Taylor, "Analysis of the swimming of long narrow animals," in *Proc. R. Soc. Lond. A*, vol. 214, 1952, pp. 158–183.
[9] T. Y. Wu, "Swimming of a waving plate," *J. Fluid Mech.*, vol. 10, pp. 321–344, 1961.
[10] M. J. Lighthill, "Note on the swimming of slender fish," *J. Fluid Mech.*, vol. 9, pp. 305–317, 1960.
[11] ——, "Aquatic animal propulsion of high hydromechanical efficiency," *J. Fluid Mech.*, vol. 44, pp. 265–301, 1970.
[12] ——, "Large-amplitude elongated-body theory of fish locomotion," in *Proc. R. Soc. Lond. B*, vol. 179, 1971, pp. 125–138.
[13] J. Y. Cheng and R. Blickhan, "Note on the calculation of propeller efficiency using elongated body theory," *J. Exp. Biol.*, vol. 192, pp. 169–177, 1994.
[14] K. Hirata, "Development of experimental fish robot," in *Proc. 6th Int. Symp. Marine Engineering*, 2000, pp. 711–714.
[15] J. Czarnowski, R. Cleary, and B. Creamer, "Exploring the possibility of placing traditional marine vessels under oscillating foil propulsion," in *Proc. 7th Int. Offshore Polar Engineering Conf.*, Honolulu, HI, May 1997, pp. 76–82.
[16] D. Barrett, M. Triantafyllou, D. K. P. Yue, M. A. Grosenbaugh, and M. J. Wolfgang, "Drag reduction in fish-like locomotion," *J. Fluid Mech.*, vol. 392, pp. 183–212, 1999.
[17] D. Barrett, M. Grosenbaugh, and M. Triantafyllou, "The optimal control of a flexible hull robotic undersea vehicle propelled by an oscillating foil," in *Proc. IEEE AUV Symp.*, pp. 1–9.
[18] K. A. Harper, M. D. Berkemeier, and S. Grace, "Modeling the dynamics of spring-driven oscillating-foil propulsion," *IEEE J. Ocean. Eng.*, vol. 23, pp. 285–296, 1998.
[19] S. D. Kelly and R. M. Murray, "Modeling efficient pisciform swimming for control," *Int. J. Robust Nonlinear Control*, vol. 10, pp. 217–241, 2000.
[20] R. Mason and J. Burdick, "Experiments in carangiform robotic fish locomotion," in *Proc. Int. Conf. Robotic Automation*, 2000, pp. 428–435.
[21] K. Morgansen, V. Duindam, R. Mason, J. Burdick, and R. Murray, "Nonlinear control methods for planar carangiform robot fish locomotion," in *Proc. Int. Conf. Robotic Automation*, 2001, pp. 427–434.
[22] N. Kato and M. Furushima, "Pectoral fin model for manuever of underwater vehicles," in *Proc. IEEE AUV Symp.*, pp. 49–56.
[23] K. Hirata, T. Takimoto, and K. Tamura, "Study on turning performance of a fish robot," in *Proc. 1st Int. Symp. Aqua Bio-Mechanisms*, 2000, pp. 287–292.
[24] F. Bullo, N. E. Leonard, and A. D. Lewis, "Controllability and motion algorithms for underactuated Lagrangian systems on Lie groups," *IEEE Trans. Automat. Contr.*, vol. 45, pp. 1437–1454, Aug. 2000.
[25] S. Saimek and P. Y. Li, "Motion planning and control of a swimming machine," *Int. J. Robot. Res.*, vol. 23, pp. 27–54, 2004.
[26] M. Borgen, G. Washington, and G. Kinzel, "Introducing the Carangithopter: a small piezoelectrically actuated swimming vehicle," in *Proc. Adaptive Structures Material Systems Symp., ASME Int. Congress Exposition*, 2000, pp. 247–254.
[27] J. R. Hunter and J. R. Zweifel, "Swimming speed, tail beat frequency, tail beat amplitude and size in jack mackerel, Trachurus symmetricus, and other fishes," *Fish. Bull.*, vol. 69, pp. 253–266, 1971.

[28] P. Domenici and R. W. Blake, "The kinematics and performance of fish fast-start swimming," *J. Exper. Biol.*, vol. 200, pp. 1165–1178, 1997.
[29] H. Ying, "Fuzzy systems technology: a brief overview," *IEEE Trans. Circuits Syst. Soc. Newsletter*, vol. 11, pp. 28–37, 2000.
[30] J. Zhu, *Mechanism and Application of Fuzzy Control*. Beijing, R.O.C.: Mechanical, 1995.
[31] H. Ying, *Fuzzy Control and Modeling: Analytical Foundations and Applications*. Piscataway, NJ: IEEE Press, 2000.
[32] J. Z. Yu, S. Wang, and M. Tan, "A parallel algorithm for visual tracking of multiple free-swimming robot fishes based on color information," in *IEEE Int. Conf. Robotics, Intelligent Systems, Signal Processing*, Hunan, China, Oct. 2003, pp. 359–364.
[33] O. Khatib, "Real-time obstacle avoidance for manipulators and mobiles," *Int. J. Robot. Res.*, vol. 5, pp. 90–98, 1986.
[34] R. Jarvis, "Distance transform based path planning for robot navigation," in *Recent Trends in Mobile Robots*, Y. F. Zheng, Ed. Singapore: World Scientific, 1993, vol. 11, Robotics and Intelligent Systems, ch. 1.



Junzhi Yu received the B.E. degree in safety engineering and the M.E. degree in precision instruments and mechanology from the North China Institute of Technology, Taiyuan, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 1998, 2001, and 2003, respectively.

He is currently a postdoctoral researcher with the Center for Systems and Control Peking University, China. His research interests include autonomous robots, embedded system, and intelligent information processing.

Min Tan received the Ph.D. degree in control theory and control engineering from Institute of Automation, Chinese Academy of Sciences, Beijing.

He is a Professor in the Lab of Complex Systems and Intelligent Science, Institute of Automation, Chinese Academy of Sciences. His research interests include multirobot system, advanced robot control, biomimetic robot, manufacturing system, and system reliability.

Shuo Wang received the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing.

He is an Assistant Professor in the Lab of Complex Systems and Intelligent Science, Institute of Automation, Chinese Academy of Sciences. His research interests include multirobot system and biomimetic robot.

Erkui Chen received the Ph.D. degree in control theory and control engineering from China University of Mining and Technology, Xuzhou.

He is a Vice Professor of the School of Communication and Control Engineering, Southern Yangtze University, Wuxi, China. His research interests include mobile robot and intelligent control method.