Design and Control of Soft Biomimetic Pangasius Fish Robot Using Finray Actuator and Reinforcement Learning

Samuel M. Youssef, MennaAllah Soliman, Mahmood A. Saleh, Ahmed H. Elsayed, and Ahmed G. Radwan, *Senior Member, IEEE*

***Abstract*—Soft robots provide a pathway to accurately mimicking biological creatures and being integrated in their environment with minimal invasion or disruption to their ecosystem. These robots made from soft deforming materials possess structural properties and behaviors comparable to the bodies and organs of living creatures. However, they are difficult to develop in terms of integrated actuation and sensing, accurate modeling, and precise control. This article presents a soft robotic fish inspired from the Pangasius fish. The robot employs a finray actuator driven by a servo motor, to act as the body of the robot and provide the undulatory motion to the caudal fin of the fish. To address the modeling and control challenges, reinforcement learning (RL) is proposed as a model-free control strategy for the robot fish to swim and reach a specified target goal. By training and investigating the RL through experiments on the real hardware, we illustrate the capability of the fish to learn and achieve the required task.**

***Index Terms*—Soft Robotics, Underwater Robotics, Biomimetics, Reinforcement Learning.**

I. Introduction

UNDERWATER depths have proven to be very challenging environments for humans to venture into. Researchers and engineers strive to build underwater robotic systems to accomplish this dangerous endeavor. From oceanic investigation and marine life exploration, to execution of underwater missions and sample gathering, to monitoring and maintenance of offshore and underwater structures, many complex tasks need to be done in harsh unpredictable conditions. Leveraging the new technological advancements in biomimetics and soft robotics provides promising solutions to build robotic systems capable of operating more naturally and withstanding these harsh environments.

Studying the various biological marine creatures offers insights on the characteristics allowing them to live in and populate vast oceanic regions. Taking inspiration from the morphologies of underwater living organisms, their techniques for swimming and locomotion, and their sensory capabilities aids in the development of bioinspired robotic systems similar to these creatures, making these robots more suitable for underwater applications. Swimming motion amongst underwater creatures shows a variety of locomotion techniques, guided by the morphological structures and shapes of such creatures [1]. The majority of aquatic creatures possess compliant bodies and rely on their body deformation to generate the thrust needed for locomotion.

The field of soft robotics offers successful approaches for building bioinspired robotic systems in general [2], and more specifically robots inspired from biological marine creatures. The use of soft materials to develop robots with compliant bodies and large degrees of freedom can take us a step closer to mimicking creatures with complex locomotion [3]. Several attempts have been made to exploit their deformability to design biomimetic soft robots capable of imitating biological swimming motion [4]. One approach made use of hydraulic elastomers to develop a soft robotic fish capable of performing several swimming maneuvers [5]. Another bioinspired robotic fish uses Ionic Polymer-Metal Composite (IPMC) actuators as the pectoral and caudal fins [6]. Another team was able to mimic the cephalopod molluscs by using hydraulic smart soft bending actuators to build the tentacles that aid the cephalopod in maneuvering [7]. A brittle star-inspired soft robot uses twenty Shape Memory Alloy (SMA) wires to actuate five flexible legs and perform underwater crawling [8].

One of the biggest challenges in soft robotics are the modeling and control of these non-linear complex systems [9]. Researches have been tackling these challenges using various approaches [10]. Model-free control doesn't require a model or prior information about the system, but relies mainly on the input-output behavior collected directly from the system to learn an approximate representation of it. Reinforcement learning (RL) is one of these model-free control techniques that have been providing promising results in recent years [11]. RL is a data-driven learning process that depends on having the agent interact with its environment by taking certain actions and observing its new state. The agent is then given a reward based on the task it needs to complete and the RL algorithm learns a policy to map the state-action pairs. In particular, RL has been implemented for soft robotics control in general and in the case of underwater soft robotics specifically [12].

One group used a Q-learning algorithm with experience replay to maximize the swimming speed of a cuttlefish soft

Samuel M. Youssef is with the Smart Engineering Systems Research Center (SESC), Nile University, Sheikh Zayed City 12588, Egypt (email: s.youssef@nu.edu.eg).

MennaAllah Soliman, and Mahmood Abdallah Saleh are with Bio-Hybrid Soft Robotics Laboratory (BHSRL), Nile University, Sheikh Zayed City 12588, Egypt (email: msoliman@nu.edu.eg; mabdullah@nu.edu.eg).

Ahmed H. ELsayed is with Innovation Hub, Nile University, Sheikh Zayed City 12588, Egypt (email: aharbi@nu.edu.eg).

Ahmed G. Radwan is with the Nanoelectronics Integrated Systems Center (NISC), Nile University, Sheikh Zayed City 12588, Egypt, and also with the Department of Engineering Mathematics and Physics, Cairo University, Giza 12613, Egypt (email: agradwan@nu.edu.eg).

robot actuated by a dielectric elastomer (DE) membrane [13]. A soft robot actuated by shape memory alloys (SMAs) uses Q-learning to learn a control policy for end effector locomotion [14]. One approach used a deep deterministic policy gradient (DDPG) algorithm to learn a control policy for soft continuum arms [15]. However, training RL agents is a costly process in terms of computation time and resources, and it becomes more complex for soft robots due to their non-linear dynamics and elastic properties. To solve this problem, a research group implemented an RL method that ignores the soft materials properties and structure of the robot, and it was applied on the Honeycomb PneuNets soft robot [16].

In this paper, we propose a design for a biomimetic fish robot inspired from the pangasius fish, using the finray concept for soft body actuation to mimc the fish's body and tail undulation. We investigate the use of three RL algorithms to teach the robot to swim to a specific goal.

II. Methods

A. Pangasius Fish Morphological Analysis

In order to build a soft underwater robot that mimics fish locomotion, an actual fish is studied through visual motion analysis to obtain insight and parameters relevant to the design and control of the equivalent biomimetic fish robot. Markerless tracking was used by preparing a motion capture setup. It is comprised of water tank of dimensions 120 x 70 cm and a Logitech C920 visual monocular camera. The fish recorded for the analysis is from the Pangasius genus, a fresh water class of medium-large to very large shark catfishes, as shown in Figure 1.

1) Pose Estimation: Studying the swimming patterns and motion of the fish requires performing pose estimation on the recorded video data to track the deformation and motion of the fish's different body parts during its swimming sequences. The pose estimation step was performed using DeepLabCut, a deep learning platform for markerless animal pose estimation [17].

During the pose estimation process, three body parts of the fish are defined to be tracked: the head, the center of the pectoral fins, and the caudal fin. Several samples were taken from the captured videos and annotated with the body parts. A ResNet neural network with 152 layers is trained using the video streams to estimate the position of these points. The network was trained for 200000 epochs reaching training and testing errors approximately 3 and 6 pixels, respectively.

2) Swimming Analysis: After obtaining the positions of the needed body parts, further analysis is conducted on the predicted pose of the fish to investigate the important parameters responsible for the swimming motion of the fish, which should help design the biomimetic fish. Such crucial parameters include the fish's tail frequency and amplitude, and the resulting velocity at which the fish is able to swim due to its undulating motion. During carangiform swimming, the locomotion rely mostly on the undulation motion of the body and caudal fin (tail), while the pectoral, pelvic, and dorsal fins help the fish balance and swim up and down. By analyzing several sample videos of the fish's swimming, the tracking of the tail's motion, allows us to obtain the undulation frequency of the fish's body. While stationary, the frequency of the tail ranged from 0.7 to 2 Hz. During low-speed swimming, the frequency ranged from 1 to 2.5 Hz, while it reached up to 4.5 Hz during high-speed swimming. The attained speed during low-speed swimming was in the range of 5 to 6 cm/sec, and up to 65 cm/sec for high-speed swimming.

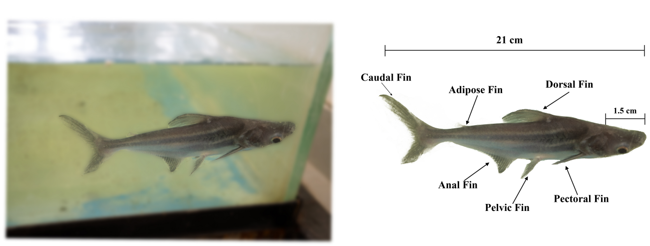


Fig. 1: Real Pangasius fish anatomy captured by image processing motion system that presents the real dimension and morphological structure for the fish.

III. Results

A. Soft Biomimetic Pangasius Fish Robot Design and Prototyping

The soft biomimetic Pangasius fish is designed based on the dimensions of the real fish that is captured using the vision system, as described previously. The robot dimensions is scaled to double dimensions of real fish. The design is distributed to main four parts; fish head, fish body, caudal fin and pectoral fin as shown in Figure 2 (a) and (b). The fish body is responsible on undulation motion needed to move fish in the water. Soft finray actuator is selected to mimic the needed motion. The finray actuator consists of a flexible outer body, rigid links between its segmentation and rigid connections between the servo motor - source of actuation - and the actuating points in the finray itself as shown in Figure 2 (c).

Different manufacturing techniques based on additive manufacturing are followed to produce the biomimetic Pangasius robot [18]. Due to high complexity of fish head part, selective laser sintering (SLS) is selected for the production using Sinterit Lisa Pro ^{TM}. The material used for SLS printing is PA12 Smooth, Nylon based material that is selected for its high durability. For flexible finray, high hyperelastic material is needed for its construction and due to its low of its complexity of design. Fused Deposition Modelling FDM 3D printer is selected to manufacture this part using Felix 4Tec ^{TM} with flexible material; Extrudr FLEX ^{TM}, medium material. Finally, high rigidity material is needed for rigid links and rigid connectors to withstand with tension forces exerted by servo motor. The material being used is glass-reinforced epoxy laminate material (FR-4) and is cut through CO2 laser machine.

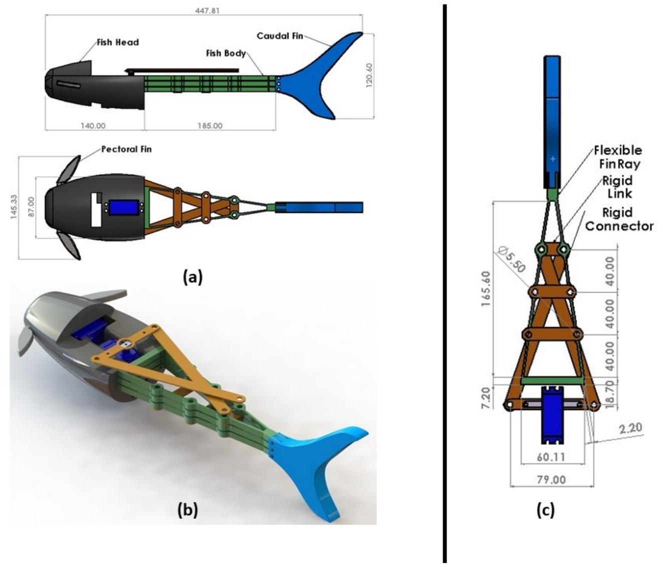


Fig. 2: Soft biomimetic Pangasius fish geometrical design. (a) Detailed design with approximate double scale of real fish dimensions. (b) Complete CAD of robot. (C) Finray actuator design and dimensions

B. Reinforcement Learning

The main objective of this experiment is to make the soft robot swim to a certain predetermined goal location in the tank. The setup is as shown in Figure 3. The tank and robot are monitored using a Logitech Brio camera at 60 fps that captures the environment and feeds the frames to DeepLabCut to perform pose estimation. To perform reinforcement learning training, generally simulation tools are first used to train the agent, then the learning is transferred to the actual robot. However, due to the complexity of simulating soft materials that exhibit high deformation and the fluid-structure interaction between the robot and environment, the RL training algorithm was implemented directly on the experimental setup. By using stable baselines 3 [19] on top of OpenAI Gym [20], the RL environment is built by defining the observations and actions spaces for the agent.

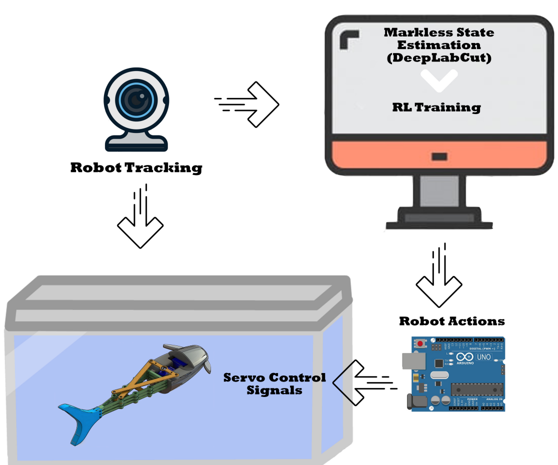


Fig. 3: Schematic of the experiment setup

The actual possible actions for the robot could be described as a continuous space of varying servo speeds and angles, up to the maximum values according to the servo's specifications. However, having a continuous action space could make the problem more complex to solve for the RL algorithm. Hence, the discretization of the actions would help simplify the task. Thus, the oscillation of the servo is fixed as the maximum travel of the servo, which is 130° according to the manufacturer's specifications. The servo oscillation speed becomes the only variable for the actions. By using the fish swimming analysis as a base, several oscillatory frequencies of the tail were chosen to be applied as the varying speed of the actuation servo motor. These speeds comprise the action space of the robot. A total of 10 actions are defined, ranging between 110 and 200 milliseconds to perform a tail stroke, with a 10 milliseconds step.

The observations space is comprised of several parameters related to the robot and its environment. First, the x and y positions are obtained from the state estimation performed through DeepLabCut for 3 points on the robot: the head, the servo horn, and the tail. The distances in the x and y directions between the robot's head and the destined goal point are also added. Finally, a queue of previous actions is appended to the state.

$$s_t = \{p_1(x,y)_t, p_2(x,y)_t, p_3(x,y)_t, \delta x_t, \delta y_t, a_{t-k}, ..., a_{t-1}\} \tag{1}$$

$$a_t = 0, ..., 9 \tag{2}$$

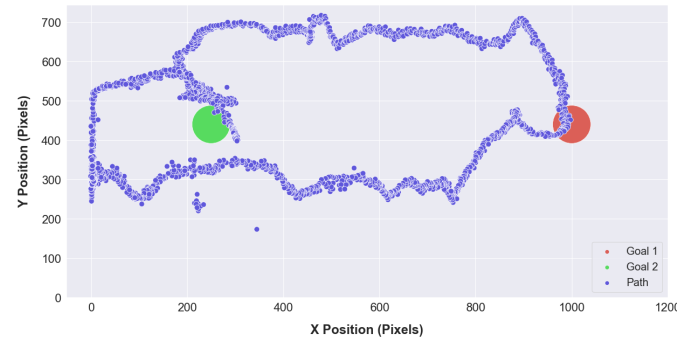
where s is the observation space. $p_1(x,y), p_2(x,y), p_3(x,y)$ are the x and y coordinates for the head, servo horn, and tail at step t, respectively. δx and δy are the x and y distances between the robot's head point and the current goal. a is the action space consisting of 10 actions from 0 to 9, corresponding to the servo speed ranging from 110 to 200 milliseconds, with an increment of 10 milliseconds. k is taken as 100, which is the predefined maximum episode length for this experiment.

Two goals are defined at the two ends of the tank. The robot's task is to reach the current goal, then the goal changes to the other end once the robot succeeds. To simplify the task, an error tolerance is defined and the robot is considered successful in reaching the target if it swims within a distance of 50 pixels away from the target. The reward function r defined to achieve the task is:

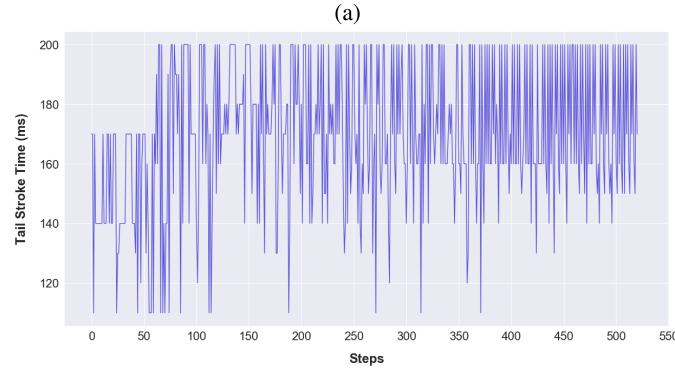
$$r_t = \alpha \times e^{\frac{-dist_t}{\beta}} - \phi(dist_t \times i_t) \tag{3}$$

where the reward is the exponential of the euclidean distance $dist$ between the robot and the goal and a penalty term as a factor of the distance and the episode step i. β is a reward decay factor, α is a reward multiplier, and ϕ is a penalty factor. An additional reward is added when the robot reaches the goal point. Due to the fact that the only terminal state for an episode is reaching the goal with no specific failure state, a maximum limit for steps per episode is defined and the penalty applied to the reward relied on the number of steps elapsed during the episode, increasing as the episode goes longer. The steps limit and the variable factors in the reward function were chosen by trial. Maximum steps per episode, α, β, ϕ, and goal reward are set as 100, 10, 200, 10^{-5}, and 200, respectively.

To train the robot, three RL algorithms are used to compare their performances. The first two are on-policy algorithms: proximal policy optimization (PPO) [21] and actor-critic (A2C) [22], which are policy-gradient methods. The third one is the deep q-network (DQN) [23], an off-policy value-based method. Due to the hardware limitation, each algorithm was trained for about 25000 steps to compare their performance. The mean reward per episode for the three algorithms during the training steps is shown in Figure 4. Taking into account these results, the PPO algorithm is chosen to be used for further training. Three PPO agents with different random seeds were trained for about 50000 steps each. The agents' losses during training are shown in Figure 5. The best agent is tested on the task to reach the goal within the least amount of steps. Figure 6 shows the path and actuation oscillation frequencies taken by the robot to reach the two defined goals.



(a)



(b)

Fig. 6: Results of the test run (**a**) The path taken by the robot to reach the two goals. (**b**) The sequence of actions represented as varying tail stroke time by changing the servo speed.

IV. Conclusion

In conclusion, this paper proposed a design for a biomimetic soft robotic fish inspired from the Pangasius fish. The soft robot utilizes a finray actuator made from soft elastic materials and is driven by a servo motor. The deformation of the soft finray tail of the robot fish mimics the undulatory motion of the Pangasius fish during carangiform swimming. The varying undulation frequency of the tail allows the robot to perform underwater locomotion similar to the actual fish.

Fig. 4: Mean episode reward during training of the three algorithms

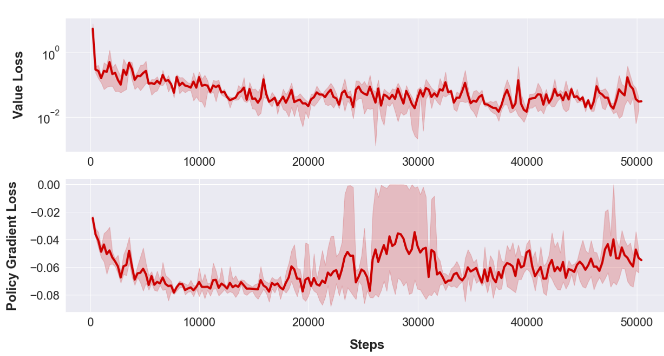


Fig. 5: Value and policy gradient losses during training of the three PPO agents

We also investigate the possibility of learning a control policy to teach the robot a certain task, which is reaching a specific target goal in this case. By using reinforcement learning (RL), the robot was able to learn to reach two different goals at opposite locations in the tank. Despite the complexity of the soft robot dynamics, the fluid-structure interaction, and the hydrodynamic forces, the learning process provides good results for the specified task. Training the agent for more steps would possibly allow it to exploit the environment more and learn more complex swimming behavior.

However, one of the main challenges was running the RL training directly on the robotic hardware. The number of training episodes and the possibility to train multiple agents becomes limited as the training is time consuming and it affects the lifespan, durability, and the properties of the soft material, changing its behavior with time. One solution would be the development of an appropriate physics simulator capable of simulating and performing RL on multi-body soft robots within underwater environments, then optimizing the learning through sim-to-real techniques. Future work could incorporate differentiable simulators and neural network hydrodynamic simulations to achieve this purpose.

Acknowledgment

The Authors would like to thank the Academy of Scientific Research and Technology (ASRT) for funding the project

#4779 and Nile University for facilitating all procedures required to complete this study.

References

[1] L. Maddock, Q. Bone, J. M. Rayner *et al.*, *The Mechanics and Physiology of Animal Swimming*. Cambridge University Press, 1994.

[2] S. Kim, C. Laschi, and B. Trimmer, "Soft robotics: a bioinspired evolution in robotics," *Trends in biotechnology*, vol. 31, no. 5, pp. 287–294, 2013.

[3] M. Hermes, M. Ishida, M. Luhar, and M. T. Tolley, "Bioinspired shape-changing soft robots for underwater locomotion: Actuation and optimization for crawling and swimming," *Bioinspired Sensing, Actuation, and Control in Underwater Soft Robotic Systems*, pp. 7–39, 2021.

[4] A. Ming and W. Zhao, "Design of biomimetic soft underwater robots," in *Mechatronic Futures*. Springer, 2016, pp. 91–111.

[5] R. K. Katzschmann, J. DelPreto, R. MacCurdy, and D. Rus, "Exploration of underwater life with an acoustically controlled soft robotic fish," *Science Robotics*, vol. 3, no. 16, 2018.

[6] T. Yang and Z. Chen, "Development of 2d maneuverable robotic fish propelled by multiple ionic polymer-metal composite artificial fins," in *2015 IEEE International Conference on Robotics and Biomimetics (ROBIO)*. IEEE, 2015, pp. 255–260.

[7] Z. Shen, J. Na, and Z. Wang, "A biomimetic underwater soft robot inspired by cephalopod mollusc," *IEEE Robotics and Automation Letters*, vol. 2, no. 4, pp. 2217–2223, 2017.

[8] Z. J. Patterson, A. P. Sabelhaus, K. Chin, T. Hellebrekers, and C. Majidi, "An untethered brittle star-inspired soft robot for closed-loop underwater locomotion," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 8758–8764.

[9] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.

[10] T. George Thuruthel, Y. Ansari, E. Falotico, and C. Laschi, "Control strategies for soft robotic manipulators: A survey," *Soft robotics*, vol. 5, no. 2, pp. 149–163, 2018.

[11] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[12] S. Bhagat, H. Banerjee, Z. T. Ho Tse, and H. Ren, "Deep reinforcement learning for soft, flexible robots: Brief review with impending challenges," *Robotics*, vol. 8, no. 1, p. 4, 2019.

[13] T. Yang, Y. Xiao, Z. Zhang, Y. Liang, G. Li, M. Zhang, S. Li, T.-W. Wong, Y. Wang, T. Li *et al.*, "A soft artificial muscle driven robot with reinforcement learning," *Scientific reports*, vol. 8, no. 1, pp. 1–8, 2018.

[14] W. Liu, Z. Jing, H. Pan, L. Qiao, H. Leung, and W. Chen, "Distance-directed target searching for a deep visual servo sma driven soft robot using reinforcement learning," *Journal of Bionic Engineering*, vol. 17, no. 6, pp. 1126–1138, 2020.

[15] S. Satheeshbabu, N. K. Uppalapati, T. Fu, and G. Krishnan, "Continuous control of a soft continuum arm using deep reinforcement learning," in *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2020, pp. 497–503.

[16] H. Zhang, R. Cao, S. Zilberstein, F. Wu, and X. Chen, "Toward effective soft robot control via reinforcement learning," in *International Conference on Intelligent Robotics and Applications*. Springer, 2017, pp. 173–184.

[17] A. Mathis, P. Mamidanna, K. M. Cury, T. Abe, V. N. Murthy, M. W. Mathis, and M. Bethge, "Deeplabcut: markerless pose estimation of user-defined body parts with deep learning," *Nature Neuroscience*, 2018. [Online]. Available: https://www.nature.com/articles/s41593-018-0209-y

[18] Y. L. Yap, S. L. Sing, and W. Y. Yeong, "A review of 3d printing processes and materials for soft robotics," *Rapid Prototyping Journal*, 2020.

[19] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021. [Online]. Available: http://jmlr.org/papers/v22/20-1364.html

[20] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 2016.

[21] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[22] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*. PMLR, 2016, pp. 1928–1937.

[23] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.