Fisho: a Cost-Effective Intelligent Autonomous Robot Fish

Pablo Romero and Berardi Sensale-Rodriguez
Instituto de Matematica y Estadistica
Facultad de Ingenieria, Universidad de la Republica
Julio Herrera y Reissig 565, Montevideo - Uruguay
E-mail : promero@fing.edu.uy, bsensale@fing.edu.uy

Diego Astessiano and Rafael Canetti
Instituto de Ingenieria Electrica
Facultad de Ingenieria, Universidad de la Republica
Julio Herrera y Reissig 565, Montevideo - Uruguay
E-mail : caladi@adinet.com.uy, canetti@fing.edu.uy

Abstract*—We developed a cost-effective fish-like Autonomous Underwater Vehicle (AUV) with a sophisticated intelligence. The robot fish, named *Fisho*, is able to avoid obstacles in a swimming pool, reflecting the usual behavior of a biological fish. As a general rule, a robot captures external stimulus by means of electrical sensors; *Fisho* has infrared sensors to find the distance to close objects and pressure sensors to estimate its vertical physical position (depth). Its brain is a small-sized 200-MHz Single Board Computer (SBC), that collects information from the sensors and sends decisions to electro-mechanical servos (in order to act on its tail and flippers, thus control its trajectory). *Fisho*'s behavior is signed by its velocity and the vibration mode of its tail. Inspired in real fish, we designed its intelligence with a three-layer Markov chain by means of a full characterization of tasks, purposes and fish-moods. Its design and physical structure is both non-expensive and simple, trying to emulate a real fish, whereas addressing all typical challenges in an AUV. In this paper we discuss *Fisho*'s physical implementation: sensing platforms, electro-mechanical structure and intelligence.

Keywords*— *AUV, robot-fish, electro-mechanics, task planning

I. INTRODUCTION

Unmanned underwater vehicles are usually classified into two categories: autonomous underwater vehicles or AUVs -which do not require human assistance-, and remotely operated underwater vehicles or ROVs. This work, discusses the physical implementation of a robot-fish, i.e. an AUV whose shape is based on the morphology of real fish. A highly remarkable work in this field is a fully autonomous robot fish developed at the University of Essex [1]. Although Liu et al. [1] developed a highly effective fish-like AUV, its implementation cost reached half a million dollars. Moreover, in spite that efficient propulsion methods [2], [3] and mechanical implementations [4], [5], [6] are highly available in the literature, these are not often applied into fully autonomous systems.

This paper gives a succinct description of *Fisho*, a fully autonomous cost-effective biologically-inspired intelligent robotic fish. The main features of its fish-like movement and behavior are: (a) its stochastic intelligence that reflects that from real fishes, and (b) its novel swimming mechanical implementation, which uses two joints in order to produce smooth tail movements. The design is simple, and its production cost is just a few hundred dollars.

In order to address a full implementation of a robot-fish several design aspects should be tackled: (i) its physical architecture, (ii) communications with the external world, (iii) highly sensible sensor platforms -suitable for underwater-, and (iv) dynamic motion based on kinematic equations to predict and decide its movement/trajectory. Biological inspiration is a primary need at the moment of design. The carangiform fish has been selected as a natural reference from previous studies of fish propulsion [7], [8] and by taking into account the trade-offs between implementation complexity and movement likeness. The swim style is based on a mechanical architecture with two degrees of freedom for the tail motion. Once the robot motion implementation was chosen, we selected suitable sensors for underwater environments -in order so that *Fisho* can discover its surrounding world-. Infrared sensors (IR) were employed to determine the presence and distance to close objects in the horizontal plane, whereas a relative-pressure sensor was selected to estimate its depth.

Fisho's novel stochastic intelligence was inspired in the behavior of real fish, defining tasks, moods and purposes through a three-level Markov chain, which imprints a highly dynamic behavior to the resulting prototype. Prior to the prototype construction, Matlab based simulations were carried-out inspired by Ref. [9]; this way, the control algorithms for the robot tasks were developed and tested. Given the angular functions for its joints, and the simulated sensor readings, the Matlab model can adjust the robot trajectory in "real time".

This paper is organized as follows. A description of the mechanical implementation and signal processing is discussed in Section II. The hardware architecture and stochastic intelligence are presented in Section III and IV, respectively. The results and Matlab simulation are addressed in section V. Finally, section VI presents our concluding remarks and discusses future work.

II. MOTION GENERATION AND MECHANICAL IMPLEMENTATION

From the three fish swimming styles (carangiform, ostraciform and anguiloform, see Ref. [7]), carangiform is the fastest and mechanically most efficient. For implementing this style, we developed a two joint realization as shown in Fig. 1. In this case, two actuators (servos) were employed, which control the tail and tail-fin movement by modifying its oscillation frequencies, amplitude and offset in relation to the fish body-axis. Each movement (straight swim, right turn, left turn) can be constructed by controlling θ_i as a function of time, i.e. $\theta_i = \theta_i(t)$, where θ_i is the relative angle of the i-th joint and t represents time. By means of employing sinusoidal control signals, it is possible to generate every movement by simply changing some signal parameters (see Ref. [10], [11]). For signals of the form:

$$\theta_1(t) = A_1 \sin(\omega t) + K_1 \quad , \qquad (1)$$

$$\theta_2(t) = A_2 \sin(\omega t) + K_2 \quad , \qquad (2)$$

when $K_1 = K_2 = 0$, the resulting movement becomes symmetric with respect to the body-axis, thus the fish swims straight. Turns can be achieved by inserting offsets with respect to the body axis. The other controllable parameters, i.e. oscillation amplitude (A_i) and frequency (ω), affect the fish linear speed - i.e. velocity in the direction of the fish displacement [10], [11]-. The larger the oscillation amplitude and frequency, the largest the linear speed. For the case of straight-line swimming, this can be easily understood from the mathematical expression for the fish terminal speed [7]:

$$U = f A / S_t \quad , \qquad (3)$$

where U is the fish terminal speed, f is the tail oscillation frequency and A the maximum tail oscillation amplitude. S_t is the Strouhal number, which is a constant (dependent on the fish geometry) and usually takes values between 0.25 and 0.35.

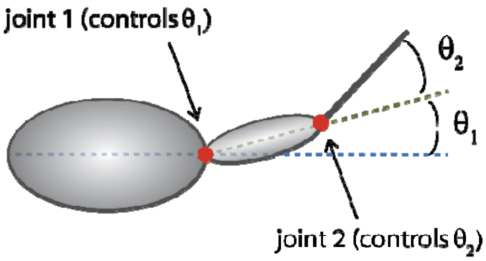

Fig. 1. Detail of the joint design, which has two controllable angles θ_1 and θ_2. Each joint is controlled with a servo-motor.

As discussed in Ref. [10], [11], large oscillation offsets (with respect to the body-axis) result on small turning radius. These results allow us to understand the generation of simple movements, which is a key element for the stimuli response as well as for the definition of the fish behavior. The tail mechanical implementation was performed using a novel yet simple mechanism, which produces smooth tail movements similar to those of real fish. Basically, the fish consists of an inner rigid skeleton containing the electric components and mechanical structure, and an external rubber skin (fish-like skin). The particularity of this design consists in that the fish skin is made from a unique very flexible piece, thus the tail movement results in nearly perfect oscillations when applying very small joint oscillations. Figures 2 and 3 depict the inner tail skeleton and a side view of *Fisho*, respectively.

III. SYSTEM ARCHITECTURE

The system consists of a Gumstix single board computer (SBC) based on an Intel XScale microprocessor at 200-MHz (in which the main programs are run), with a WI-FI expansion (in order to send sensor/stimuli data into an external PC and also debug codes), and with a robotics specific expansion which consists of an Atmel Atmega128 microcontroller that communicates via I2C with the main processor and has available input/output pins for the connection of sensors and motors.

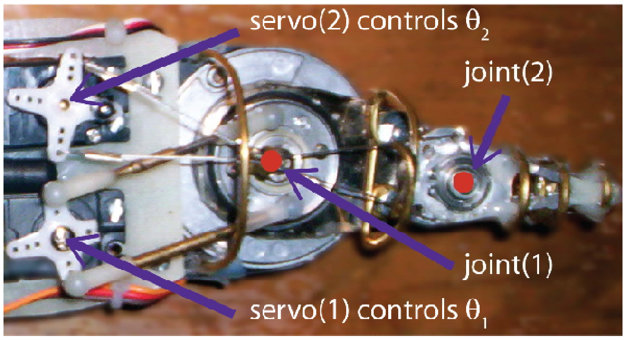

Fig. 2. Picture of the tail skeleton and joints, the main elements controlling the robot motion are highlighted.

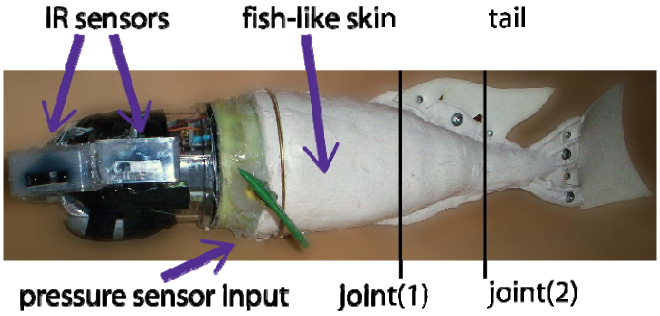

Fig. 3. Side view of *Fisho* and detail of the sensor locations.

In terms of motors, we used three servo motors, one for the depth control (implemented with lateral fins), and the other two for the tail motion control. The sensory system consists of five IR distance sensors (Sharp GP2XX series) and one relative pressure sensor (Freescale MVXP2010). IR sensor were used to determine the presence and distance to objects in the horizontal plane; the relative pressure sensor was employed in order to estimate its depth. The system is powered with two Ni-Mh battery banks, and has an autonomy of around one hour. Figure 4 presents a diagram of the hardware architecture.

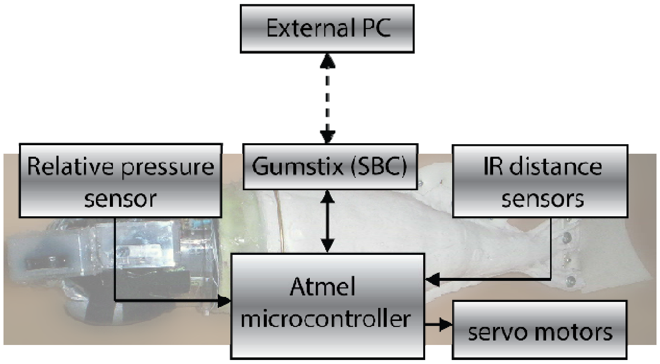

Fig. 4. Detail of the Hardware architecture.

IV. INTELLIGENCE

A fish performs several ***tasks*** during the day, with different purposes and moods. In this section, we will select characteristic tasks and discuss their representation in carangiform fishes in order to design *Fisho*'s intelligence.

As many other animals, it is well known that fishes sleep, even if they do not have eyelids to close. Surprisingly enough, some species need to keep swimming while they sleep (i.e. sharks), and fishes do not present REM patterns found in human beings. However, sleeping is a fundamental task for fishes in order to recover their energy and structural functions. For this reason, *Fisho* will be programmed to rest from time to time. We will use the term "*rest*", since our concept of sleeping in human beings is quite different than that of fish.

The other *tasks Fisho* will perform are mainly geometrical, and greatly inspired in the movements a fish usually undertakes when in an Aquarium. A typical movement is to follow a wall, that is, to swim along the borders of a closed environment. This task will be called "*follow wall*". Another usual feature in fishes is to make round-trips along a straight line, with regular returns of 180 degrees (not necessarily when reaching the border of the aquarium), we will call this task "*swim straight*". If we further relax this special movement pattern we can extend it to piecewise paths (straight lines), which we will call "*polygonal trajectory*" task. A limit case of this polygonal trajectory is a quite curious but rather frequent movement: to move in a circle; a task we will call "*swim in circles*". Finally, the fish will find one border of the aquarium with the "*find border*" task, and touch it with the "*touch border*" task. We are aware that these movements are not all the possibilities; however, we feel they faithfully represent the most usual ones. To summarize, we aim to represent the basic movements of fishes with seven tasks:

1) *Rest*
2) *Follow border*
3) *Swim straight*
4) *Polygonal trajectory*
5) *Swim in circles*
6) *Find border*
7) *Touch border*

Additionally, these tasks are executed with a certain ***purpose***. We will focus on survivability purposes: rest, feed and reproduction. A simple purpose characterized with a reduction in energy consumption is to *rest*. Fishes commonly *exhibit* themselves before mating, a purpose with a remarkable importance for species preservation. Clearly, a fish would not be able to survive without food. For the sake of simplicity, we will stick only to these three *purposes*, namely: "*rest*", "*exhibition*" and "*preying*".

Finally, each *task*/*purpose* pair can be executed with a different ***mood***. For instance, *Fisho* could trace circles with a predatory purpose to find food, in two modes: "*quiet*" (to confuse the prey) or "*excited*" (to catch the prey). The three-level intelligence design is then: ***task-purpose-mood***, and contains 7 *tasks*, 3 *purposes* and 2 *moods* ("*quiet*" and "*excited*"). Each tern (*task-purpose-mood*) is called a ***presentation***. Figure 5 illustrates the three-level design with tasks executed for a certain purpose and within a given mood.

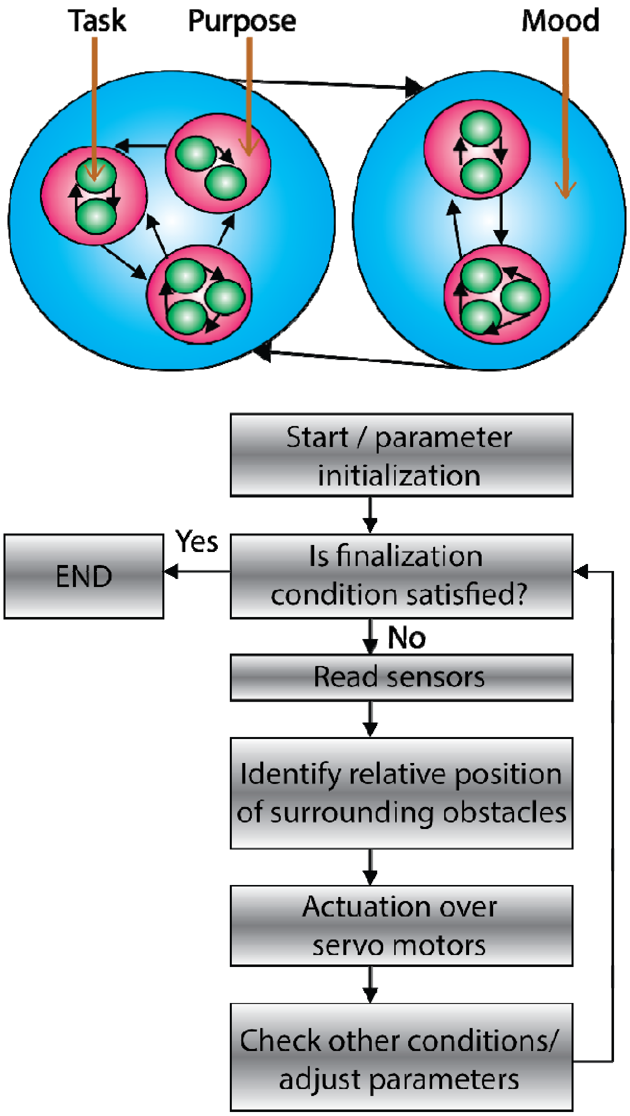

Fig. 5. (up) three-level design -where each level is characterized by a Markov Chain-, and generic form of the control loops (bottom). *Fisho*'s state is fully characterized by a presentation (task-purpose-mood).

It is clear that not all *presentations* are feasible. Moreover, they are not ordered deterministically in time when performed by a fish. As a consequence, we can define *Fisho*'s behavior by employing a three-layer discrete Markov Chain, where some presentations have null probability. A homogeneous discrete-time Markov Chain can be fully characterized by a transition matrix. *Fisho*'s *presentations* will switch in-between different homogeneous chains. More specifically, we will have transition matrices P_1 , P_2 and P_3

between the different *tasks* within the respective *purposes*. The *moods* do not alter the transition probabilities, but affect *Fisho*'s linear velocity. For completeness, one should define an additional transition matrix Q between the different *purposes*, and choose time slots to decide when to change *purposes*, *tasks*, and *moods*.

So far, a *mood* is easily understood by a change in velocity (velocity in "*excited*" mode is faster than that in "*quiet*" mode); transition probability matrices P_i between different *tasks* should be defined for each fish *purpose* $i \in \{1, 2, 3\}$, where i=1, 2, 3 represent the "*rest*", "*exhibition*" and "*preying*" purposes respectively. Provided that five seconds is time enough for the sensors to discover a new object and for the robot to respond in consequence, we define a time slot $T_t = 5$ seconds for transition between different *tasks*. The transition between different *purposes* was arbitrarily defined with a time slot $T_p = 15T_t$ in order to have a clear perception of the development of the *purpose*. The slowest transition is considered to be that between *moods*, for which we chose $T_m = 300$ seconds. After this time is elapsed *Fisho* either stays in the current *mood* state ("*quiet*" or "*excited*") or moves to another state with equal probability of 1/2.

Now, we are free to choose the different matrices Q and P_i. We start with the *purpose*-matrix Q, which contains the different transition laws between purposes. First, we chose its elements in the diagonal, to determine the expected sojourn time in each *purpose*. Let us assign equal expected sojourn time $T_0 = 4T_p$ for purposes 1 and 2 ("*rest*" and "*exhibition*"), whereas the "*preying*" purpose is assigned $T_0 = 2T_p$. Alternatively, the transition probabilities $Q = (q_{i,j})_{1\leq i,j\leq 3}$ follow $q_{1,1} = q_{2,2} = 3/4$ and $q_{3,3} = 1/2$. The matrix Q is stochastic (i.e. its rows must add to unity); the three degrees of freedom were fixed choosing a stationary distribution. Precisely, we assigned more weight to the "*exhibition*" purpose, whereas the purpose that we assumed less often to occur is the "*preying*" purpose. Mathematically, if the stationary vector $\pi = (\pi_1, \pi_2, \pi_3)^t : \pi Q = \pi$ is chosen so that $\pi = (1/2, 3/8, 1/8)$, then the local homogeneous Markov Chain given by Q is ergodic. By the ergodic theorem, *Fisho* stays in purpose 1 ("*rest*") 1/2 of the time, in purpose 2 ("*exhibition*") 3/8 of the time, and the remainder 1/8 of the time *Fisho* is acting as a predator. The so-designed matrix is:

$$Q = \begin{pmatrix} 3/4 & 1/4 & 0 \\ 1/8 & 3/4 & 1/8 \\ 1/4 & 1/4 & 1/2 \end{pmatrix}$$

For completeness, one needs to define the transition matrices $(P_i)_{i=1,2,3}$ between tasks under each purpose. The main observation is that not all tasks are feasible for a certain purpose. For instance, under the *rest purpose*, only tasks 1 ("*rest*"), 5 ("*swim in circles*") and 6 ("*find border*") make sense thus are available. The respective transition probabilities between these tasks are given by:

$$P_1 = \begin{pmatrix} 29/30 & 1/40 & 1/120 \\ 0 & 3/4 & 1/4 \\ 3/32 & 1/32 & 7/8 \end{pmatrix}$$

For the "*exhibition*" purpose, only tasks 2, 3, 4 and 6 will be available; if we assign that order to the tasks, the reduced transition matrix P_2 (between the latter tasks) is given by:

$$P_2 = \begin{pmatrix} 7/8 & 1/80 & 1/10 & 1/10 \\ 0 & 15/16 & 3/64 & 1/64 \\ 0 & 4/100 & 19/20 & 1/100 \\ 1/24 & 1/24 & 1/24 & 7/8 \end{pmatrix}$$

Finally, the "*preying*" purpose is the most sophisticated one, and combines the feasible tasks 2, 3, 4, 6 and 7; we employed the following design for P_3 (again, the transitions exclude non-mentioned tasks and order the feasible tasks by numerical order):

$$P_3 = \begin{pmatrix} 9/10 & 1/100 & 6/100 & 1/100 & 2/100 \\ 0 & 9/10 & 6/90 & 2/90 & 1/90 \\ 0 & 5/48 & 7/8 & 0 & 1/48 \\ 0 & 1/21 & 1/21 & 2/3 & 5/21 \\ 3/20 & 1/20 & 1/20 & 0 & 3/4 \end{pmatrix}$$

It is worth to remark that if *Fisho* chooses to change to a new purpose wherein the task in execution is unfeasible, a new task will be chosen between the feasible set of tasks in that purpose with equiprobability. The reader can check that all matrices P_i are ergodic. As a consequence *Fisho*'s behavior is locally ergodic (precisely, ergodic in each purpose). A full diagram of the main program implementation is depicted in Fig. 6.

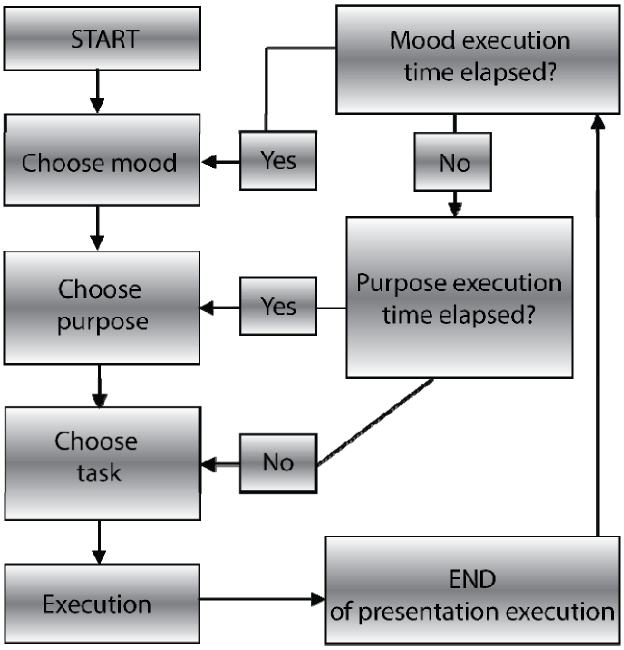

Fig. 6. Flow chart for different routines in *Fisho*'s intelligence.

V. RESULTS

Figure 7 shows the contrast between the body-curve (when swimming) of a "real" carangiform fish and *Fisho*'s body-curve. *Fisho* has precisely a fish-like oscillation shape, as desired. Additionally, an excellent hydrodynamic performance was achieved. In terms of swimming speeds, for an oscillation maximum amplitude of 30 degrees and an oscillation frequency of 0.5 Hz -when swimming straight-, *Fisho* is able to reach a terminal speed of 0.15 m/s. When the obtained terminal speed is replaced in Eqn. (3), we find a Strouhal number of 0.33, which is consistent with the range $0.25 < S_t < 0.35$ suggested in the literature [7]. The maximum speed achieved during tests was 0.3 m/s at a frequency of 1.2 Hz. All the presentations (*task-purpose-mood*) were observed in accordance with the performance predicted from simulations. Figure 8 shows *Fisho*'s trajectory when performing the "*follow border*" task and a plot of a Matlab simulation for this trajectory. The observed time evolution of the tasks was consistent with our predictions.

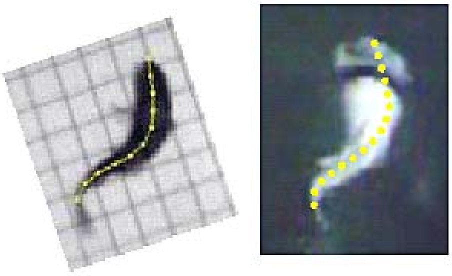

Fig. 7. Constrast between the body-curve of a carangiform fish when swimming and *Fisho*'s body curve.

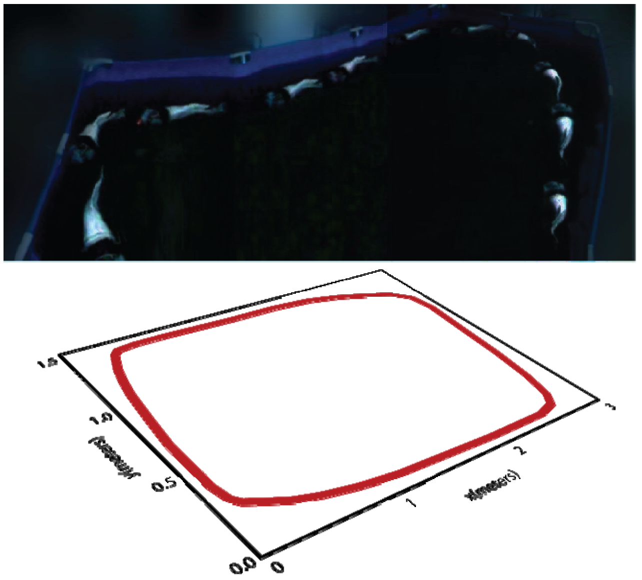

Fig. 8. *Fisho*'s trajectory when following the pool border. Picture of the experimental trajectory (up) and Matlab simulations (bottom)

VI. CONCLUSIONS

We presented *Fisho*: an intelligent and fully autonomous robotic fish. The main elements that characterize its fish-like behavior are its stochastic intelligence (governed with a three-layer Markov chain), its two-joint-based motion mechanism which implements a carangiform swimming style, and its fish-like external morphology. Since its skin is made from a unique very flexible piece, its tail movement results in nearly perfect oscillations which closely resemble those of real fish. *Fisho* achieves a maximum terminal speed of 0.3 m/s, and its whole design/fabrication cost is only a few hundred dollars.

Our experimental observations confirm that *Fisho* is able to develop several tasks with a clear similarity to real fish. As a future work, we would be largely enriched with an exchange with experts in biology, and oceanographic researchers to find underwater applications, possibly related with non-invasive sea-life exploration and an in-depth understanding of fish life.

REFERENCES

[1] J. Liu, H. Hu, and D. Gu, "A hybrid control architecture for autonomous robotic fish." in IROS. IEEE, 2006, pp. 312–317.

[2] P. V. y Alvarado and K. Youcef-Toumi, "Performance of machines with flexible bodies designed for biomimetic locomotion in liquid environments." in ICRA. IEEE, 2005, pp. 3324–3329.

[3] R. Mason and J. Burdick, "Experiments in carangiform robotic fish locomotion," in Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on, vol. 1, 2000, pp. 428–435 vol.1.

[4] A. Crespi, D. Lachat, A. Pasquier, and A. J. Ijspeert, "Controlling swimming and crawling in a fish robot using a central pattern generator," Auton. Robots, vol. 25, no. 1-2, pp. 3–13, aug 2008. [Online]. Available: http://dx.doi.org/10.1007/s10514-007-9071-6

[5] M. Epstein and N. University, Generating Thrust with a Biologically-inspired Robotic Ribbon Fin. Northwestern University, 2006. [Online]. Available: http://books.google.com.uy/books?id=wDwohLevDcQC

[6] X. Deng and S. Avadhanula, “Biomimetic micro underwater vehicle with oscillating fin propulsion: System design and force measurement,” in Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on, 2005, pp. 3312–3317.

[7] M. Sfakiotakis, D. Lane, and J. Davies, “Review of fish swimming modes for aquatic locomotion,” Oceanic Engineering, IEEE Journal of, vol. 24, no. 2, pp. 237–252, 1999.

[8] J. Colgate and K. Lynch, “Mechanics and control of swimming: a review,” Oceanic Engineering, IEEE Journal of, vol. 29, no. 3, pp. 660–673, 2004.

[9] J. Liu and H. Hu, “A 3D simulator for autonomous robotic fish,” International Journal of automation and computing, vol. 1, no. 1, pp. 42–50, 2004.

[10] K. Hirata, T. Takimoto, and K. Tamura, “Study on turning performance of a fish robot,” in First International Symposium on Aqua Bio-Mechanisms, 2000, pp. 287–292.

[11] J. Yu, S. Wang, and M. Tan, “A simplified propulsive model of bio-mimetic robot fish and its realization,” Robotica, vol. 23, no. 1, pp. 101–107, jan 2005. [Online]. Available: http://dx.doi.org/10.1017/S0263574704000426