A Biomimetic Actuation Method for a Miniature, Low-Cost Multi-jointed Robotic Fish

Katerina Soltan, Jamie O'Brien, and Jeff Dusek
Franklin W. Olin College of Engineering
Needham, Massachusetts 02492
Email: [katerina.soltan, jamie.obrien]@students.olin.edu,
jeff.dusek@olin.edu

Florian Berlinger and Radhika Nagpal
John A. Paulson School of Engineering and Applied Science
Harvard University
Cambridge, Massachusetts 02138
Email: [fberlinger@seas, rad@eecs].harvard.edu

***Abstract*—This paper presents the Miniature Oscillating Robot Agent (MORA). MORA is a small (12 cm) and low-cost (∼$100) robotic fish which was designed to demonstrate a biomimetic actuation method for efficient swimming. Our goal is to enable the development of underwater robot swarms that can access tight, fragile environments and gather data from the perspective and scale of real fish. Conventional actuation methods are often too large, expensive, or mechanically complex to use in collective behavior applications, which to be practical must be easy to manufacture, low-cost, and small. We arranged six magnet-in-coil (MIC) actuators, at $1/unit, in a multi-jointed configuration of three independently controlled joints. Oscillating the joints in a sinusoidal waveform allowed us to replicate the efficient undulatory body motion seen in fish. to replicate the efficient undulatory motion seen in fish. In initial straight-line swimming experiments, MORA achieved a speed of 0.37 BL/s with the potential for faster and more coordinated movement with further experimentation of MIC control settings.**

I. Introduction

Exploring the ocean and collecting data to make informed decisions about its protection is an integral part of being good stewards of one of our most valuable resources. An important aspect of this responsibility is observing the state of the environment in the context of the marine species that it most immediately affects. Underwater robots with biomimetic actuation have the unique potential to integrate into an ecosystem with minimal disruption and interact with it in a natural manner [1]. Imitating the locomotion of real fish not only allows a robot to blend in with its surroundings but also increases its efficiency. Fish use the energy of the vortices they create to aid in their swimming by coordinating the motion of their bodies [2]. Decreasing the power consumption of a robot and increasing its speed enable longer missions and the use of higher-power sensors for more detailed data collection. A robot whose size and maneuverability approaches that of a small fish would be very useful for exploring fragile, hard-to-access areas of particular interest to scientists. It could perform tasks such as pipe inspection in tight-fitting spaces, or species monitoring from the inhabitants' scale and perspective in coral reefs. These robots could also be assembled into a distributed, dynamic sensing network, creating opportunities for collecting data over large areas, easily scaling search and sense missions, and providing a synthetic biology test bed for replicating swarming behaviors in biological systems.

While these robot schools could be a very effective method of collecting data, to be practical, each robot agent must be low-cost and relatively easy to manufacture. The cost and size constraints make designing a robot for collective behavior applications challenging because a miniature, low-cost, and waterproof actuator is required. Most conventional actuation methods, such as rotary shaft motors, hydraulics, and pneumatics, are too large, expensive, or mechanically complex. This need inspired the development of the magnet-in-coil (MIC) actuator by Berlinger et al., at $1/unit, allowing for multiple actuators to be used on a single robot without appreciably increasing its cost [3]. The MICs can be arranged for greater maneuverability at small scales or utilized to explore a bio-inspired swimming gait that could lead to greater efficiency and speed.

In this paper, we present a multi-jointed actuation method using the MIC actuator to enable the development of miniature, biomimetic robots. Each joint is individually controlled, allowing for the exploration of various swimming gaits and motions. Additionally, to test this configuration of actuators and support further development for swarming applications, we designed the Miniature Oscillating Robotic Agent (MORA) robot.

II. Previous Work

The hydrodynamic advantages of biomimetic actuation are explored in a large body of work, supported by perspectives and analysis from biological, physical, and engineering disciplines [4], [5], [6]. In the robotics community, various actuation methods have been developed for underwater robots that mimic the undulating swimming gait of fish, with the aim of reaching their speed and maneuverability. The iSplash-I robot used a crankshaft to create a cohesive, undulating motion with an open-framed, multi-jointed body, reaching a straight-line speed of 3.4 body lengths per second (BL/s) [2]. Its successor, the iSplash-II, achieved the remarkable speed of 11.6 BL/s with a simplified power transmission system driven by a continuously rotating motor [7]. Liu and Hu's G9 series fish had independent servo motors on multiple joints whose turning angle and synchronization could be variably controlled to replicate different swimming gaits and turns; its top speed was 1.02 BL/s. In the realm of soft robotics,

978-1-5386-4814-8/18/$31.00 ©2018 IEEE

propulsion methods range from a hydraulic pump that cycles seawater, moving a flexible body [1], to an under-actuated tail, whose carefully designed material properties mimic the biological structure of a fish tail and propagate a variable amplitude sinusoidal wave when powered by one motor [8]. These actuation methods also vary in what swimming gaits they are optimized for. The iSplash and underactuated robots were designed for a particular swimming gait, while other robots can perform multiple, i.e., both turning and straight line swimming.

Although conventional actuation methods are successful on robots of larger sizes, the smallest being the iSplash-I at 25 cm, our target size was about 10 cm, requiring a smaller actuator. Berlinger's MIC enabled the development of a 10 cm long, autonomous, and 3D maneuverable robot at a cost of $100/unit, designed for the exploration of collective behaviors [3]. We developed a multi-jointed configuration of these MICs that could lead to an improvement in a small robot's underwater performance by replicating a biomimetic swimming gait.

While the size and cost of the magnet-in-coil actuator are favourable for our application, one of its limitations is the lack of positional feedback. Berlinger's robot employed a simple bang-bang control scheme for the actuation of single-joint fins. The more efficient yet complex undulatory body motion across multiple joints requires the characterization of the actuator and development of an open loop control method of the joint's position. By achieving individual control of each joint, we not only are able to replicate an undulating swimming gait, but also provide the opportunity for experimenting with other gaits and behaviors using the same robot fish.

III. Robot Design

We designed the Miniature Oscillating Robot Agent (MORA) to demonstrate an actuation method that can be used to implement a biomimetic swimming motion on a small, low-cost robotic fish. MORA, depicted in Fig. 1, has three individually controlled joints, each powered by two coupled magnet-in-coil (MIC) actuators. An Arduino Pro-Mini powered by a 9V battery, two L298 dual H-bridges, and a variable power supply compose MORA's operational unit, which is off-board; the system is depicted in Fig. 2. MORA's longest dimension is 12 cm, measured from the nose to its tail. Because we would like to use MORA for swarming applications in the future, its design encourages the use of inexpensive components and minimizes the time and filament required to 3D print it. The final prototype costs about $50, where the plastic filament makes up 25% of the cost and the main components (six total actuators, one Arduino, two H-bridges, and six tiny bearings) make up 60%. MORA is a tethered prototype, but we hope to shrink the size of the electronics board and integrate it into the robot's body in future work.

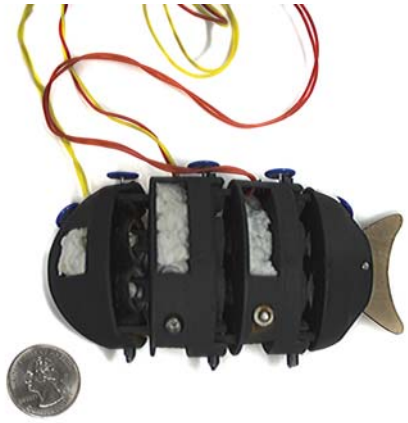

Fig. 1. MORA (miniature oscillating robotic agent) was designed to demonstrate a multi-jointed configuration of the MIC actuators. Each of the three joints is individually controlled and powered by two coupled MICs. This arrangement of the actuators allowed for the imitation of the smooth, undulating swimming motion seen in fish.

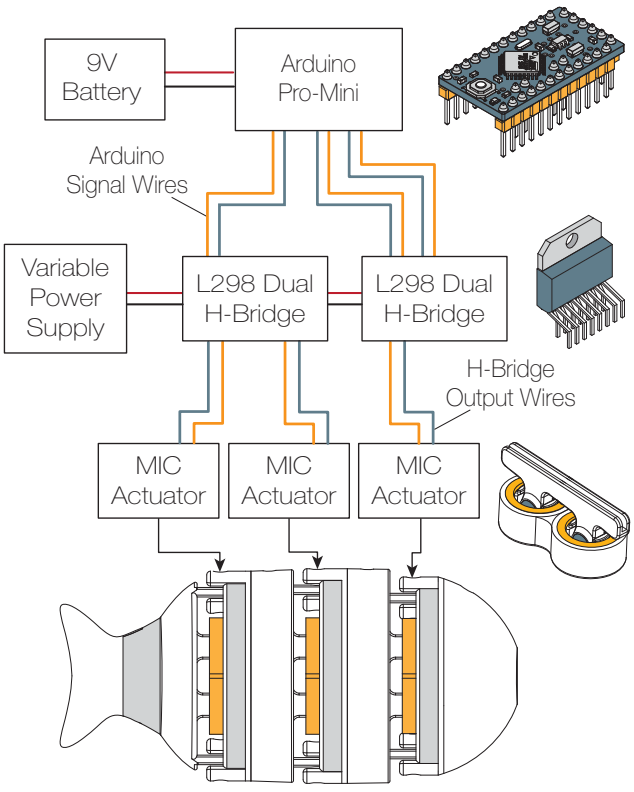

Fig. 2. An Arduino Pro-Mini is responsible for the timing and control of the actuators. Each set of two signal wires feeds into an H-Bridge which passes the current in the desired direction through a coil. As the direction of the current is alternated, the actuator in each joint flips from side to side, propelling the robot forwards. We used a separate, variable power supply to drive the coils, which allowed us to experiment with higher current throughput and torque outputs.

A. Actuator

The magnet-in-coil actuator used by Berlinger et al. is a Lorentz force actuator in which a cylindrical magnet is rotated by alternating current through a coil. As the current is reversed, the magnetic field generated by the coil flips direction, and

the magnet moves to align its own magnetic field with that of the coil's. To harness this motion, the magnet is restricted to rotating about the axis shown in Fig. 3. MORA uses double MIC actuators, which are constructed with two coils wired in parallel to increase the torque output. Each set of propulsors is controlled with an H-bridge, which is driven by signals from an Arduino Pro-Mini that encodes the fin's desired orientation (right or left), illustrated in Fig. 2.

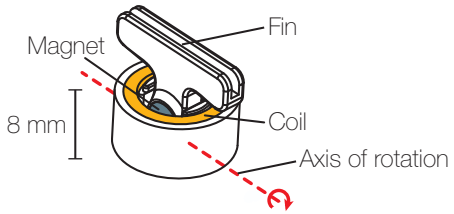

Fig. 3. The MIC actuator consists of a magnet suspended inside of a coil. As current is alternated through the coil, the magnet will try to align with the coil's changing magnetic field. The magnet's motion is physically restricted to a rotation about a single axis.

The simplest way to generate motion with the MIC is to alternate current through the coil to move the fin, or attached segment, from side to side. Because the magnet has a relatively fast response time, if the frequency is too low, the fin snaps to its maximum angular deflection, as shown in Fig. 4, and remains in this position until the current flips again. This bang-bang control was sufficient for Berlinger's use of the actuator for 3D maneuvering, but it cannot generate a smooth, undulatory motion because the MIC's output trajectory is a square or trapezoid sine wave (see Fig. 5). Another challenge in controlling the actuator is its lack of positional feedback. Unless the fin is at its maximum angular deflection position, we do not know its orientation. Our control strategy involves a combination of pulse-width modulated (PWM) signals and periods of no current to achieve a sinusoidal motion, described in the Actuator Control and Characterization section.

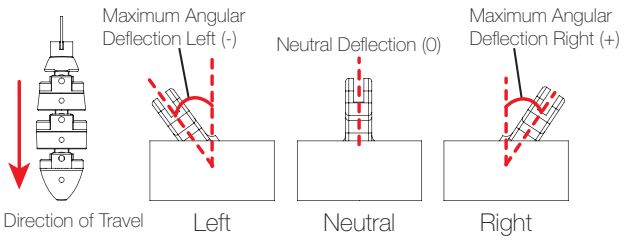

Fig. 4. To propel the robot forwards, the MIC moves from side to side in a flapping motion. It deflects to a maximum of 30 deg to each side, and until the current switches, the fin remains stationary at this position of maximum deflection.

B. Mechanical Design

The robot consists of four segments which are linked together by the three joints. There is a double MIC actuator at each joint, constructed out of two coils and two magnets

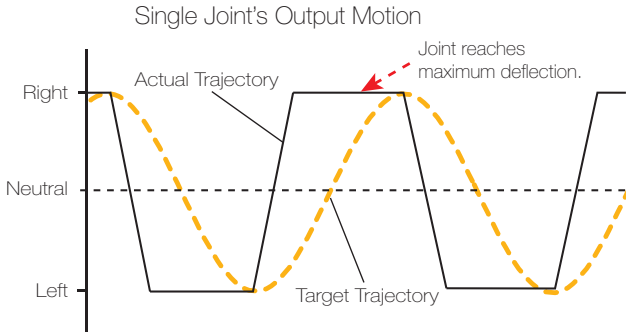

Fig. 5. The target trajectory for the MIC actuator is a smooth, sinusoidal motion. Using a bang-bang control scheme, where the MIC is getting a constant alternating current, or a PWM signal, generates a square or trapezoidal motion. The fin's low response time causes it to snap to its maximum deflection fast, and it remains stationary until the current flips again.

(see Fig. 6). The magnet is held at the center of the coil by hinges on the top and bottom of the actuator, depicted in Fig. 7. Two plastic shafts align the segments at each hinge, and a set of ball bearings helps to ensure a smooth rotation.

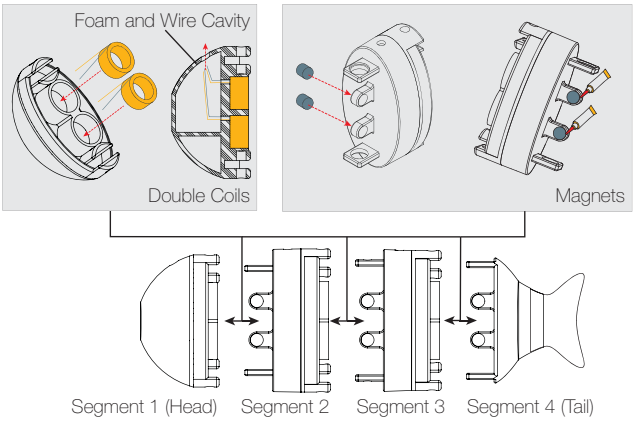

Fig. 6. The actuator at each joint of the robot is constructed using two coils and two magnets. Each component fits snugly into its corresponding retaining structure and is secured with super glue. The leads of the coils are routed into a hollow cavity at the top of the segment and out through a small hole, where they are attached to long signal wires.

Each segment has an elliptical cross section, chosen for its hydrodynamic performance, and contains two hollow cavities, shown in Fig. 8. The upper cavity is used for routing signal wires to the coils and filled with expandable or polystyrene foam for buoyancy. The bottom cavity holds a horizontally mounted bolt to which nuts are added for extra mass to achieve neutral buoyancy. The nuts can be positioned along the bolt as needed to balance the robot and prevent it from flipping onto its side due to uneven distributions of foam and weights. The tail segment does not require a foam cavity because the attached fin is made out of balsa wood, a low-density material. The size of the fin was 5.9 cm^2 in area; larger cross sectional areas provided too much added mass for the actuator

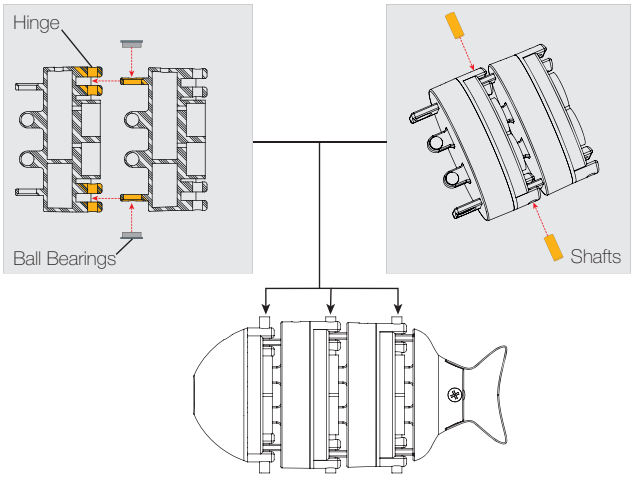

Fig. 7. The segments are connected with hinges and rotate about plastic shafts, with ball bearings ensuring a smooth motion. The position of the shafts aligns the magnet at the center of the coil, where it experiences the greatest magnetic field strength. The axis of rotation of the magnet and the hinge are also aligned.

to overcome.

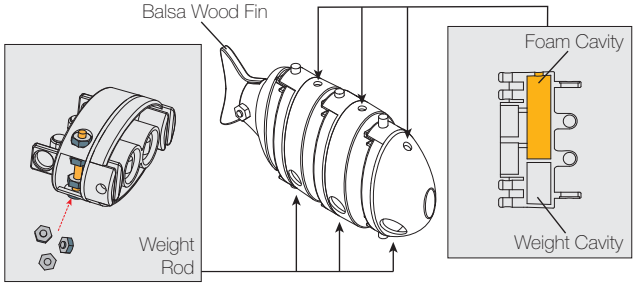

Fig. 8. To achieve neutral buoyancy, three of the segments, excluding the tail, contain hollow cavities for expandable or polystyrene foam. Another set of cavities at the bottom of the robot contains rods (bolts) onto which extra weights (nuts) can be mounted and positioned as needed for the adjustment of trim in pitch and roll.

The segments were 3D printed on a Markforged Mark Two printer with Onyx material. The closed design of the segments prevents water from passing through the robot's body and increases the generated thrust. Although we experimented with a completely closed design where we injected foam into the cavities through a small hole at the top of each segment, this approach proved to be time consuming both on the printing and part post-processing sides. Because the hollow sections and curved profiles require support material, it was almost impossible to remove it and route wires from the coil out of the segment. We settled on a half-open design where one side of the segment is opened for ease of assembly. It is consequently filled with stiff foam which effectively closes the opening.

IV. Robot Control

A. Model Approximation

Fish excel at underwater propulsion, and previous work with multi-jointed robots has shown that imitating their undulating motion increases hydrodynamic performance [2]. The ideal behavior of a fish tail is commonly characterized by the modified Lighthill model [4], a travelling sine wave of varying amplitude that represents the posture of the tail in space over time. To replicate this sinusoid with discrete, rigid segments, we used the digital approximation method proposed by Liu and Hu [9]. This approximation divides the continuous travelling sine wave function into M discrete tail postures. At each of these discrete steps, it finds the optimal position and angular deflection (q_j) that each joint should take to mimic the tail posture at that instant in time (see Fig. 9). This discretized version of the Lighthill model is expressed as $h_T(x, i)$, and is defined for tail postures $i = 0 \ldots M - 1$. We use $f(x)$ to describe the function $h_T(x, i)$ evaluated at a given posture i.

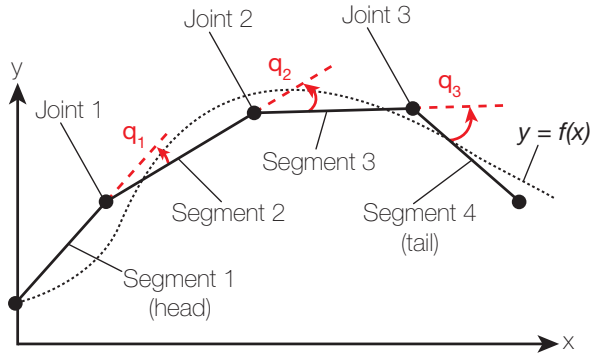

Fig. 9. The ideal motion of a fish tail can be discretized into a set of postures, $h_T(x, i)$. Each individual posture is described by $f(x)$, which is $h_T(x, i)$ evaluated at the posture i. To mimic a given posture i with a rigid, multi-jointed body, an algorithm finds the best positioning and turning angle q_j for each joint j by minimizing Liu and Hu's mean error function between the ideal posture, $f(x)$, and the segments.

To calculate the position of each segment s at a given posture i, we begin the algorithm by placing the base point of the head at the origin, assuming that the entire robot will act as the undulating tail, and defining a line, $g(x)$, to an end point that is the length of the segment. This line crosses the posture function, $f(x)$, at the cross point, as shown in Fig. 10. The objective is to find the position of the cross point which minimizes Liu and Hu's mean error between the posture function and the segment line:

$$e(x) = \left| \int_{End_x}^{Base_x} [g(x) - f(x)]dx \right|$$

This error function places more importance on preserving the overall direction of thrust between postures. The more commonly used root mean square error aims instead to achieve the exact body positioning. Once the best cross point is found, the position of the end point is calculated and set as the base point of the next joint. This process is repeated for all four

segments, the final positions of which are used to find the relative deflection angle q at each joint that can be outputted to the actuator.

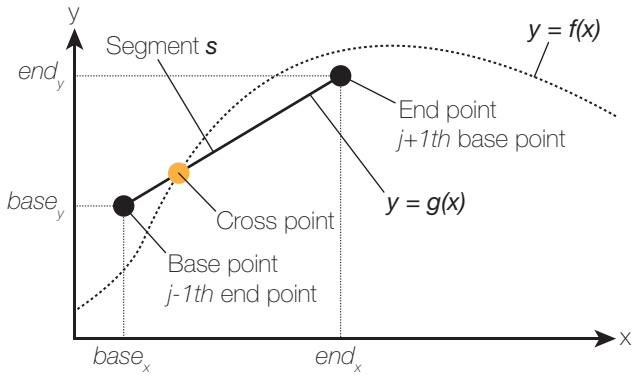

Fig. 10. To approximate a posture i, the line $g(x)$ is drawn to represent the possible orientation of segment s. The line starts at the base point, which is the end point of the previous joint, and is restricted by the length of the segment it models; the first segment's base point is placed at the origin. By moving the cross point where $g(x)$ intersects the posture function, we assess every possible orientation of the segment and choose the one that minimizes the mean error function. Once the best cross point is chosen, the end point of $g(x)$ is calculated and inputted as the base point of the next segment.

Using the Fast-Fourier transform, the motion of each joint was approximated to a first order system, showing that all three joints complete a sinusoidal trajectory at the same frequency but with varying amplitudes and phase offset from each other (Fig. 11). The algorithm also predicted that the first joint should oscillate at about half of the amplitude of the last joint but in the same direction, causing the head to counteract the motion of the tail; both the iSplash-I and G-9 series robotic fish showed similar results [2], [9].

To use MIC actuators for biomimetic propulsion, we needed to develop a way of controlling the MIC's trajectory to produce a sinusoid instead of its default square wave.

B. Actuator Control

As mentioned previously in the Actuator section, the MIC has two states that can be controlled with certainty: fully rotated to the right or to the left. There is no intermediate position control or feedback. Our challenge was to find a way to prevent the actuator from moving in its preferred square wave trajectory (see Fig. 5) and instead force a sinusoidal motion. As a first step, we used a pulse-width modulated (PWM) signal to control the voltage passed across the coil. A smaller duty cycle produced less torque which reduced the actuator's ability to push against the water in its way, increasing the time it took to transition from the leftmost position to the right. We expected to see a continuous relationship between the duty cycle and resulting amplitude. However, we found that above a 30% duty cycle, the MIC had enough torque to overcome the added mass of the water and would proceed rapidly snap to its maximum position, resulting in the square wave trajectory. Below this threshold, the actuator could not move the water and was unable to change its direction of motion.

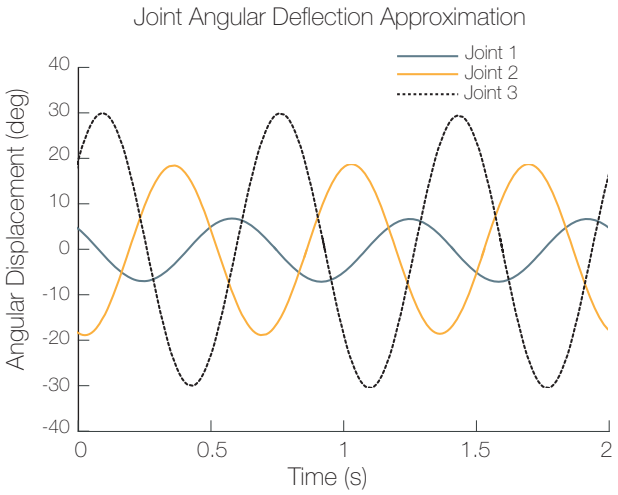

Fig. 11. The approximation of the ideal fish tail motion over M discrete postures shows that all three joints of a multi-jointed robot move in a sine wave. They oscillate at the same frequency, but with varying amplitudes: the farther the joint from the head, the larger its amplitude. The joints are also phase offset and reach their peak angular displacement at different times. Interestingly, the first joint's peak is a fraction of a second behind the last joint's, causing the head to move in opposition to the tail.

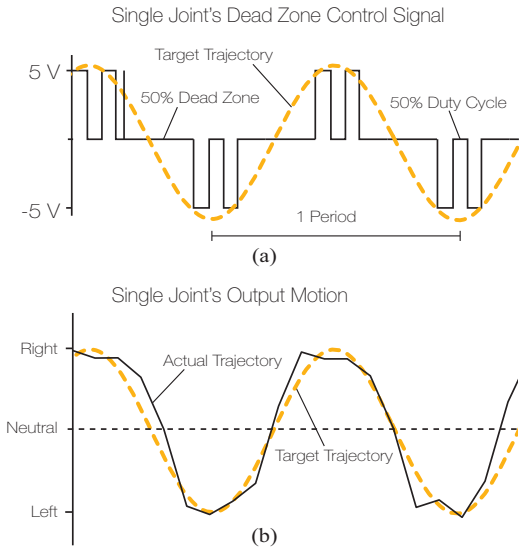

Fig. 12. (a) The dead zone parameter defines a percentage of time for which the actuator is not powered. A 50% dead zone keeps the actuator on for half of the period, generating torque only when the MIC must switch direction and pass the critical points of its target motion. (b) This control scheme allowed us to approximate a sinusoidal trajectory with the MIC.

The MIC requires the most torque when it reverses direction, which corresponds to the critical points of its sinusoidal trajectory. After this initial output of energy, there is little resistance to the joint's motion, and continuing to supply voltage to the coil causes the actuator to reach its maximum deflection very fast. By limiting the time during which the coil receives current, we were able to change the MIC's behavior from a square wave to a sine. In addition to the duty cycle, we

added the dead zone parameter which specifies the percentage of time the MIC should be powered off. A 0% dead zone represents a constant PWM signal, while a 50% dead zone turns the MIC off for half of the period, depicted in Fig. 12 (a). The resulting trajectory can be seen in Fig. 12 (b).

C. Actuator Characterization

Our earlier approximation of ideal, undulatory swimming established that the robot's joints must move not only in a sinusoidal trajectory, but at varying amplitudes. While we had determined that the duty cycle and dead zone parameters were enough to replicate a sine wave with the MIC, we needed to characterize the actuator's amplitude response. To do so, we tested a single joint's actuator over a range of frequencies with different combinations of the duty cycle and dead zone.

The frequency was swept from 1 Hz to 4 Hz in 1 Hz increments, the dead zone was set to 20%, 50%, and 80%, and the duty cycle was swept from 35% to 70% in 5% increments. The motion created by each set of parameters was recorded for 20 seconds with an overhead camera, and the technique described in the Swimming Experiments section was used to digitize the resulting trajectory and extract its approximate frequency and amplitude. The results were used to inform our choice of parameters used to control the full robot when performing straight-line swimming experiments.

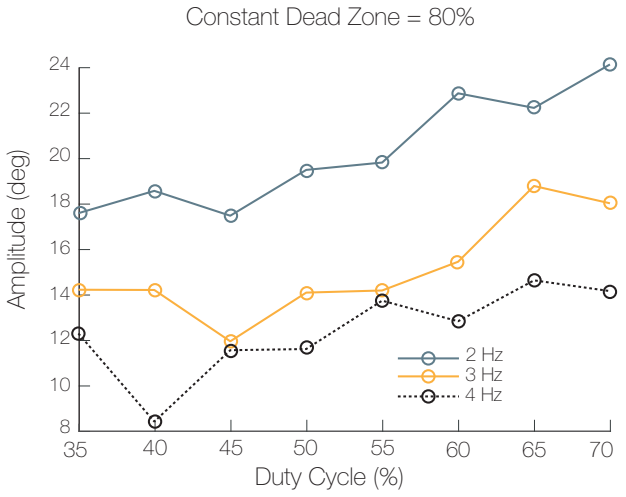

Fig. 13. Single joint experiments showed that across all frequencies increasing the duty cycle increased the output amplitude of the joint's motion. Higher frequencies attenuated the angular deflection, approaching the physical limitations of the actuator.

Although the actuator's amplitude response was not exactly linear, Fig. 13 shows that the amplitude was proportional to the duty cycle and varied inversely with the dead zone. A larger duty cycle passes more voltage across the coils and increases the torque available to overcome the added mass. The high torque also imparts a greater momentum to the actuator, and when the current turns off, based on the dead zone parameter, it can deflect further, resulting in a higher amplitude. A similar rationale explains the inverse variation with the dead zone: when the torque is sustained for longer periods of time (a decrease in dead zone), the actuator is pushed closer to its maximum deflection angle (Fig. 14).

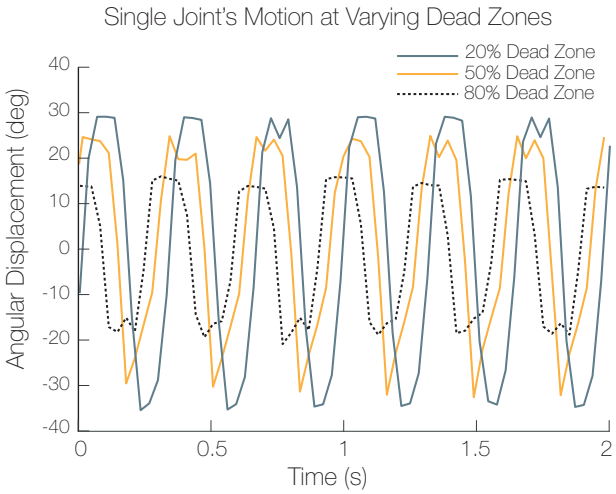

Fig. 14. Varying the dead zone setting enabled us to output a sinusoidal motion of different amplitudes with a single joint. Decreasing the dead zone increased the time during which the MIC was powered, increasing the amplitude. The duty cycle remained at 70% and the frequency at 3Hz for the experiment displayed in the graph.

The MIC's behavior was consistent across all of the tested frequencies, subject to the constraints of the actuator's dynamics. Because it cannot respond infinitely fast, the MIC's amplitude decreases at higher frequencies because it does not have enough time to finish its motion in one direction before the current switches (see Fig. 13). At very low frequencies, the default square wave motion was seen. We found that the best approximation of sinusoidal motion was produced at the 3 Hz and 4 Hz settings.

While these experiments, conducted with a single actuator, were not sufficient to create a direct mapping of duty cycle and dead zone settings to a specific amplitude, the characterization enabled us to configure our three-jointed robot MORA to propel itself forwards in straight-line swimming experiments.

V. Robot Performance

Biomimetic propulsion can be achieved with a multi-jointed robot by moving its joints in sinusoidal trajectories. Our challenge was in configuring the joints to create an overall constructive motion, where the movement of one joint augments the next instead of counteracting it. We used MORA to experiment with different amplitude and timing settings to determine whether our proposed actuation method can produce forward propulsion on a small robot.

A. Experimental Setup

Experiments were performed in a 50.8 cm x 25.4 cm x 30.5 cm clear water tank (see Fig. 15). A camera was mounted overhead to record videos of the robot. The robot was operated in a tethered configuration with long signal wires connecting

the robot to the electronics outside of the tank. To minimize the effect of the tethers on the robot, they were attached to a block of foam to make them more mobile, following the robot rather than pulling back on it. Large colored pins used for tracking and digitizing the motion were attached at the three joints, nose, and tail (see Fig. 1).

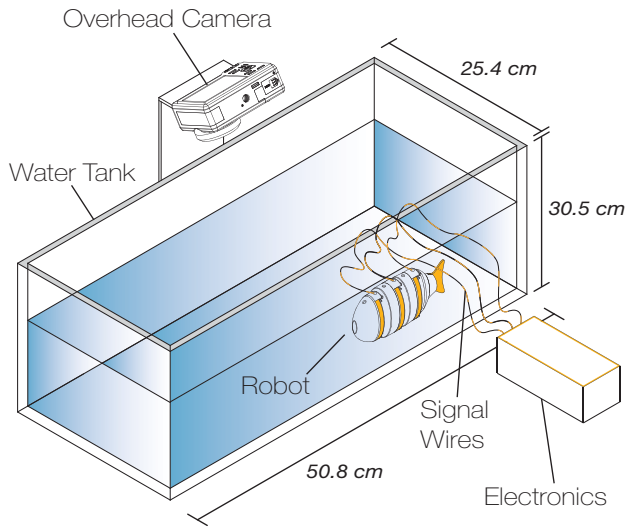

Fig. 15. Swimming experiments were conducted in a large water tank. The robot was tested in a tethered configuration, and long signal wires were used to connect the robot to the electronics located outside of the tank. An overhead camera was used to record video, which was processed in MATLAB to find the trajectories of the joints.

Four tests were performed with the settings listed in Table I. The frequency was chosen to be 3 Hz, because it produced a consistent sinusoidal motion during the single joint characterization experiments. Duty cycle and dead zone parameters were selected based on how smooth the resulting motion was and whether the robot was able to propel itself forwards. The phase offset of each joint was measured relative to the motion of Joint 1; two joints our out of phase if the offset between them is 180 degrees. We also tried two different timing sequences. The first two tests move the head in the direction opposite of the tail, while Tests 3 and 4 instruct them to move in the same direction, counterbalancing each other. The robot was placed at the right end of the tank at the beginning of each test. Because the robot was very sensitive to tether placement and motion, the tests were repeated until a relatively straight traversal of the tank was achieved.

B. Data Analysis Technique

To analyze MORA's motion, each video was processed in MATLAB to track the positions of the colored pins at the joints, nose, and tail over time. A color threshold mask was applied to each frame in the video, isolating the spots of bright blue color from the tracking pins. Using a simple blob detection algorithm, the centroid of each of these spots was determined and classified as either the nose, one of the joints, or tail pins. Because the robot's undulatory motion did not change the joints' relative order (from left to right), we were able to automate this process. Fig. 16 shows the result of this algorithms on a few frames of video from Test 2. The positions of the centroids were then used to find the angular deflection of each joint, calculated as the angle between the two adjoining segments, illustrated in Fig. 9. The angular deflection was recorded throughout the entire duration of the video, creating the trajectories in Fig. 18. The frequency and phase of this motion were then approximated using the Fast-Fourier transform, and the trajectory's peaks were averaged to determine its approximate amplitude. MORA's speed was calculated by averaging the middle joint's change in position between frames and multiplying it by the frame frequency. We chose to use the middle joint for velocity calculations because it exhibited the smallest amplitude and was the approximate location of the robot's center of mass. The average speeds for each test are listed in Table II.

TABLE I: Experiment Settings

		Duty Cycle (%)	Dead Zone (%)	Phase Offset (°)
Test 1	Joint 1	70	50	0
	Joint 2	70	50	107
	Joint 3	100	20	-145
Test 2	Joint 1	70	50	0
	Joint 2	70	20	107
	Joint 3	100	20	-145
Test 3	Joint 1	70	70	0
	Joint 2	70	20	180
	Joint 3	100	20	-109
Test 4	Joint 1	70	70	0
	Joint 2	90	20	180
	Joint 3	100	20	-109

C. Results

Tracking each individual joint revealed that we achieved an approximately sinusoidal motion with each actuator (see Fig. 18). Although our model predicted that the head joint should move at the smallest amplitude, MORA's middle joint moved the least, suggesting that there was destructive interference between the joints' motions. In Tests 1 and 2, shown in Fig. 16, the head and tail moving directly opposite of each other canceled out any significant motion in the middle. Tests 3 and 4 showed a higher amplitude, comparable to the other joints, suggesting that the second timing sequence created a more constructive interaction between them (see Fig. 17). Furthermore, the head and tail counterbalancing behavior was observed in previous works, [9], [2], and MORA's highest speed of 0.37 BL/s (see Table II) was achieved with Test 4's settings. We intend to continue experimenting with the timing and amplitude parameters to create an even smoother undulation of the robot's body.

D. Discussion

During the experiments, MORA tended to veer off of a straight-line course due to an asymmetrical flapping motion,

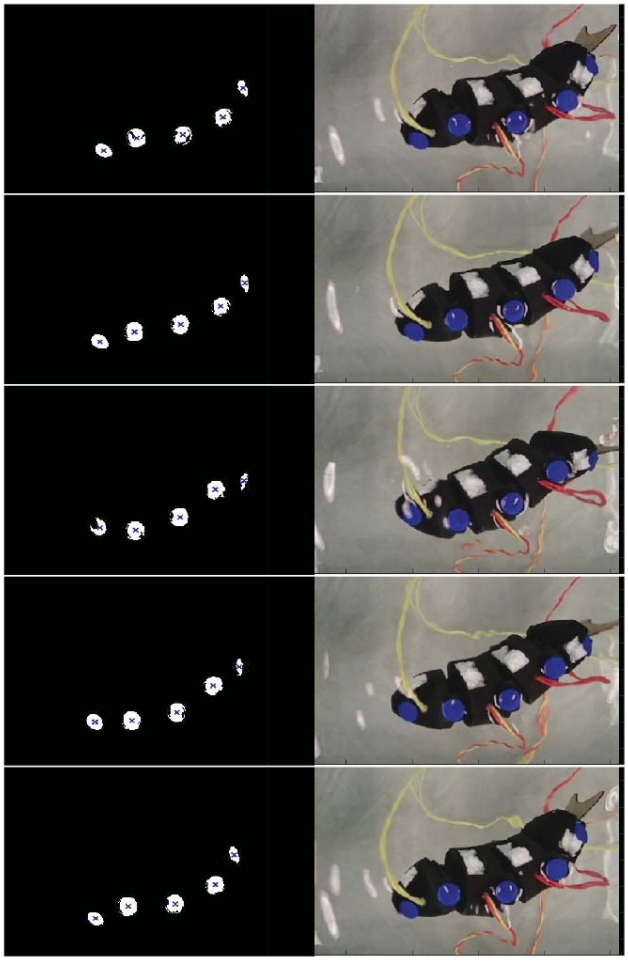

Fig. 16. Tests 1 and 2 (depicted) placed the first and last joints out of phase. When the head was fully turned to one side, the tail pointed in the opposite direction. The body then transitioned through an equilibrium state where all of the joints lined up to reverse the directions of the head and tail joints. The images on the left are the results of our object detection algorithm, run on the video frames depicted on the right. Each white blob corresponds to a pin, and a blue x marks the calculated centroid.

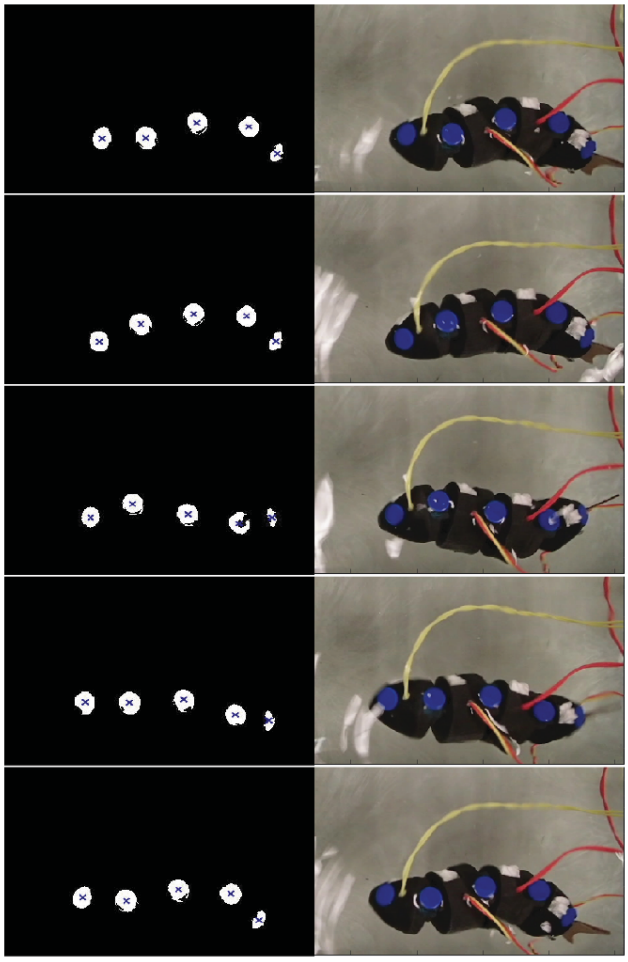

Fig. 17. In Test 3 and 4 (depicted), the first joint was out of phase with the middle. When the tail reached its maximum deflection, the head was moving towards the same side. This counterbalancing behavior was observed in previous works, [9], [2], and MORA's highest speed was achieved with this configuration.

TABLE II: Speed Results

	Average Speed (BL/s)
Test 1	0.26
Test 2	0.25
Test 3	0.29
Test 4	0.37

partially caused by an uneven distribution of mass in the body and momentary binding of the joint bearings. MORA was also sensitive to the pull of the signal wires, which counteracted its forward progression and often caused the actuators to turn or stall, interrupting the smooth undulation. The jagged motion observed in the digitized trajectories was also partly due to the discrepancy of the color-thresholding algorithm and the reflections of the water, which caused the digitization to drop some tracking pin positions or extract a fragmented blob, offsetting the position of the joint's centroid. Our future work includes removing the robot's tether to eliminate the added drag by fully integrating the electronics inside of the robot. Additionally, we are continuing to improve the mechanical design of the robot to make it easier to balance consistently.

VI. Conclusion

Our challenge was to create a low-cost, biomimetic actuation method for miniature robots. We demonstrated an undulatory swimming gait with our robot MORA by using a multi-jointed arrangement of magnet-in-coil actuators. The independently controllable joints allow for further experimentation in the choice of specific actuation settings. Moreover, a wide range of biomimetic motions is possible, including efficient straight line swimming, or bending the body for turning.

As we look to replicate more complex motions with MORA, finer control of the actuators' motions will be required. While a detailed characterization of the MIC can be derived, devel-

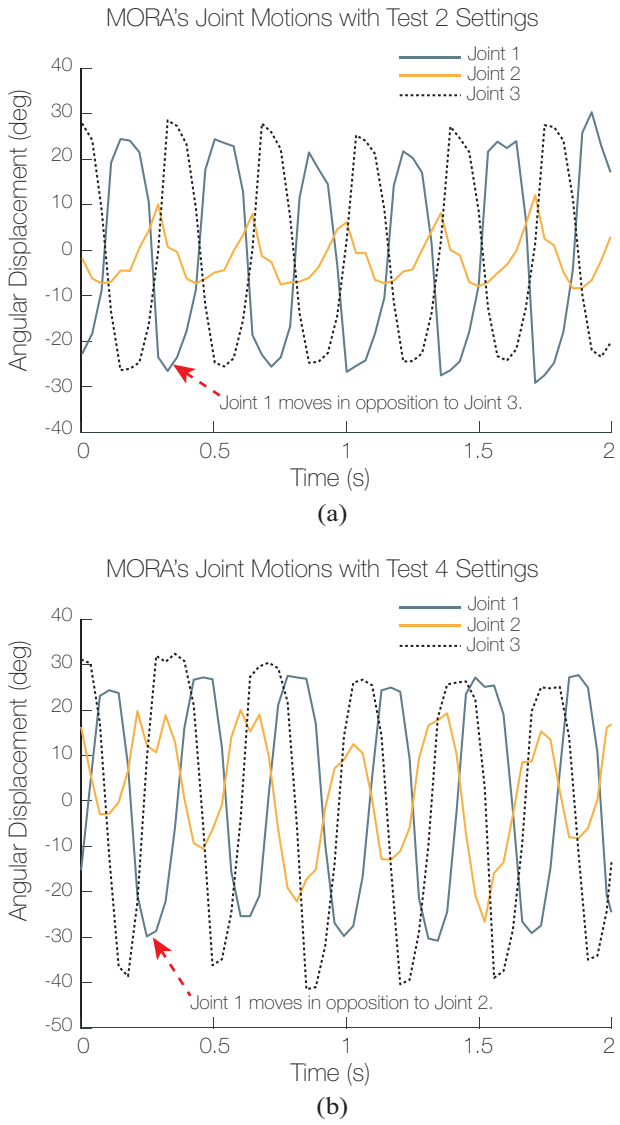

Fig. 18. The results of the joints' trajectory tracking for Tests 2 and 4 are depicted in (a) and (b) respectively. The amplitude of the middle joint in Test 2 is much smaller than in Test 4, most likely caused by the destructive interference between the first and last joint moving in opposite directions rather than counterbalancing each other.

oping a feedback system for it would provide a more dynamic response. We are investigating the use of a Hall Effect sensor to track the orientation of the magnetic field of the rotating magnet to be able to calculate the joint's position in real-time.

Improving MORA and its multi-jointed actuation method will enable the development of miniature underwater robots which show promise as robust and efficient agents in a distributed sensing network designed to navigate small spaces and sensitive ecosystems. It is vital to take into consideration the condition of the oceans with respect to their primary inhabitants for the responsible care of our planet, and the development of sensor platforms designed to gather data from this perspective is essential.

Acknowledgment

The lead author would like to thank Professor Radhika Nagpal for hosting her in the Self-organizing Research Lab at Harvard University with support from the Wyss Institute for Biologically Inspired Engineering during the summer of 2017. The authors would also like to acknowledge the Olin College of Engineering for ongoing support of undergraduate research.

References

[1] Robert K Katzschmann, Joseph DelPreto, Robert MacCurdy, and Daniela Rus. Exploration of underwater life with an acoustically controlled soft robotic fish. 2018.

[2] R. J. Clapham and H. Hu. isplash-i: High performance swimming motion of a carangiform robotic fish with full-body coordination. In *2014 IEEE International Conference on Robotics and Automation (ICRA)*, pages 322–327, May 2014.

[3] F. Berlinger, J. Dusek, M. Gauci, and R. Nagpal. Robust maneuverability of a miniature, low-cost underwater robot using multiple fin actuation. *IEEE Robotics and Automation Letters*, 3(1):140–147, Jan 2018.

[4] Large-amplitude elongated-body theory of fish locomotion. *Proceedings of the Royal Society of London B: Biological Sciences*, 179(1055):125–138, 1971.

[5] Rahul Bale, Max Hao, Amneet Pal Singh Bhalla, Namrata Patel, and Neelesh A Patankar. Gray's paradox: A fluid mechanical perspective. *Scientific reports*, 4:5904, 2014.

[6] DS Barrett, MS Triantafyllou, DKP Yue, MA Grosenbaugh, and MJ Wolfgang. Drag reduction in fish-like locomotion. *Journal of Fluid Mechanics*, 392:183–212, 1999.

[7] R. J. Clapham and H. Hu. isplash-ii: Realizing fast carangiform swimming to outperform a real fish. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1080–1086, Sept 2014.

[8] P Valdivia y Alvarado and Kamal Youcef-Toumi. Performance of machines with flexible bodies designed for biomimetic locomotion in liquid environments. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 3324–3329. IEEE, 2005.

[9] Jindong Liu and Huosheng Hu. Biological inspiration: From carangiform fish to multi-joint robotic fish. *Journal of Bionic Engineering*, 7(1):35–48, 2010.