EC280: Mini Project

Project Title: Robotic Fish

IV Semester B.Tech 2022

Project Mentor(s):

Nikhil K. S. and Sushil Kumar Pandey

Submitted By

1. **Name: Akhilesh Dimri**

   (Reg.No. 201EC104)

2. **Name: Prerana Prakash**

   (Reg.No. 201EC147)

3. **Name : Bhuvan N M**

   (Reg.No. 201EC110)

Table of Contents

Introduction

Engineers and biologists have long been interested in how aquatic organisms are able to propel themselves through water with high efficiency.

Biomimetic systems have been receiving increasing attention from the robotics community, since natural organisms can provide important insights into the theory and design of engineer systems. For example, in the area of aquatic robots, the manoeuvrability and efficiency of live fish has motivated significant scientific interest over the past two decades in developing, modelling and controlling robotic fish.

Today we introduce you a robot fish that is able to swim and recognize tank edges and obstacles. We'll find out how we can easily create it with common insulating material and a few servo motors controlled by Arduino.

Robotic fish are underwater robots that emulate locomotion of live fish through actuated fin and/or body movements. They are of increasing interest due to their potential applications such as aquatic environmental monitoring and robot-animal interactions.

In addition to providing platforms for underwater applications such as environmental monitoring, these robots offer a means to study the behaviour of live fish.

Over the past two decades, there has also been significant interest in developing underwater robots that propel and manoeuvre themselves like real fish do.

Literature Survey

The contributions of this dissertation are primarily on the complex dynamic modelling of tail-driven robotic fish, its related averaging and control theory, and an adaptive sampling framework using a group of robots. These contributions are further elaborated below. First, the guidelines of robotic fish system design are provided and several types of robotic fish prototypes are developed for aquatic environmental monitoring, robot-animal interaction study, and education and outreach. Second, a dynamic model for a tail-actuated robotic fish has been developed. In addition, a computationally efficient approach is proposed and validated for adapting the drag coefficient of the robotic fish based only on the tail-beat bias. Third, to design and control robotic fish actuated with a flexible tail, it is essential to have a faithful and efficient dynamic model. Existing studies typically adopt a linear Euler-Bernoulli beam model is adopted to describe the beam dynamics; however, these models cannot accurately capture the beam dynamics when the beam undergoes large deformations. A model consisting of multiple, serially connected rigid segments is developed to describe the flexible beam, the effectiveness of which is validated with experimental results. Fourth, a novel control-oriented data-driven averaging approach is proposed to approximate the original dynamics of robotic fish. The merit of the proposed approach, is demonstrated in the design of a hybrid controller for targeting tracking and in the application to study of steady turning. Last, a school of robotic fish is proposed for adaptive sampling and reconstruction of aquatic fields. Distinguished from most existing work, we propose to expand the sampling area successively and reconstruct the field iteratively via analytical solutions. The effectiveness of the approach is validated via extensive simulations based on real data traces of the temperature field in a large water tank. It shows that the proposed adaptive sampling solution is more computationally efficient and requires shorter travel distances for the robots than a competing matrix completion-based method.

Problem Statement

Designing a microcontroller based robotic fish.

Objectives

To make the basic biomimetic robotic fish is made up of three parts: a streamlined head, a body, and a tail.

- The head is often made of a rigid plastic material (i.e., fiberglass) and contains all control units including a wireless communication module, batteries, and a signal processor.
- The body may be made of multiple jointed segments, which are connected by servomotors. Servomotors control the rotation angle of the joint. Some designs have pectoral fins fixed on both sides of the body to ensure stability in the water
- An oscillating caudal (tail) fin connected with joints and driven by a motor provides motive power.

Methods/Execution

In order to make a robotic fish, one needs to combine the mechanical body of the fish with the circuitry required for the servo motors and the microcontroller etc. We also need to write some “Arduino” code as we are using a microcontroller that supports Arduino. Hence, we followed a three - prong methodical approach towards this project.

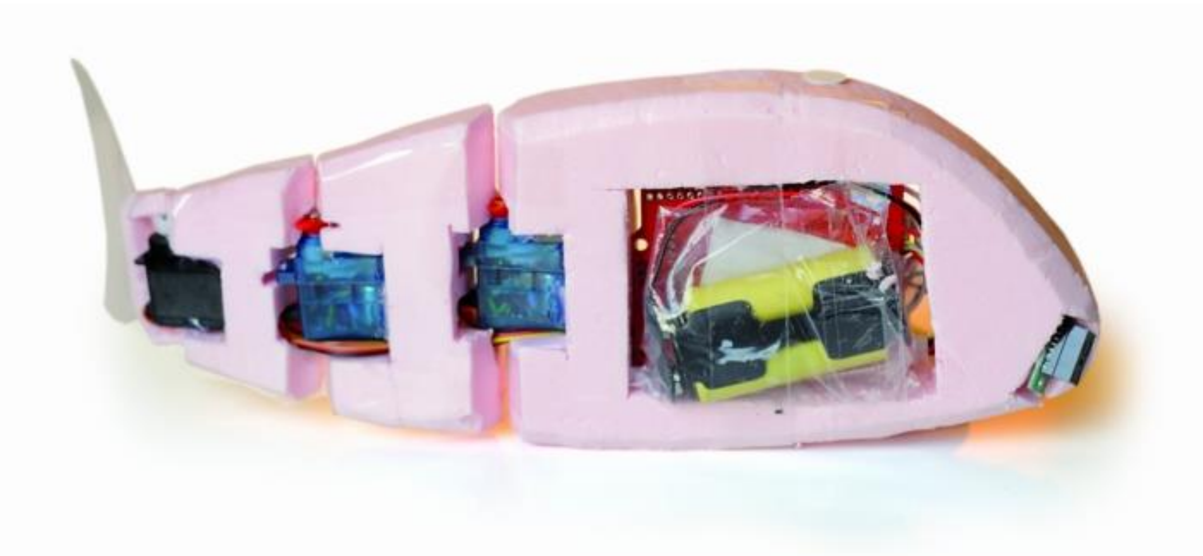
The Mechanical Part

For the fish body we used common polystyrene as the one used as insulating material for walls. Inexpensive, very robust and lightweight: it floats easily and is easily mouldable.

To make fish's swimming more realistic, we need three joints between the trunk and the caudal fin. As actuators we have chosen of common modelling servos: small, powerful enough and easily controlled by a microcontroller.

The servos are ideal because with them you can manage the movement of a small shaft connecting all segments of the fish, varying at will the position, even a few degrees.

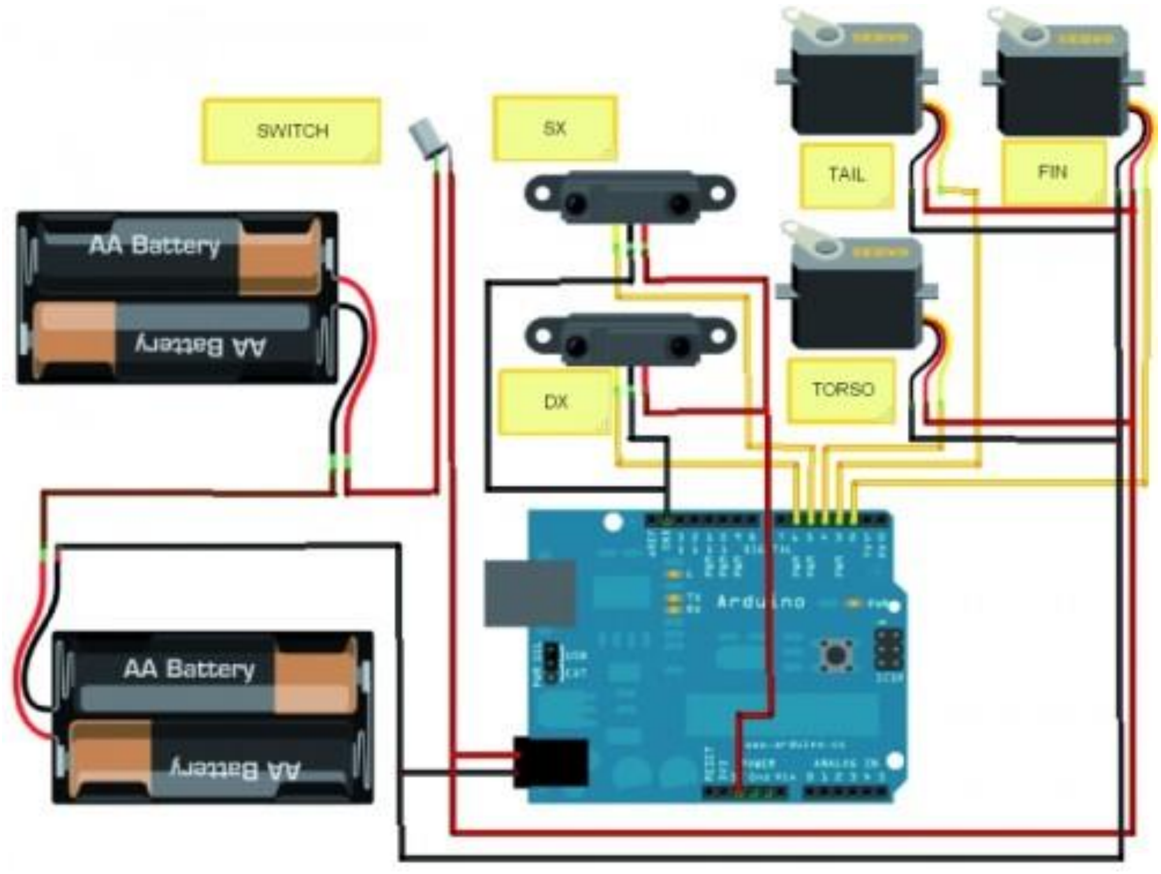
The fish body is divided into a central part and three parts, each of which moved by a servo. The caudal fin is made with plastic recycled from supermarket goods packages.
The flexibility of the fin allows you to give more realism to movement. To obtain harmonic motion, each part is attached to the next by a servo: the body of the servo should be glued on a body segment while the servo arm (connected to the shaft) shall be glued on the next.



The Electrical Connections

As control we opted for an Arduino-compatible Board called Arduino - Uno. Four simple AA batteries will provide a voltage compatible with the servo feed, while the Arduino board derives its tension thanks to its internal 5-volt regulator.

The servo motor connections are made to a bread board from where we connect them to the microcontroller using female to female jumpers. Three connections are to be made for every servo motor that has been used, one for ground, one for the voltage and the last for the switch.

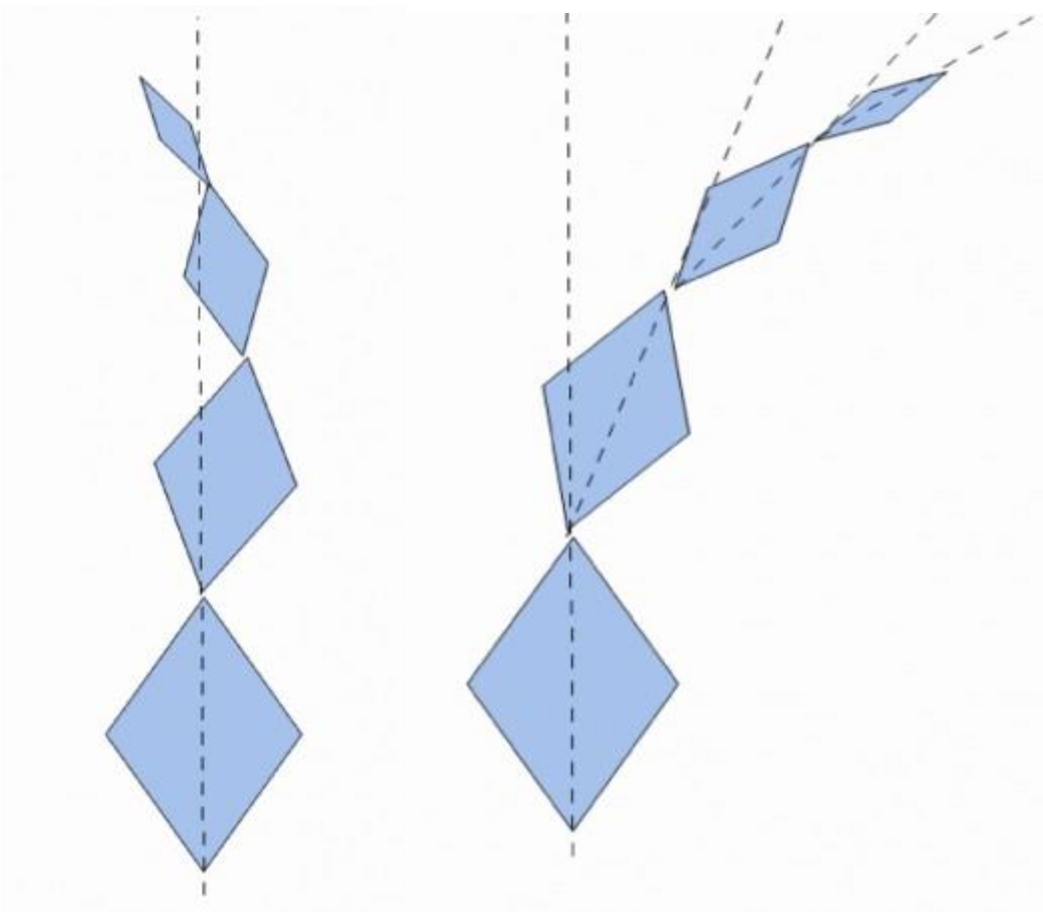


Code

```
#include <Servo.h>
Servo Servo1, Servo2, Servo3;

int i, time;
int pos1, pos2, pos3;
int pos1R, pos2R, pos3R;
int phase=45;
int velocity=2000;
int maxDeflexion=20;
int maxDefobs=20;
int actualTime;
float shift;
const int center1=98;
const int center2=90;
const int center3=105;
const int lostTime=3000;

void setup()

{

Servo1.attach(4);
Servo2.attach(3);
Servo3.attach(2);
time=velocity/360;
shift=0;

}

void loop()

{

for (i=0; i<360; i++)

{

pos1 = i+2*phase;
pos2 = i+phase;
pos3 = i;

if (pos1>359) pos1-=360;
if (pos2>359) pos2-=360;
if (pos3>359) pos3-=360;

if (pos1>179) pos1=360-pos1;
if (pos2>179) pos2=360-pos2;
if (pos3>179) pos3=360-pos3;

pos1R=map(pos1,0,180,center1-maxDeflexion-
obstacle,center1+maxDeflexion-obstacle);
pos2R=map(pos2,0,180,center2-maxDeflexion-
obstacle,center2+maxDeflexion-obstacle);
```

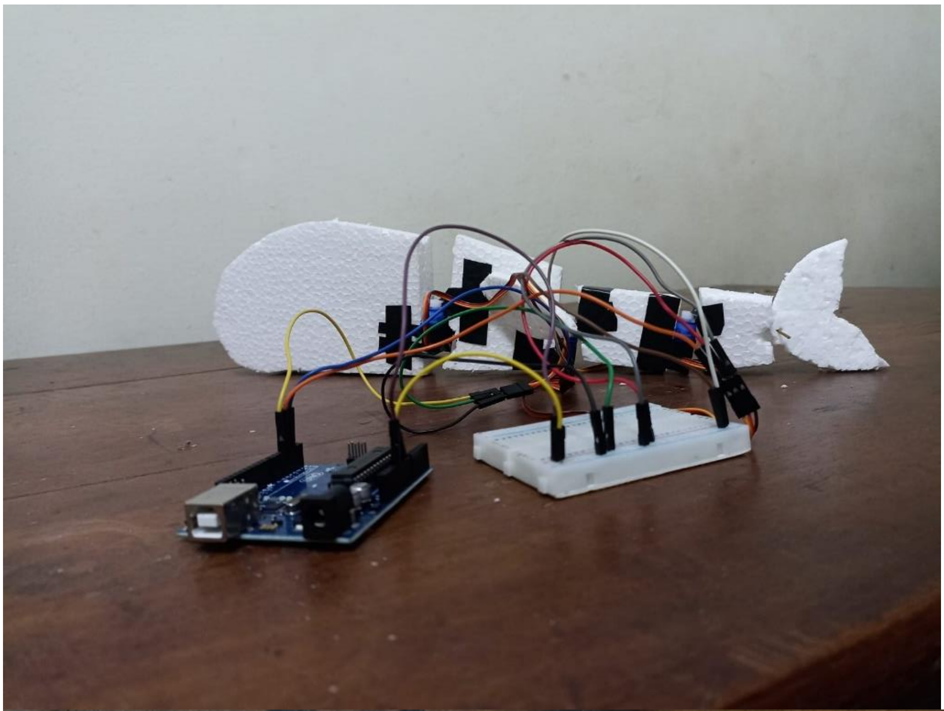
```
pos3R=map(pos3,0,180,center3-maxDeflexion-
obstacle,center3+maxDeflexion-obstacle);

Servo1.write(pos1R);
Servo2.write(pos2R);
Servo3.write(pos3R);
delay(time);

}
}
```

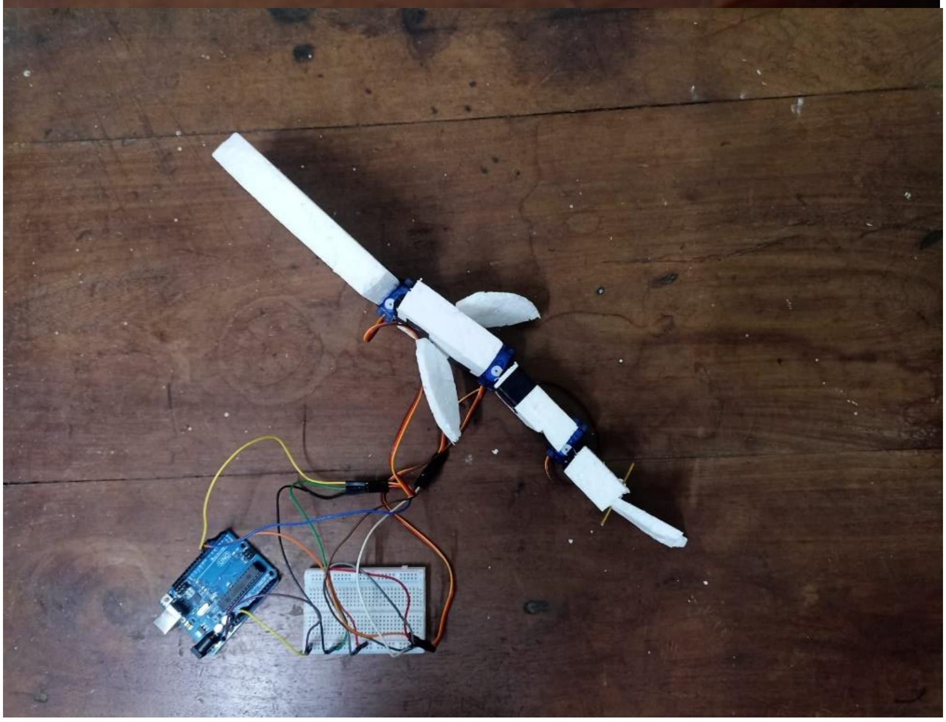
Servo movement is repetitive and follows a precise scheme. The swim takes place by moving the three servos in synchronous, according to a pattern that resembles the letter s: each segment varies slightly out of phase with respect to the previous.
For the servo movement we used servo.h library, already available on the Arduino IDE. You simply declare the servo and the multi-touch corresponding output using the servo. attach (pin). The servo.detach() method allows to release the output from the servo to use it eventually as PWM output. Commands managing servos movement are two.



Results & Discussions

By following the method listed above, the output we get is a microcontroller based robotic fish that moves in a synchronous manner with the help of three servo motors. The head, body, torso and tail of the fish are connected with servo motors which are powered by a microcontroller.





Conclusion

The aim of this project was to give hands on implementation to students on hardware especially considering how the last 3.5 semesters were online. I believe we have created a fine embedded system that not only fulfils the basic requirements of the course but we have also implemented stuff beyond just that.

The robotic fish will be able to move in water, is self-sufficient (has an internal battery) and can be used for detecting objects both near and far.

Future Scope

A system to detect obstacles such as the tank edges can be added. To create this, we used two Sharp GP2Y0D805PCB infrared sensors with digital output. These components can detect objects up to 5 cm and are quite easy to manage. Once encased in a plastic bag, can operate in water.

As it is difficult to use a conventional switch (you can’t handle it since everything is covered in plastic) we used a small magnetic contact, such as those used to detect the windows opening. It consists of a Reed contact activated by a small magnet.

References

[1] Boris Landoni, “Robofish: create your robot fish with Arduino” Retrieved from https://www.open-electronics.org/robofish-create-your-robot-fish-with-arduino/

[2] Wang, Jianxun. “Robotic fish: Development, modeling, and application to mobile sensing”, retrieved fromhttps://www.egr.msu.edu/~xbtan/Papers/Journal/Dissertations/Wang2014.pdf,p.52, Michigan State University, 2014.

[3] X. Tan, “Autonomous robotic fish as mobile sensor platforms: Challenges and potential solutions,” Marine Technology Society Journal, vol. 45, no. 4, pp. 31–40, 2011. doi: 10.4031/MTSJ.45.4.2

[4] Badamasi, Yusuf Abdullahi. "The working principle of an Arduino." 2014 11th international conference on electronics, computer and computation (ICECCO). Pp.1-4, IEEE, 2014. Doi : 10.1109/icecco.2014.6997578

[5] Jainil Shah, “ARDUINO BASED SERVO MOTOR CONTROL”, pp. 1-4, February 2016, doi:10.13140/RG.2.2.19317.06883