Development of a Modular and Submersible Soft Robotic Arm and Corresponding Learned Kinematics Models

W. David Null1,2, James Menezes1, and Y Z1,2,3,4∗

Abstract—Many soft-body organisms found in nature flourish underwater. Similarly, soft robots are potentially well-suited for underwater environments partly because the problematic effects of gravity, friction, and harmonic oscillations are less severe underwater. However, it remains a challenge to design, fabricate, waterproof, model, and control underwater soft robotic systems. Furthermore, submersible robots usually do not have configurable components because of the need for sealed electronics and mechanical elements. This work presents the development of a modular and submersible soft robotic arm driven by hydraulic actuators which consists of mostly 3D printable parts and can be assembled or modified in a relatively short amount of time. Its modular design enables multiple shape configurations and easy swapping of soft actuators. As a first step to exploring machine learning control algorithms on this system, we also present preliminary forward and inverse kinematics models developed using deep neural networks.

I. INTRODUCTION

Submersible soft robots have interesting potential use cases which include exploring the deepest parts of the ocean [1], interacting with delicate sea creatures [2]–[4], studying the locomotion methods of underwater animals [5]–[11], maintaining and inspecting underwater equipment in the presence of radiation [12], and creating safer and more versatile minimally invasive surgery procedures [13]. However, it remains a challenge not only to fabricate and maintain, but also to model and control underwater soft robotic systems. With these challenges in mind, this work presents the development of a modular hydraulically-actuated underwater 2D soft robotic arm. In addition to the robot’s design, we present some preliminary forward and inverse kinematics models learned using neural networks.

Our modular design addresses common challenges which arise when designing soft robotic arms. Primarily, we call each segment in the arm a module and allow for arbitrary connection and disconnection of modules with minimal hardware alterations. Furthermore, any actuator on any module can be easily replaced. This is a desirable quality for research as actuators tend to break, leak, or show fatigue after long experiment sessions. It also may be necessary to test different actuator materials and it is helpful if this process is as painless as possible. Additionally, the electronics, sensing, valves, and fluid networks are local to each module of the robot. This allows the robot to take on many different configurations while only needing one interconnecting hydraulic line, power connection, and communication bus between each module. By contrast, other underwater soft robotic arms run a tube for each hydraulic chamber through the center of the robot and house all electronics and valves in a separate compartment [14], [15].

To address the problem of modelling and controlling soft robots, promising methods are found in the literature using machine learning [16], [17]. Neural networks can capture
more nuances of nonlinear dynamics than analytical models such as piecewise constant curvature (PCC) and Cosserat rod theory. However, many learned models can also run in real time which is a challenge for models generated using finite element methods (FEM) [18]. Inspired by the literature, this work presents preliminary kinematics modelling for the modular underwater soft robotic arm using neural networks. Both forward and inverse kinematics models are presented. These models form a benchmark for studying control techniques on this system.

II. BACKGROUND

Underwater environments are uniquely suited for the soft-body organisms we observe in nature and similarly for soft robots. This is partially due to the buoyant force of water counteracting the force of gravity and dampening the inherent oscillations that arise from actuating an elastic system. This work primarily focuses on soft hydraulic actuators [19] which are fitting for remote mobile underwater robotics applications [2], [3], [15], [20], [21] because they only require a pump to drive actuation which can draw water from the surrounding environment. This gives them the advantage of continuous operation and also keeps the necessary external hardware light-weight and compact. Additionally, they have a relatively simple waterproofing process. In contrast, soft pneumatic actuators require pressurized air canisters with limited capacity, and cable-driven soft actuators typically require multiple electric motors each with precise sensing which need to be housed in a box external to the robot itself.

Researchers have exploited these advantages by implementing several underwater soft robotic arms and manipulators [2], [3], [14], [15], [20]–[22]. Miniaturized versions of these robots have potential use cases in minimally invasive surgery (MIS) [13] where the robot can be surrounded by blood [23]. Furthermore, insights on the characteristics of underwater soft robots can also be applied more generally as Du et al. showed by improving a simulation model developed for an underwater robot [24].

Although hydraulically actuated underwater soft robots have immense potential, designing accurate and robust controllers for these systems remains a challenge despite their natural advantages. High nonlinearity and large degrees of freedom render analytical models, such as piecewise constant curvature (PCC) [25], insufficient for many real-world applications. The finite element method has been applied successfully to simulate soft robots [26] but suffers from large computational complexity which limits its usability in real time [18]. In traditional controller design, feedback techniques are used to compensate for modelling errors. However, despite exciting new developments in soft sensors research [27], [28], the sensors themselves are not readily available, and are difficult to fabricate, install, and calibrate.

At this point, researchers have turned to data-driven modelling and control strategies to capture the nuances of a system’s nonlinearity, while running in real time [17]. A
widely-used method is learning the forward and inverse kinematics of a soft robot with a neural network [29]. Deep neural networks and recursive neural networks have also been used to develop a dynamics model for use in a model predictive controller [30]–[32]. Additionally, reinforcement learning has been proposed as a model-free approach to the problem [17], [33]. An issue with these approaches is that a disturbance or change to the robot’s weight can cause the estimated models or policies to break down for lack of sufficient data. An interesting approach to solving this problem using online learning with Gaussian Process regression has been proposed to account for disturbances in system dynamics [34].

III. MATERIALS AND METHODS

This work presents the development of a submersible soft robotic arm with a modular design which is simple to assemble, and useful for research. The mechanical structure consists of 3D printed parts, and the actuator molds can also be 3D printed. The robot can collect internal state data from pressure sensors and solenoids positioned locally to each actuator and synchronize this with ground truth positioning information gathered from an overhead camera. The soft hydraulic actuators can be easily swapped with those of different shapes, sizes, and material characteristics. Additionally, the robot’s overall structure consists of individual segments, which we call modules, whose electronics and fluid networks can attach to each other. These modules can be added and subtracted to reach a desired robot configuration. This work also presents the development of preliminary learned forward and inverse kinematic models for use in robot modelling and control.

A. Experimental Setup

The entire system was built to work in a desktop configuration for rapid algorithm development and testing. In the desktop configuration, the robot’s base is mounted to a sheet of acrylic and submerged in about 5 cm of water along with the water pump which creates the hydraulic pressure (Fig. 1(f)). Computer vision (CV) markers are arrayed along the robot’s centerline (Fig. 1(c)) which are used to track the true shape of the robot. The water level is such that the CV markers rest just above the surface so they can easily be seen by the camera positioned directly above the robot (Fig. 1(b)). Computer vision was chosen over motion capture because of the lack of sufficient data. An interesting approach to solving the problem [17], [33]. An issue with these approaches is that a disturbance or change to the robot’s weight can cause the estimated models or policies to break down for lack of sufficient data. An interesting approach to solving this problem using online learning with Gaussian Process regression has been proposed to account for disturbances in system dynamics [34].

B. Modular Design

One of the key features of this soft robotic system is its modular design. The electronic, mechanical, and hydraulic components are all built so that modules can be stacked on top of each other to form different configurations for different experiments and applications. Similarly, the actuators themselves can be swapped out easily. This enables easier maintenance as well as quicker iterations on shapes, sizes, and material properties for testing. The system’s modularity creates potential for many different experimental parameters and applications.

B. Modular Design

One of the key features of this soft robotic system is its modular design. The electronic, mechanical, and hydraulic components are all built so that modules can be stacked on top of each other to form different configurations for different experiments and applications. Similarly, the actuators themselves can be swapped out easily. This enables easier maintenance as well as quicker iterations on shapes, sizes, and material properties for testing. The system’s modularity creates potential for many different experimental parameters and applications.

C. Mathematical Representation of Robot

A mathematical representation of the robot is presented in Fig. 4. The robot is constructed of two modules with the bottom-most marker of the base module $(m_{0,0})$ representing the origin. Each module has five CV markers labelled $m_{i,j}$ where $i \in \{1,2\}$ and $j \in \{1,2,3,4,5\}$. Each CV marker contains three circles whose centroid locations can be identified by the overhead camera. The red circle in the middle represents the center of that section of the robot and is given an $(x, y)$ location. All the red circles together represent the underlying centerline shape of the robot. The blue markers...
make it possible to receive the orientation of that section of the robot with respect to the horizontal. This orientation is represented by an angle $\phi$. All this information is captured in one marker tuple as $m_{i,j} = (x_{i,j}, b_{i,j}, \theta_{i,j})$. The pressure value for each actuator is denoted by the variable $P_{a,b}$ where $a \in \{1, 2\}$ for modules 1 and 2 and $b \in \{1, 2\}$ for left and right.

D. Basic Actuation Experiment

Experimental data shows the nonlinear characteristics and hysteresis associated with this robot. Fig. 5 shows the bend angles $\theta_1, \theta_2$ (defined in Fig. 4) of each module as a function of the pressure difference between that module’s actuators represented by $P_{i,\text{diff}} = P_{i,2} - P_{i,1}$ where $i \in \{1, 2\}$ for modules 1 and 2. The data represents five runs of pressurizing and depressurizing the right actuator of each module where the robot starts from a fully depressurized state corresponding to the coordinate $(0, 0)$ on the plot and the target max pressure difference for each experiment is 20 kPa. Module 1 took on average 37 sec. to fully pressurize, and 62 sec. to fully depressurize. On the other hand, module 2 averaged 42 sec. to pressurize and 73 sec. to depressurize but achieved a higher bend angle than module 1. A stronger pump could increase the speed of pressurization, however, quicker depressurization would require additional springs or a stiffer actuator material since this process relies on the elasticity of the actuators to force water out.

Fig. 5 gives insight into the hysteresis and nonlinear dynamics of the robot. Both modules need to overcome friction with the bed, fluid resistance, and the robot’s inertia. The effects are more severe for module 1 than module 2. During pressurization, module 1 starts to bend when $P_{1, \text{diff}} > 2$ kPa while module 2 starts to bend when $P_{2, \text{diff}} > 1$ kPa. Module 1 reaches a max bend angle of just 29° while module 2 reaches a max bend angle of 33° for the same pressure difference. During depressurization, it takes a much greater pressure differential for module 1 than for module 2 to move back to the starting position. In fact while module 2 is able to return completely to the starting position, module 1 comes to rest at a bend angle of 8.5°. If the robot was floating and there was no contact with the bed, we expect module 1 would eventually return to the home position. For a larger robot configuration with three or four modules, we expect that these effects would increase.

E. Preliminary Learned Kinematics Models

Two deep neural networks (DNNs) were designed, trained, and tested to model the forward and inverse kinematics of the submersible soft robot. The forward kinematics model takes as input previous and current pressure and solenoid values and outputs estimated CV marker positions for the centerline markers. The inverse kinematics model takes as inputs the centerline marker positions for a single time step and outputs estimated pressure values for each actuator.

Both networks were built in Python using the Keras package. Having a similar internal structure, they consist of 4 dense hidden layers each of which uses a relu activation function and is followed by a dropout layer with a probability value of 0.2. The dense layers have a size of 128, 64, 32, and 16 neurons respectively. The difference between the forward and the backward model are the input and output dimensions. In total, the forward network has 17,476 trainable parameters while the backward network has 13,620 trainable parameters. Both networks are trained with a mean squared error loss function and the “adam” optimizer. In prior work, smaller networks with 2 hidden layers and 36 neurons were used to generate forward and inverse kinematics models for a 3D pneumatically actuated soft robot arm [35]. For our submersible robot, through multiple trials we found that this larger network structure achieved lower training loss than smaller networks without overfitting.

1) Forward Kinematics Model: The shape of the robot at time $t$ is highly dependent on previous and current measurements of pressure in each actuator and the states of the in/out solenoid valves. We represent the actuator pressures at time $t$ by the vector $p_t \in \mathbb{R}^I$ where $i = 4$ for the 2-module robot configuration. Similarly, the binary states of the in/out solenoid valves at time $t$ are contained in the vector $u_t \in \mathbb{R}^J$ where $j = 8$ for the 2-module configuration. To achieve higher accuracy, we also consider previous time-steps with a back-step size of $\tau = 7$ and a total number of previous samples $n = 3$. Since data was collected at a frequency of 2 Hz, the DNN will take as input 4 samples of $p$ and $u$ over the last 10.5 seconds. The outputs of this neural network are the estimated $x$ and $y$ coordinates at time $t$ of the centerline markers represented by the vector $x_t \in \mathbb{R}^{2k}$ where $k = 10$ for the 2-module configuration. By contrast, for a 1-module configuration $i = 2$, $j = 4$ and $k = 5$, and for a 3-module configuration $i = 6$, $j = 12$ and $k = 15$. If the configuration of the robot is changed, the network needs to be retrained. The learned forward kinematics model is represented by $\hat{f}_{FK}$ and is used as follows:

$$x_t = \hat{f}_{FK}(p_t, p_{t-\tau}, ..., p_{t-n\tau}, u_t, u_{t-\tau}, ..., u_{t-n\tau}) \quad (1)$$

2) Inverse Kinematics: Our learned inverse kinematics model simply takes as inputs the $x$ and $y$ coordinates for each centerline marker corresponding to a desired shape represented by the vector $x_t \in \mathbb{R}^{2k}$ where $k = 10$ and outputs an estimated pressure vector $p_t \in \mathbb{R}^I$ where $i = 4$. 

Authorized licensed use limited to the terms of the applicable license agreement with IEEE. Restrictions apply.
Thus, a low level pressure regulation method is needed to use this model for control. Since this model does not consider previous state data, it will be less accurate, but simpler to use in practice because the programmer only needs to specify a desired shape and not a full shape trajectory. The learned inverse kinematics model are mostly concentrated below 1.5 kPa of error.

B. Discussion

Examples of well-performing and poor-performing estimates given by the learned forward kinematics model are shown in Fig. 6. The poorest estimates of the learned forward kinematics model only reach about 1.5 cm RMSE. These poor-performing cases seem to occur at the limits of the robot’s task space as in Fig. 6(g,h), or near the fully depressurized state as in Fig. 6(i). In the cases where the robot is near the limits of its task space, the issue could lie in not having enough training data for those regions. In the fully depressurized instance, the actuators in the first module are not able to drag the second module all the way back to the centered home position due to friction between the robot and the bottom of the tank. Thus, the pressure is equalized for an extended period, but the position of the robot is more uncertain. A potential solution to this issue would be to increase the stiffness of the actuators on the first module.

While the learned inverse kinematics model was not as accurate as the forward kinematics model, it was also not given any previous states as inputs. This is because it is not always possible to track the shape of the robot over time. Often the robot arm will be operating in a space where a camera cannot verify its full shape. Thus, only the target shape and the robot’s current pressure sensor and solenoid valve states are known. Even with simple inputs, the model performs relatively well. Examples of poor-performing and well-performing estimations is shown in Fig. 6.

V. CONCLUSION

This work presents an underwater soft robotic arm that is not only relatively simple to assemble but also modular and configurable and can be used in a desktop environment. The electronics and fluid networks in the robot are designed so that segments can be added or removed to form a desired robot size. Also, the mounting method used to attach the soft
actuators allows for swapping of different types of actuators which may vary in size, shape, and material characteristics. This also allows an actuator to be easily maintained or replaced if it leaks or degrades over time. In addition, learned forward and inverse kinematics models were developed using data collected from the robot. Some current limitations motivate upgrades to the design as future work such as experimenting with different size pumps, different actuator shapes, sizes, and stiffnesses, and adding a 3rd actuator to each module to enable 3D motion.

REFERENCES

[1] G. Li, X. Chen, F. Zhou, Y. Liang, X. Xiao, X. Cao, Z. Zhang, M. Zhang, B. Wu, S. Yin, Y. Xu, H. Fan, Z. Chen, W. Song, W. Yang, B. Pan, J. Hou, W. Zou, S. He, X. Yang, G. Mao, Z. Jia, H. Zhou, T. Li, S. Qu, Z. Xu, Z. Huang, Y. Luo, T. Xie, J. Gu, S. Zhu, and W. Yang, “Bioinspired soft robot in the Mariana Trench,” *Nature*, vol. 591, no. 7848, pp. 66–71, 2021.

[2] J. Luong, P. Glick, A. Ong, M. S. deVries, S. Sandin, E. W. Hawkes, and M. T. Tolley, “Eversion and Retraction of a Soft Robot Towards the Exploration of Coral Reefs,” *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 801–807, 2019.

[3] K. C. Galloway, K. P. Becker, B. Phillips, J. Kirby, S. Licht, D. Tchernov, R. J. Wood, and D. F. Gruber, “Soft Robotic Grippers for Biological Sampling on Deep Reefs,” *Soft Robotics*, vol. 3, no. 1, pp. 23–33, 2016.

[4] B. T. Phillips, K. P. Becker, S. Kurumaya, K. C. Galloway, G. Whitredge, D. M. Vogt, C. B. Teeple, M. H. Rosen, V. A. Piribone, D. F. Gruber, and R. J. Wood, “A Dexterous, Glove-Based Teleoperable Low-Power Soft Robotic Arm for Delicate Deep-Sea Biological Exploration,” *Scientific Reports*, vol. 8, no. 1, p. 14779, 2018.

[5] A. D. Marchese, C. D. Oonal, and D. Rius, “Autonomous Soft Robotic Fish Capable of Escape Maneuvers Using Fluidic Elastomer Actuators,” *Soft Robotics*, vol. 1, no. 1, pp. 75–87, 2014.

[6] Z. Shen, J. Na, and Z. Wang, “A Biomimetic Underwater Soft Robot Inspired by Cephalopod Molluscs,” *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2217–2223, 2017.

[7] Q. Shen, Z. Olsen, T. Stalbaum, S. Traoba, J. Lee, R. Hunt, K. Kim, and I.-K. Oh, “Basic design of a biomimetic underwater robot with switchable swimming modes and programmable artificial muscles,” *Smart Materials and Structures*, vol. 29, no. 3, p. 035038, 2020.

[8] J. Shintake, V. Cacucciolo, H. Shea, and D. Floreano, “Soft Biomimetic Fish Robot Made of Dielectric Elastomer Actuators,” *Soft Robotics*, vol. 5, no. 4, pp. 466–474, 2018.

[9] H. Feng, Y. Sun, P. A. Todd, and H. P. Lee, “Body Wave Generation for Anguilliform Locomotion Using a Fiber-Reinforced Soft Fluidic Elastomer Actuator Array Toward the Development of the Eel-Inspired Underwater Soft Robot,” *Soft Robotics*, vol. 7, no. 2, pp. 233–250, 2020.

[10] M. Krieg, I. Sledge, and K. Mohseni, “Design considerations for an underwater soft-robot inspired from marine invertebrates,” *Biospiration & Biomimetics*, vol. 10, no. 6, p. 065004, 2015.

[11] F. Giorgio-Serchi, A. Arienzi, F. Corucci, M. Giorelli, and C. Laschi, “Hybrid parameter identification of a multi-modal underwater soft manipulator,” *Robotics*, vol. 12, no. 2, p. 025007, 2017.

[12] O. D. Yirmibesoglu, T. Oshiro, G. Olson, C. Palmer, and Y. Menguc, “Evaluation of 3D Printed Soft Robots in Radiation Environments and Comparison With Molded Counterparts,” *Frontiers in Robotics and AI*, vol. 6, p. 40, 2019.

[13] M. Rumana, A. Darzi, and G. P. Mylonas, “Soft Robotics in Minimally Invasive Surgery,” *Soft Robotics*, vol. 6, no. 4, pp. 423–443, 2019.

[14] Z. Gong, J. Cheng, K. Hu, T. Wang, and L. Wen, “An Inverse Kinematics Method of a Soft Robotic Arm with Three-Dimensional Dimensional Locomotion for Underwater Manipulation,” *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pp. 516–521, 2018.

[15] S. Kurumaya, B. T. Phillips, K. P. Becker, M. H. Rosen, D. F. Gruber, K. C. Galloway, K. Suzumori, and R. J. Wood, “A Modular Soft Robotic Wrist for Underwater Manipulation,” *Soft Robotics*, vol. 5, no. 4, pp. 399–409, 2018.