ETH zürich

Soft Robotics Laboratory



Swim Platform - Semester Project Final Presentation

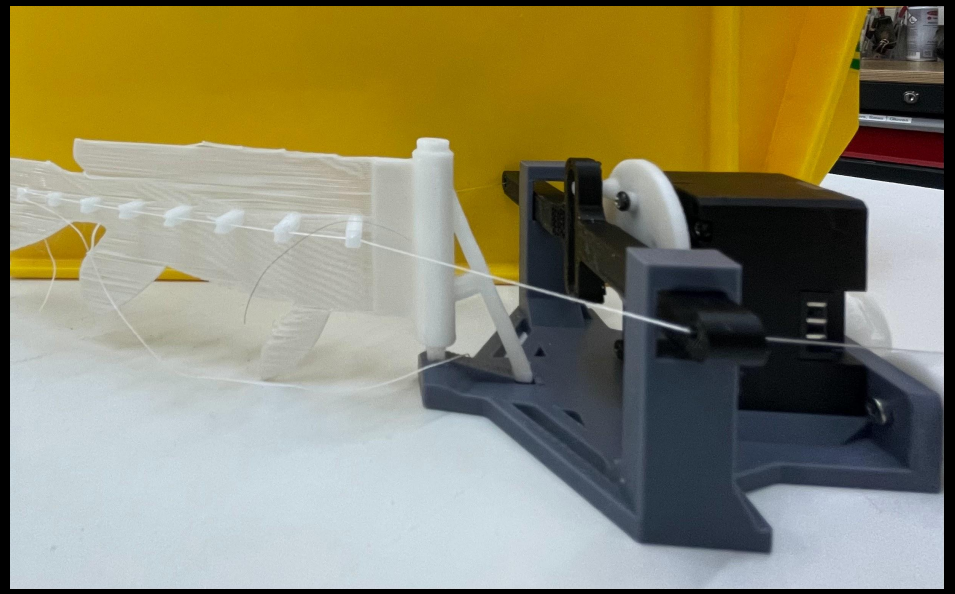
Jonathan Distler
Undergraduate Researcher
21.08.2025, Ithaca, NY

Institute of Robotics and Intelligent Systems

D MAVT

Background

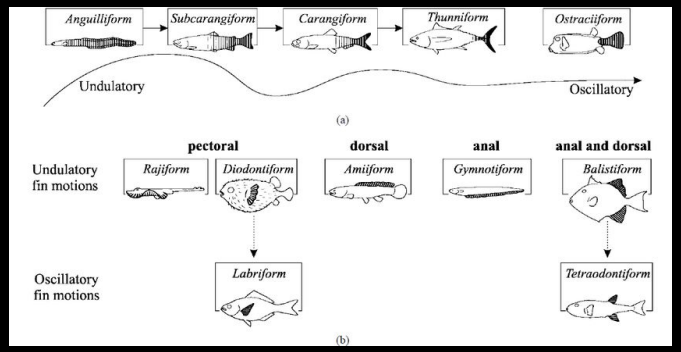
- With explosions in manufacturing and deep-learning methods, soft robotics have become more and more commonplace and relevant as a tool to solving complex issues (*Kim, 2021*)
- Underwater soft robotic systems are adaptable and don't require user input, meaning all actions can be performed autonomously; soft manipulators also allow for movement and grasping patterns without fear of damaging delicate systems (*Youssef, 2022*)
- UAVs can also be equipped with flexible tails for higher degrees of maneuverability underwater (*Fish*, 2013)



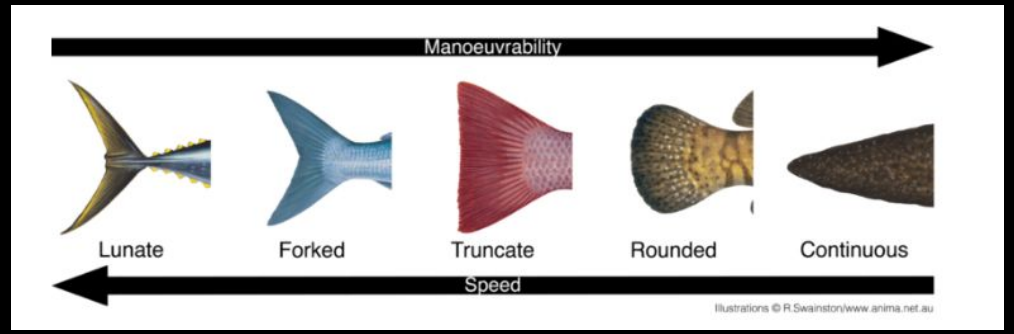
Project Goals

Throughout my time on the swim platform project, there were two broad goals:

1) The primary goal of this project was to develop an optimized tail for the SoFi swim platform to produce realistic thrust results for robotic fish with similar thrust results compared with fish exhibiting BCF locomotion (Body and/or Caudal Fin Propulsion) (*Xenyu*, 2022)
   - This goal would be worked on in parallel through Simulation and Experimentation



Arslan, 2019



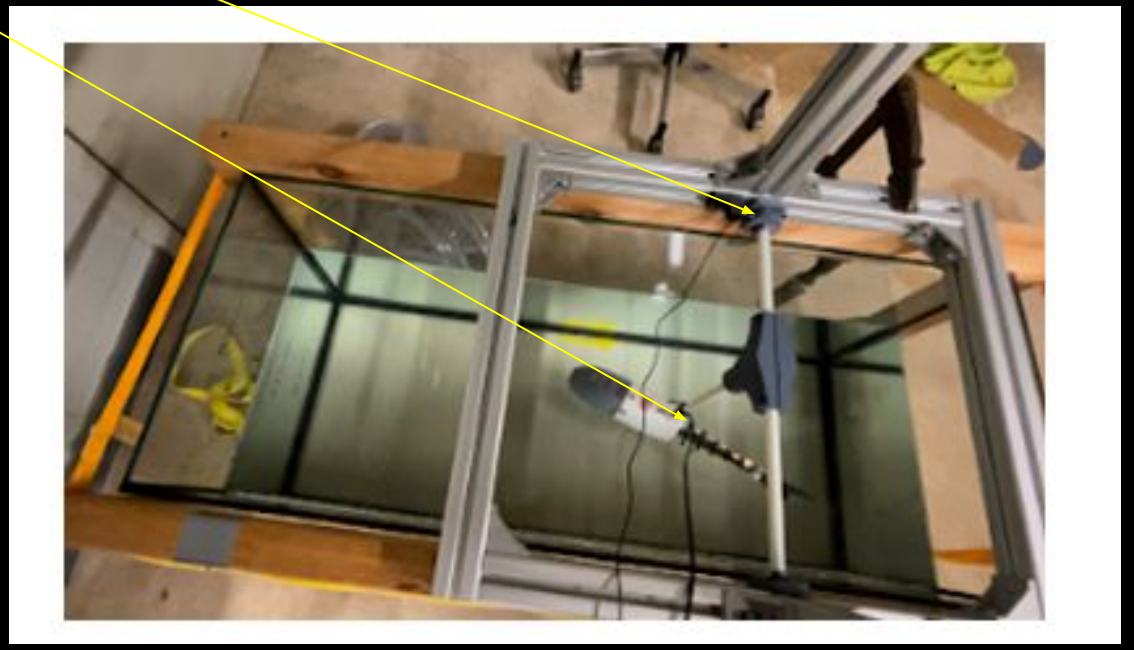
MarineWaters, 2020

2) A secondary goal was to implement autonomous control

Hardware Setup

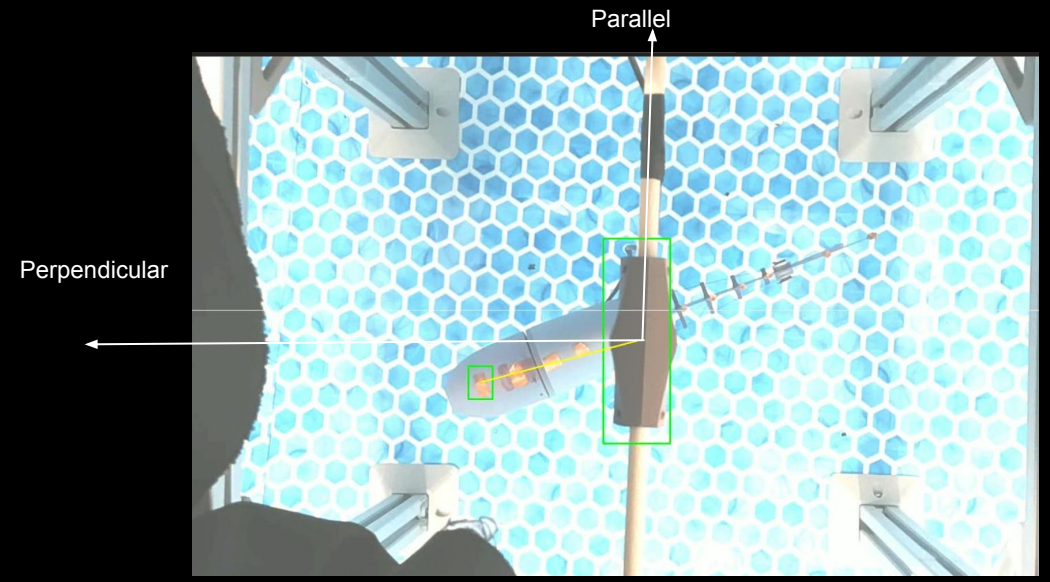
- Our hardware setup consisted of the following pieces of equipment:
  - An Oak-D camera with an appropriate camera control script
  - An HX-711 load-cell (rated for 10 Kg) with a serial streaming script
  - A water-resistant Dynamixel XW430 along with Matthew's Dynamixel Class fork
    - All scripts were combined to be run and stop simultaneously given motor and camera cues





Computer Vision and Tracking

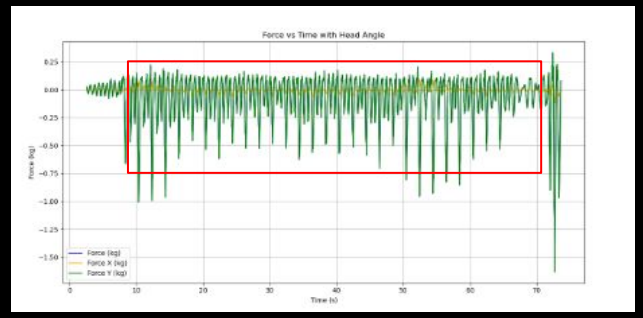
- To complement the hardware setup, I helped develop a computer-vision script that takes a video and two user-selected bounding boxes; one for the *head of the fish*, the other for the *semi-rigid force-amplification rod*
- The CV script tracks the angle of the center of the head with respect to the center of the bounding box
- The angle is overlaid with the load-cell data to determine the force components parallel and perpendicular to the semi-rigid force-amplification rod



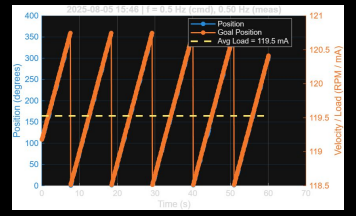
Thrust Data Analysis

Introduction | Test Results | . . .

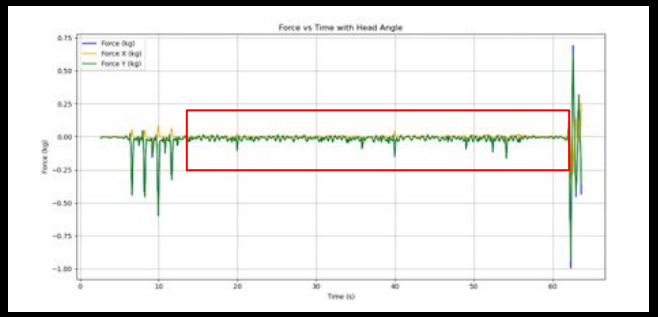
- Test data is provided in the following drop boxes:
  - Mean Force Component
  - CV Force Component
  - CV Outputs
  - Force Graphs
- Analysis from 0.1 hZ to 0.5 hZ shows a trend of unequal amplitudes with a clear negative thrust bias



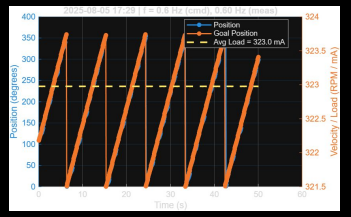
0.5 hZ test



- Analysis from 0.6 hZ onwards shows almost no amplitude with the only amplitude being from motor positional resets



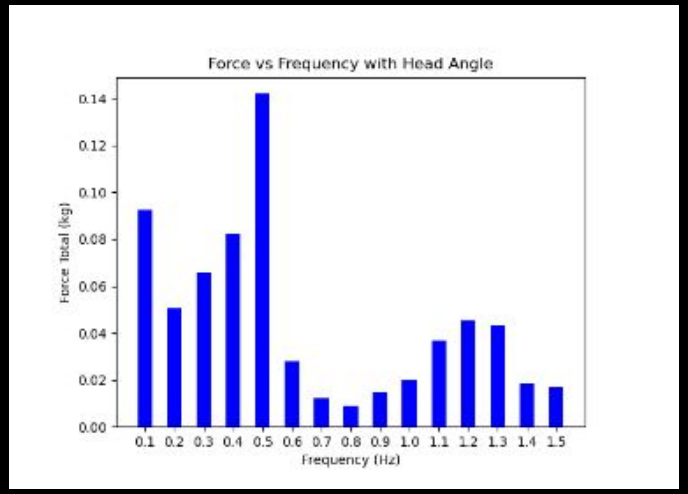
0.6 hZ test

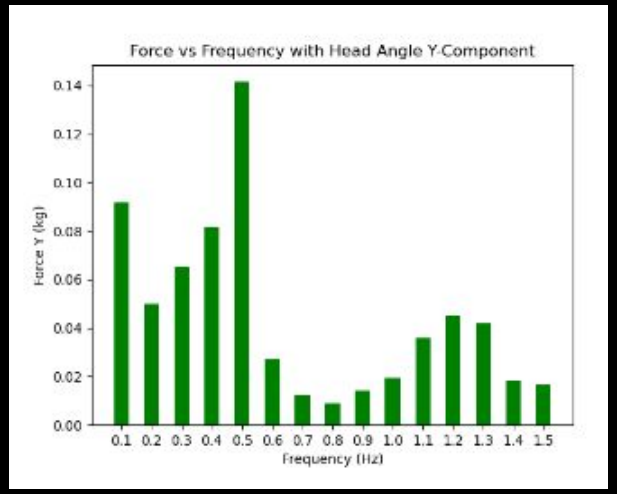


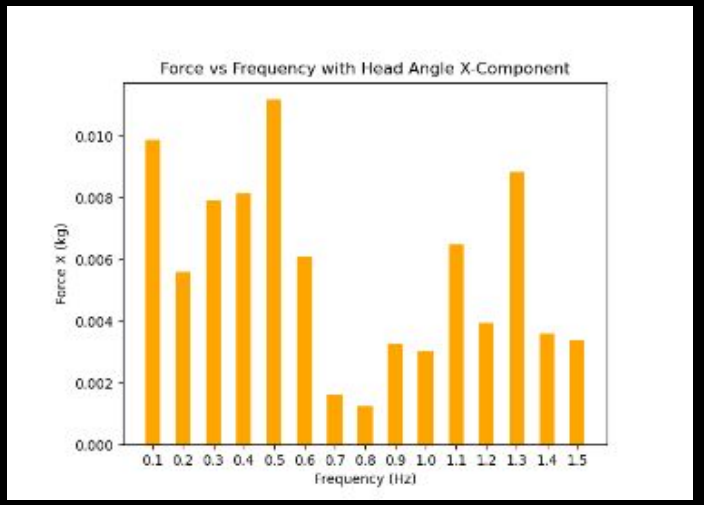
Thrust Component Data Analysis

- Ideally, one would want to maximize the ratio between the perpendicular component (X-component) and the parallel component (Y-component) to the semi-rigid force-amplification setup, maximizing thrust in the direction normal to the fish's starting orientation

- The following are the thrust component results across all frequencies, however, given poor experimental data (thrust was only produced on one half of the strokes), they don't provide conclusive results

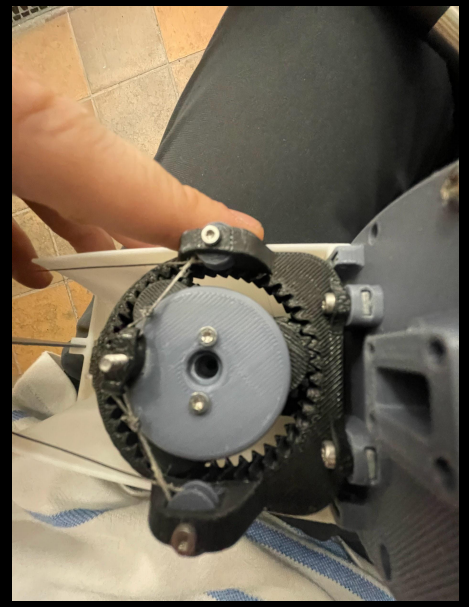






Hardware Next Steps

- More careful considerations need to be taken to prevent hardware mishaps and provide better/more useful experimental data
    - Better experimental data can better inform simulations
- Finally, given our robot setup, we could iterate over the different tail prototypes referenced in *MarineWaters, 2020* to optimize for given characteristics (most likely thrust)

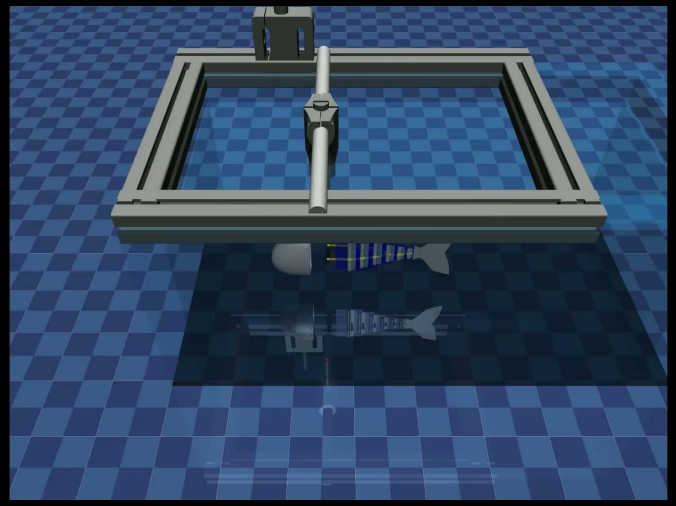




Simulation

- For simulation, it was thought that the MuJoCo (Multi-Joint dynamics with Contact) simulation would provide a versatile platform to quickly develop a tendon driven fish

- I developed a tendon driven fish using the scaled version of STL files from our real-world setup



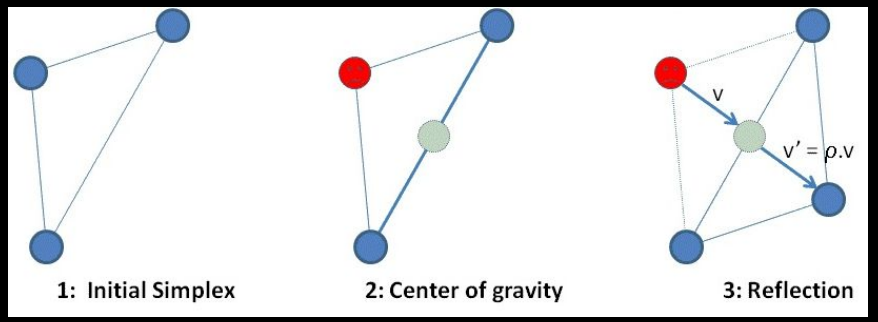
- Once the *Time vs. Position* and *Time vs. Force* graphs were analogous to the real-world setup, then one could quickly iterate over different tail shapes

Simulation Optimization

- To minimize the differences between the simulation and the real-world results, it was recommended to me to use a Nelder-Mead optimization strategy

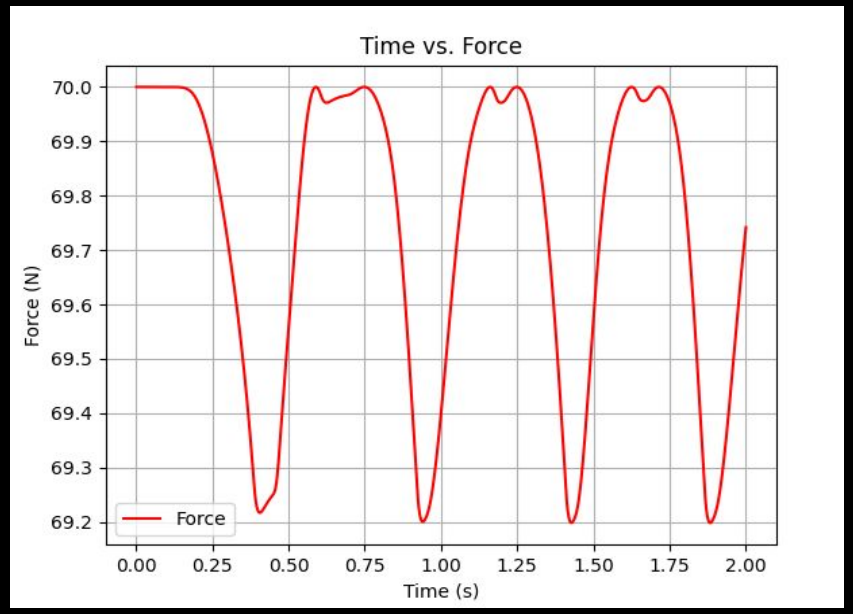
- I developed multiple iterations to minimize the difference between key aspects of the simulation and real-world results:
  - Maximum thrust force
  - Number of oscillations in a given timeframe
  - Difference in thrust force in forward and backwards thrust

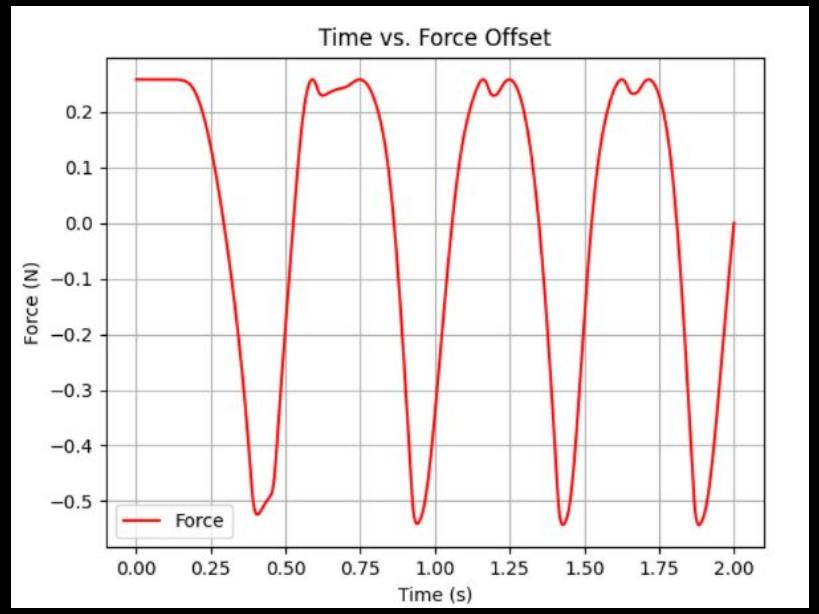
- The optimization technique worked for arbitrary goalposts, however, no methodology has been implemented for the dynamic outputs of the real-world results

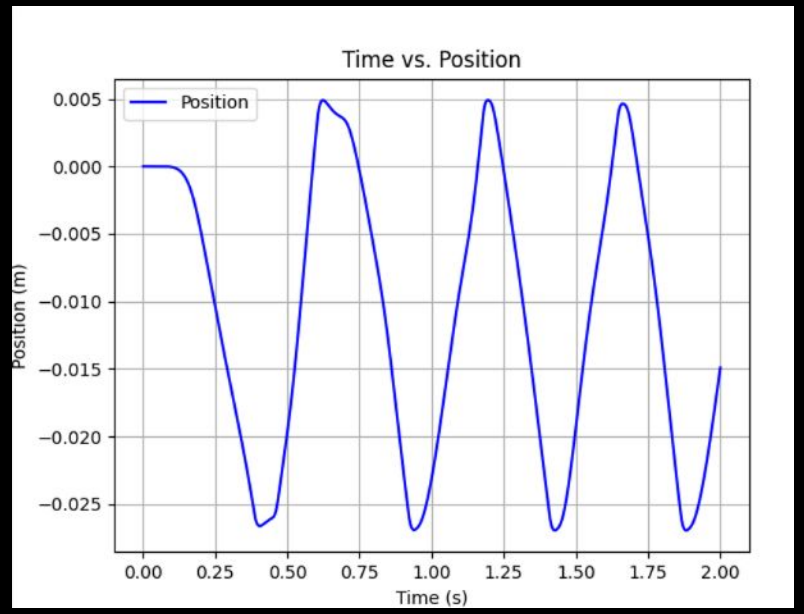


Simulation Output

- At the time of presentation these are the outputs of the simulation for a 2s time-frame:





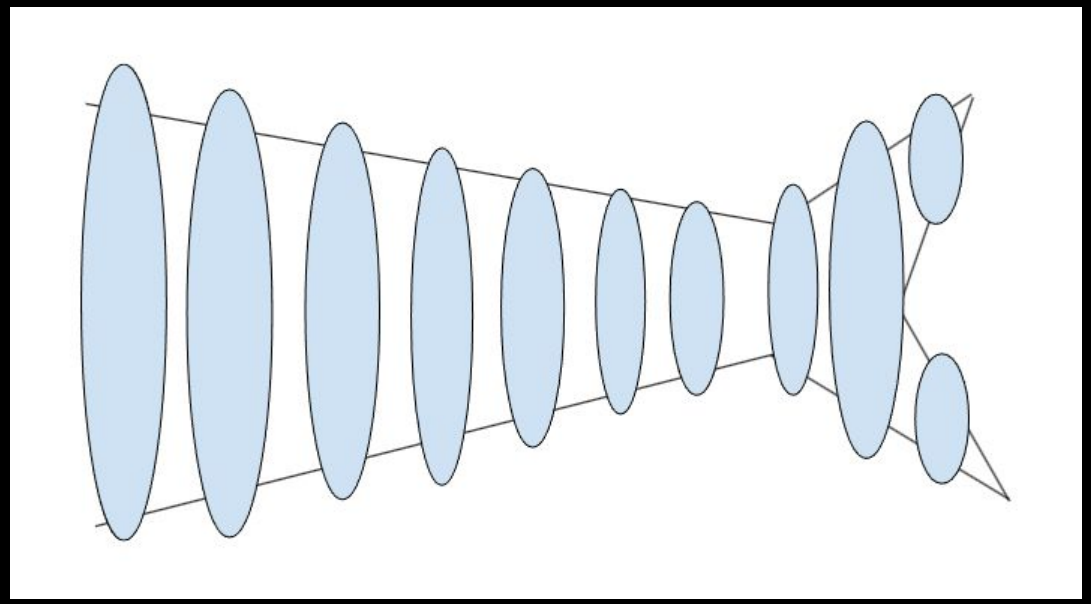


- The graphs show similar phenomena to the real-world setup. However, given many of the hardware failures (as was discussed on slide 6), it was hard to optimize for real-world success with a high-level of confidence in reproducibility in further iterations

Simulation Next Steps

- Firstly, a new methodology needs to be implemented to incorporate dynamic outputs of the more accurate real-world test results

- For the tail shape, the MuJoCo simulation deals with fluid-forces as ellipsoidal cross-sections
  - Mike and I devised a plan to “Riemann-Sum” the cross-sections of the tail, breaking the tail versions into smaller segments (still using the ellipsoidal fluid-force cross-section), and approximate the real-form factor of the tail



Dynamical Setup for First Linkage

- In conjunction with the MuJoCo simulation, I also tried to develop a dynamical model to possibly overcome the usage of the ellipsoidal approximation technique
- The setup is as follows:

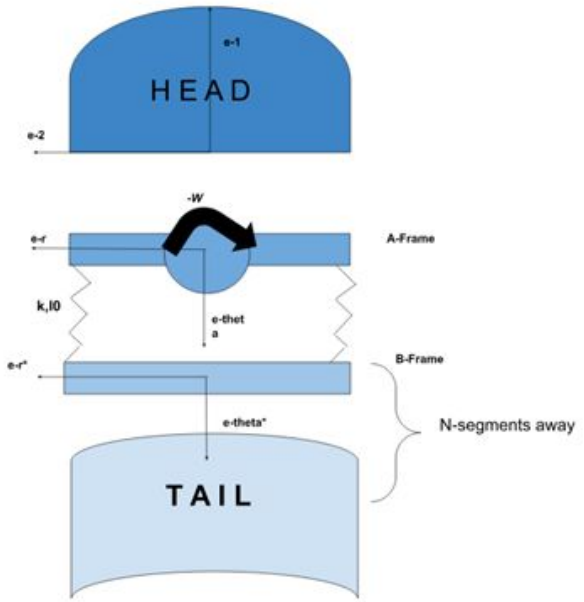


Figure 14: Top-View of the Fish with Labeled Frames for the Dynamical Derivation

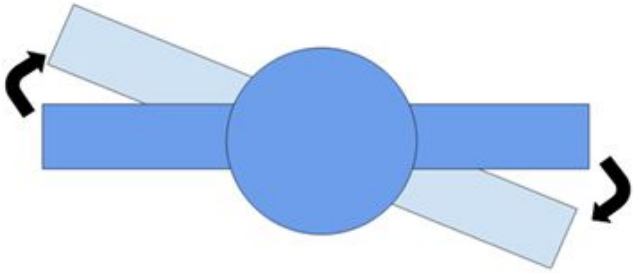


Figure 15: Top-View of the Motor-Movement, showing general approach to how the motor moves in the derivation

Dynamical Approach for First Linkage

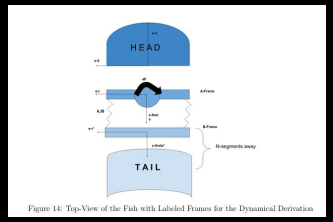
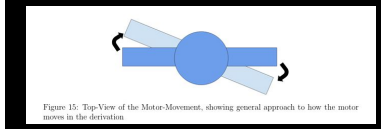


Figure 14: Top-View of the Fish with Labeled Frames for the Dynamical Derivation


Figure 15: Top-View of the Motor-Movement, showing general approach to how the motor moves in the derivation

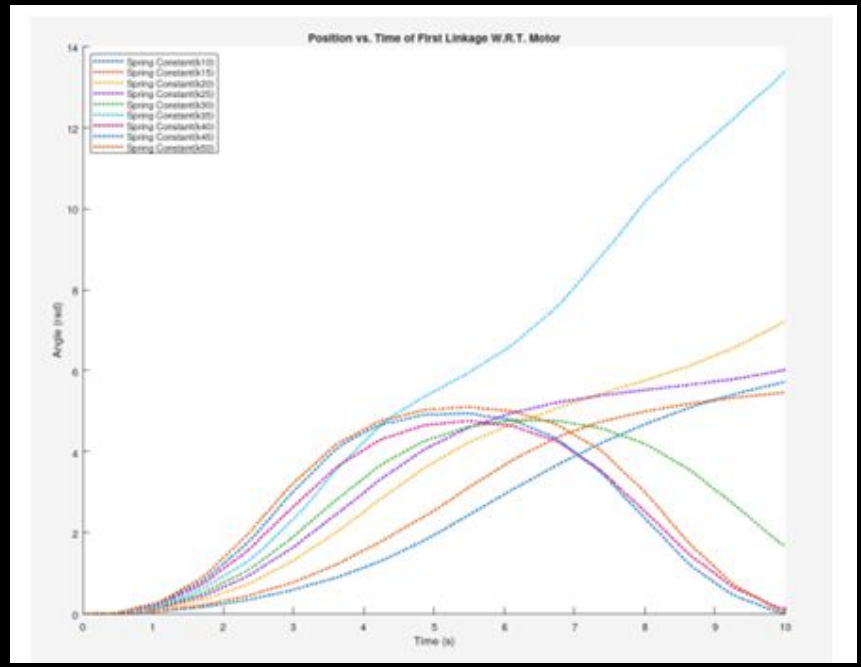
- After some hand-calculations, I arrived at the following result:

$$\sum \mathbf{M}_p = d\Big[\big(\cos\theta\cos\alpha + \sin\theta\sin\alpha\big)\hat{\mathbf{e}}_r + \big(\cos\theta\sin\alpha - \sin\theta\cos\alpha\big)\hat{\mathbf{e}}_\theta\Big]$$
$$\times \left(-k\left(|h - l\sin\theta|\,\hat{\mathbf{e}}_\theta + |l\cos\theta - l_0|\,\hat{\mathbf{e}}_r\right)\ldots\right.$$
$$\left.\cdot\frac{(h - l\sin\theta)\hat{\mathbf{e}}_\theta + (l\cos\theta - l_0)\hat{\mathbf{e}}_r}{\sqrt{(h - l\sin\theta)^2 + (l\cos\theta - l_0)^2}}\right)$$

Using $I_p = I_{\text{rod}} + m\left(\frac{d}{2}\right)^2$ from the parallel axis theorem, as well as the fact that:

$$I_p\ddot{\alpha}_p = \sum \mathbf{M}_p$$

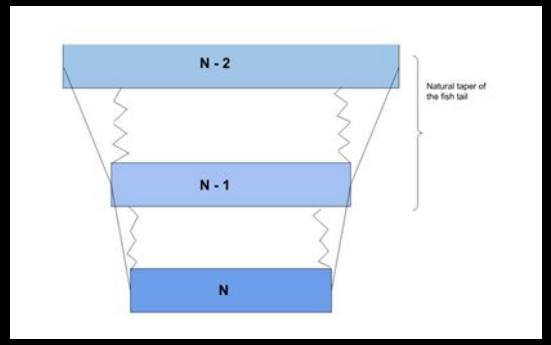
The terms can be reorganized to solve for $\sum \mathbf{M}_p$ and numerically integrated.



- The trade-off with this model compared with the aforementioned MuJoCo simulation is the usage of *Mp*, the moment acting on the linkage at *point p*. It is not a known quantity and would need to be informed through experimental data

Dynamical Approach for n-th Linkage

- Using a similar approach for subsequent linkages, except for the assumption that the n-1st linkage can't be assumed to be without acceleration, one arrives at the following equation:



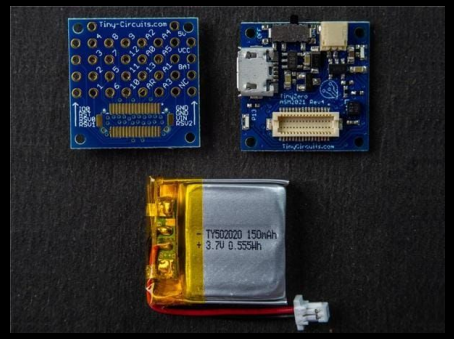
$$\ddot{\alpha}_i = d_i \left(\cos\theta_{i-1}\cos\theta_i + \sin\theta_{i-1}\sin\theta_i\right) \cdot \left(- k \left(\left(\left| h - l_{i-1}\sin\theta_{i-1} + l\cos\theta_{i-1} \right| - h_{i/i-1} \right) \cdot \frac{h_{i/i-1} - l_{i-1}\sin\theta_{i-1}}{\sqrt{\left(h - l_{i-1}\sin\theta_{i-1}\right)^2 + \left(l - l_{i-1}\cos\theta_{i-1}\right)^2}} \right) \right) + \frac{\sum_{j=1}^{i-1} \frac{M_j}{I_j}}{I_i} \hat{\mathbf{e}}_r$$

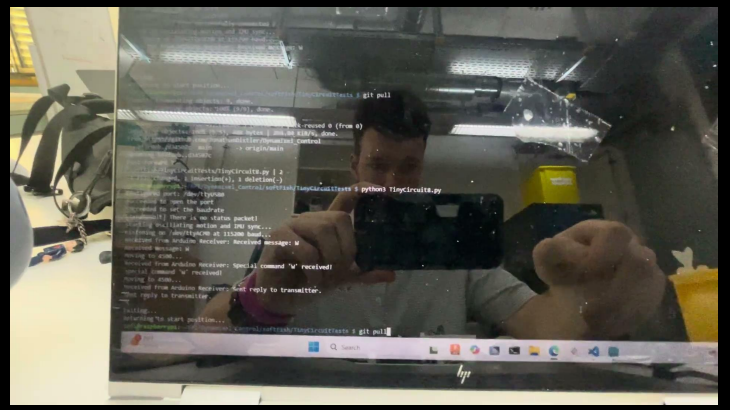
- Again, it comes with the issue of how to deal with the *Mj* (could possibly taylor-expand or take the terms to be constant)

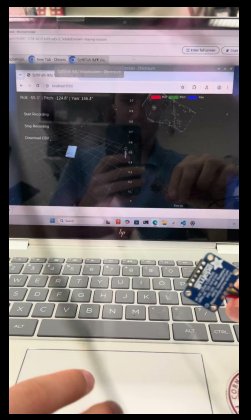
First Autonomous Option

- As a side project, we developed an autonomous method to control the Dynamixel motor remotely via an onboard Raspberry-Pi and a Tiny-Zero module

- The user inputs a command (in this case it was generalized to “W”). The key is decoded and sent via an out-of-water TinyZero transmitter module at 433 MhZ to the corresponding TinyZero receiving module embedded in the fish
  - If it is able to decode the message and it is an appropriate message (ie. not a letter or number besides “W”), the receiving module will output an appropriate response
  - When paired with the previously shown motor control, this provides a form of autonomy with different letters translating into different movements patterns
  - Could also implement with our onboard 6-DoF IMU to avoid human-element







Second Autonomous Option

- As another form of autonomy, I trained a very simple PyTorch model on the hand-drawn numbers dataset (MNIST)
- Then, I provided an example camera undistortion matrix (which would need to be tuned for different environments and different distances), such that the fish would take pictures underwater at some given rate, then it would undistort the images and make a prediction based on the pathway to the train model
  - Again, this could be paired with specific movement patterns for specific numbers along with the Dynamixel motor control

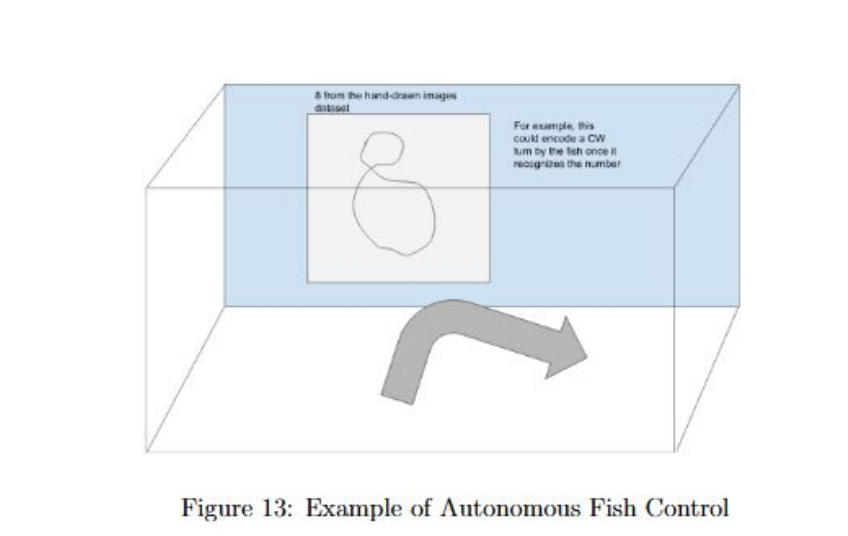
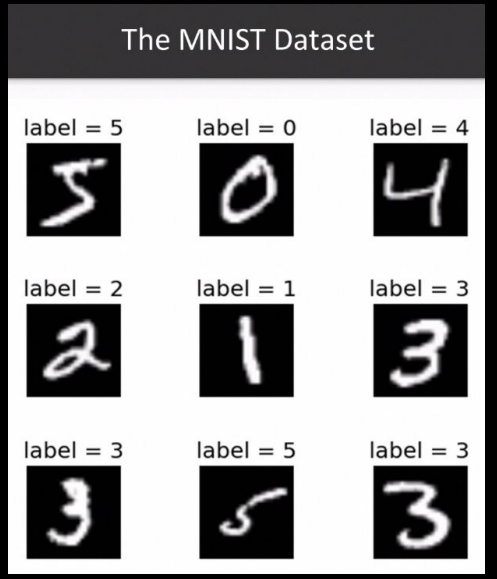


Figure 13: Example of Autonomous Fish Control



Sources

Youssef, S. M., Soliman, M., Saleh, M. A., Mousa, M. A., Elsamanty, M., & Radwan, A. G. (2022). *Underwater soft robotics: A review of bioinspiration in design, actuation, modeling, and control. Micromachines, 13*(1), 110. https://doi.org/10.3390/mi13010110

Arslan, E., & Akça, K. (2019). A design methodology for cuttlefish shaped amphibious robot. *European Journal of Science and Technology* (Special Issue), 214–224. https://doi.org/10.31590/ejosat.637838

Fish, F. E. (2011). *Advantages of natural propulsive systems.* [Review]. *Journal/Conference Title not identified.* [PDF available from ResearchGate]

Jian, X., & Zou, T. (2022). A review of locomotion, control, and implementation of robot fish. *Journal of Intelligent & Robotic Systems*. https://doi.org/10.1007/s10846-022-01726-w