Proceedings of the 43rd Chinese Control Conference
July 28-31, 2024, Kunming, China

Reinforcement Learning Methods in Robotic Fish: Survey

Penghang Shuai[1,2], Haipeng Li[1]*, Yongkang Luo[3], Liangwei Deng[1,2]

1. The Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, P.R.China

2. School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 101408, P.R.China

3. State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, P.R.China

E-mail: {shuaipenghang2022, haipeng.li, yongkang.luo}@ia.ac.cn

Abstract: Reinforcement learning control has gained increasing attention in the field of biomimetic robotic fish, due to its advantage of universal applicability without prior knowledge of dynamic modeling. In this paper, we review the research work on the application of reinforcement learning in the field of robotic fish. We present and discuss the general model of applying reinforcement learning approaches to robotic fish control, such as the design of environment, state, reward and the selection of reinforcement learning algorithm. Furthermore, we propose to divide the typical tasks of reinforcement learning applied in robotic fish into single control and swarm control, and review the recent advancements separately.

Key Words: Reinforcement Learning Control, Robotic Fish, Single Control, Swarm Control

1 Introduction

Robotic fish generally achieve various swimming purposes through the flapping bioinspired structure such as body, caudal fin and pectoral fin[47]. Compared to traditional underwater vehicles that use propellers, robotic fish can achieve higher maneuverability with low noise. Considering disturbances from the complex hydrodynamics around fish, it is essential for controllers to have the ability to adapt to disturbances despite inaccuracy in the dynamic model of robotic fish. The challenges in improving motion performance of robotic fish arise from the following aspects:(1)The dynamics of the biomimetic thruster-propellers are highly complex, and establishing an accurate dynamic model is difficult or even impossible.(2)The ocean environment that robotic fish operate in is characterized by various uncertainties such as ocean currents, eddies, tides, etc., making it a chaotic system.(3)There are also uncertainties regarding the stability and reliability of robotic fish devices. To address these challenges, with the development of artificial intelligence, researchers begin to try intelligent methods to control robotic fish. As a model-free approach, reinforcement learning(RL) has attracted more and more attention in recent years. As controller is trained by trial and error of an $agent$ interacts with $environment$, reinforcement learning approach has the great advantage that the control objective can be achieved without prior knowledge of robot's dynamic model. RL methods have been widely applied in robot control fields including quadruped robots[45] and snake robots[46], which demonstrated satisfactory performance in complex environments where traditional control methods fall short.

The impact of current around swimming robot play a import role in dynamic models of robotic fish. Unfortunately, the understanding about hydrodynamic is not yet clear enough to accurately analyze the effects of fluid on robotic fish control, this is also the most obvious difference of applying RL in robotic fish compared to other fields of robotics. The abovementioned challenges can be possibly addressed by using RL approaches for robotic fishes, since that robotic fish continuously interact with the environment to learn optimal policy for specific control task. It is meaningful performance if the motion control of robotic fish can obtain the adaptive ability to various uncertainties through learning.

The rest of this paper is organized as follows. In Section II, we briefly introduce the framework to RL in robotic fish control, together with design details about framework. Then, Section III introduces typical robotic fish control tasks applied RL methods which are divided into single and swarm control. Finally, Section IV concludes this survey.

2 The Design of Reinforcement Learning on Robotic Fish

A standard reinforcement learning can be regarded as a sequential decision process. As illustrated in Fig.1, a basic process of reinforcement learning is that an *agent* gets *state* from *environment*, then takes an *action* based on the control policy π and receives *reward*. The *environment* will be changed its *state* by *agent*'s *action* with a transition probability $p(state'|state, action)$. The goal of reinforcement learning is to find an optimal policy that maximizes the expected *reward* for *agent*. Before using RL methods, usually the problem is defined as a Markov Decision Process(MDP) with a set of $state \in S$, $action \in A$, $p \in P$ and $reward \in R$. The MDP can be defined as a tuple (1)

$$D \equiv (S, A, P, R) \tag{1}$$

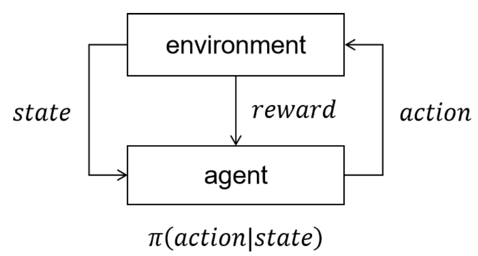

Fig. 1: A basic process of reinforcement learning.

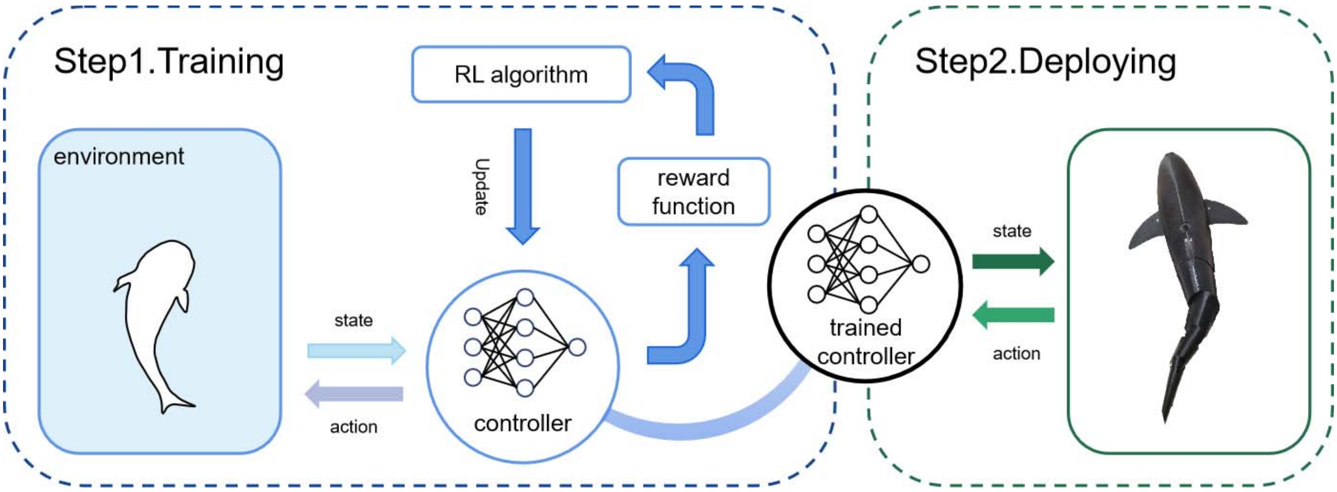

Fig. 2: General learning framework of generating controller for robotic fish by RL approach. Firstly, controller constantly interacts with an environment, and it is trained by certain RL algorithm. Until termination of training, the trained controller is deployed in robot system.

In the robotic fish control, the $state$ refers to the performance of the robotic fish in an underwater environment and its perception of water conditions, such as water velocity, pressure information or fish position. The $action$ typically represents variable parameter in the controller that can alter the posture of the robotic fish such as swing frequency. The $reward$ is manually set as scalar feedback to evaluate the performance of robotic fish in control tasks.

The framework of RL methods applied to robotic fish can be summarized as Fig.2, which consists of two phases, training phase and deploying phase. During the training phase, action signals are sent by the controller to guide the behavior of the robotic fish. The reward function is formulated based on control objectives, where both state and action signals of controller will be used as the input of reward function to generate rewards. The controller is continuously updated through RL training until it maximizes expected values. At this point, the trained controller is considered to be able to control robotic fish to perform specific control tasks within its environment. In deploying phase, the model of trained controller is deployed onto the real system for task execution.

2.1 Design of State Space

In reinforcement learning (RL)-based control, the state space encompasses information obtained from environment which could be perceived by agent and being changed due to its output action. As the basic information of training system, state is utilized for updating control policy and evaluating long-term benefits. The quality of state space design directly influences whether the RL training can converge, the convergence speed and the final performance. Similar to feature engineering in deep learning, designing the state space involves manually selecting relevant information suitable for the control task. There is no fixed model outcome for designing the state space of robotic fish when adopting RL approach; different platforms and tasks require different state space modes. In robotic fish field, most researchers first consider specific control tasks before designing state space. For instance, in path-following task, state space included the attitude and distance from the target trajectory of fish[13]; whereas in gliding task, researchers designed the state space encompassed variables like the speed and pitch angel of fish[14]. In real-world scenarios, robotic fish can acquire state information through onboard sensors with design of state space around these sensing units(see Fig.3a). Liu's[15] bionic robot dolphin addressed the path-following task using RL method, where pose information of the robot along with relative distance from desired paths were considered. Therefore, the posture data obtained from inertial sensors on the dolphin as well as distance between fish position and target trajectory derived from solving algorithms were added to the state space. Furthermore, the required data can also be obtained by camera suspended over the pool that recognizes the pose and position information of robotic fish(see Fig.3b). The camera outside fish body was the data source of state information[11]. However, it is natural to consider camera outside the fish body is uneasy to deploy in wild.

The swimming behavior of robotic fish is influenced by various complex factors in the water during swimming, among which the impact of water flow is crucial and should not be ignored in the state space design of reinforcement learning control. However, quantifying this influence proves challenging due to its inherent instability, lack of direct perception and detection convenience. Based on the dynamic model of biomimetic robotic fish, the influence of water flow can be reflected in the attitude, position, speed and other motion status of the robotic fish.

2.2 Design of Action Space

In RL method, action space is composed of a set of executable actions being output by agent based on environment information, which can be categorized into two type: discrete and continuous[50]. The objective of reinforcement learning is obtaining an optimal policy through learning to maximize expected rewards for the output actions of agent's controller. The controller of robotic fish changes some parameters, for instance the angle of the fin actuated by motor, to achieve different posture, and swimming speed, heading angle, etc. The design of action space incorporates these pa-

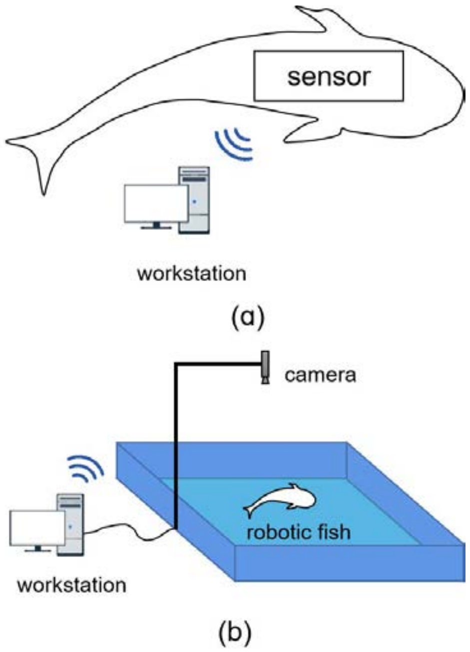

Fig. 3: (a)State information from sensor in robotic fish.(b)State information from overhead camera.

rameters, which is closely related to the swimming model employed by each robotic fish. Zheng et al.[9]developed a RL-based control approach, and utilized it on a boxfish-inspired robotic fish which had a single tail driven by a servomotor to propel forward in water. By altering the frequency, compensation, and amplitude of the caudal tail motion, different robot behaviors can be achieved; thus adding deflection angle of fin into the action space. Inspired by jellyfish, Youssef et al.[1] designed a robotic fish with soft-rigid hybrid structure, which generated propulsion force through periodic flapping of a soft tail. Therefore, they included different frequencies of servo in their action space design. In multi-joint robotic fish models, central pattern generator(CPG) is a common approach to generate rhythmic control signals with few adjustable parameters producing diverse periodic signals. Zhang et al.[10] incorporated one parameter from CPG equations into their action space design to enable robot fish to change swimming posture during RL training process. In practical applications, there are limitations on actions due to potential motion interference or damage risks posed on fish. So it is necessary to consider maximum angles for actuators like servos when designing an appropriate action space within RL training framework.

In practical environments, certain motion statuses such as frequency and amplitude of fish tail flapping are continuous values. However, continuous action spaces make the training process more complex for RL algorithms. Therefore, in some RL training tasks, the action spaces are discretized to simplify the training process. Youssef et al.[1] extracted the action space of the original continuous servo oscillation speed equidistantly, and finally discretized them into several actions. Yu et al.[16] discretized both state and action spaces in their design and controlled a robotic fish to hit a ball on the water surface. To make this discretization process more reasonable, Yu utilized fuzzy logic methods to discretize both continuous state and action spaces.

2.3 Design of Reward Function

Reward is a scalar feedback signal in RL algorithms to evaluate the output action of controller. The reward function should be designed to match the state space, with higher reward to state closer to the task goal, thereby incentivizing trained agent to take action that bring it closer to achieving the goal. Moreover, there is no fixed form for designing reward function, and the same optimal control policy can be learned by different reward functions. The design of reward function can consist of multiple parts. Costa et al.[18] aimed at achieving controlling fish swarm to form a specific formation using RL approach and then designed the reward function with four parts: (1) penalizing fish collisions when their distance falls below a threshold value; (2) negatively correlating rewards with inter-fish distances, encouraging proximity; (3) positively correlating rewards with rotational movements of fishes, promoting more rotations; and (4) imposing penalties when fish speed drops below a certain threshold value to prevent stagnation. Costa tested the reward function in four different group formations and achieved ideal results using RL approach. For the similar control task of robotic fishes, Zhang et al.[12] only considered two parts in the reward function, penalizing deviations from circular trajectories and incorporating penalties related to inter-fish distances — also achieving ideal results.

Although the control tasks of the abovementioned researches are different, they both belong to the collective swimming control with varying reward functions, and both achieved optimal results. This demonstrates that the design of reward function format and specific parameters setting are not fixed, allowing researchers to carry out various reward function forms of trial combined with control task and training effects.

2.4 Design of Training Environment

During the training phase of RL framework(see Fig. 2), the trained agent constantly interacts with the environment. The environment is changed due to the action of robotic fish and transmits information to agent, which will affect next action output of controller based on policy π. Unlike fully virtual environments in video games, the trained robot controller will be deployed in real-world system. Therefore, the training environments used for robot control tasks in RL approach, whether simulation or reality, should be as close as possible to the physical scenarios.

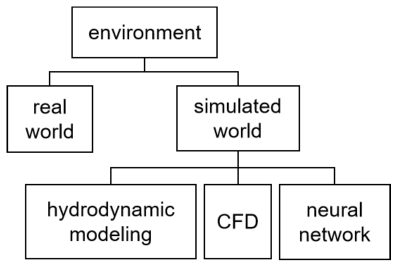

Fig. 4: Category of environment used for reinforcement learning on robotic fish.

Some teams used RL methods to control robotic fish through training in reality environment. For instance, Youssef et al.[1] designed a Pangsius-inspired robotic fish and employed RL approach to achieve the target-reaching task. The robot solely relies on a servo as its driver, swimming via the fin ray effect(FRE). Youssef directly placed the robotic fish in the water tank for training, and the relevant data was feedback to the workstation which was training the policy network in real time. Subsequently, the workstation continuously output the executable action signals to control the robot's movement. Yan et al.[2] created a bionic robot shark, which interacted with environment in the real water tank to complete the training of RL algorithm, and achieved the autonomous navigation task. Consequently, Yan compared the results from placing the RL training processes in the simulation versus reality environments; later experiments yielded better navigation efficiency. Direct training in real-world eliminates prior knowledge requirements for building simulated environments. Additionally, the controllers trained in reality environment exhibit better robustness and control effects. However, RL training requires large amounts of interaction data with environment, which means that it will take a lot of time to operate robot; unpredictably output actions may damage robots during their learning process as well. These time and hardware costs make this approach difficult to implement despite its effectiveness. Until now, only a little studies have successfully trained robotic fish directly within real world using RL methods.

Utilizing simulation environments for reinforcement learning training is a viable approach. While real-world training may save time and hardware costs, the gap between simulation and reality can make transferring well-trained networks from simulation to practical application more challenging or even unsuccessful. The goal of simulation environment construction is reducing reality and simulation gap caused by the differences between simulations and physical systems. The construction of hydrodynamic model can simulate the underwater environment, which usually includes analytical and numerical analysis methods. Verma et al.[3] conducted direct numerical simulations(DNS) of Navier-Stokes equations(NS) to construct a simulation environment for fish, and used RL algorithm to train fish to learn the most energy-efficient swimming method behind leader fish. Computational fluid dynamic(CFD) can be used to identify the parameters of the dynamics model of robotic fish through numerical calculation of the constructed hydrodynamic mathematical model. Some teams used CFD to build training environments for robotic fish in RL methods. Zhang et al.[10] proposed the Tri-S system as a framework of RL training and deploying on physical robots; CFD software was used in Tri-S as a simulation environment during training phase. Afterward, Zhang deployed the network model trained in the simulation environment onto the physical system where the robotic fish successfully completed the tasks of path-following and pose control. The result also demonstrated that the training in CFD environment brought small physical gap between simulation and reality. Due to the current limitations, there remains a big gap between this virtual and real environments. Additionally, CFD environments require substantial computing power, resulting in longer training times for simulation environments. At present, only a handful of studies have utilized CFD as a virtual environment for training, and got successful deployment of trained network on physical robotic fish.

With the development of machine learning, fitting the hydrodynamic model by artificial neural network(ANN) is also a method that can be tried. Collecting the data of water around the robotic fish, analyzing the data, the end-to-end hydrodynamic model can be realized through ANN. If the learning effect of the data is ideal, using the neural network can fit any function. This method can have faster calculation speed and greater accuracy than CFD method, but it also requires more learning skills and data. Xie et al.[10–13] have done a lot of works on using neural networks as a RL training environment for robot fish control, and also achieved application on physical systems. Typically, they modeled the boxfish and the lateral line system around the fish that senses water pressure, using reinforcement learning to reach desired angle of attack in a varied flow field. They collected the water pressure data around the body of the robotic fish for training the hydrodynamic model of the robot fish, and the trained neural network was the mapping function between the environmental information and the status of the fish, which was used as the virtual environment for reinforcement learning training of the robotic fish[9]. Since the training environment of reinforcement learning was directly from the neural network trained by the data collected in the real environment, the gap between virtual and real was small, and the direct application of the trained controller still showed good performance in reality. In addition to boxfish robot, they applied ANN as a training environment of robotic fish, and achieved tasks such as posture adjustment, swarm control, and path-following. Generating the mapping function about the real environment by training the neural network can save the costs and a lot of time for data collection of the interaction between the fish and the underwater environment. Moreover, because the data directly comes from the working environment, the difference between the virtual and the real is small under the condition of sufficient training amount, and the success rate of sim-real transfer is better than other.

2.5 Selection of Reinforcement Learning Algorithm

Reinforcement learning field encompasses a variety of algorithms[48, 49]. In the early stages, due to the limitation of computing power, Q-learning, Sarsa, and policy gradient methods[55] which are suitable for discrete state space and discrete action space had more applications. Deep Q network (DQN)[52], which uses neural network to fit the state-action value function, can be applied to reinforcement learning training tasks with continuous state space. Deep deterministic policy gradient (DDPG)[54] using multiple neural networks is used in continuous action space and continuous state space tasks. DDPG has been widely used in the field of robotic fish. It should be noted that there is no optimal reinforcement learning algorithm that can solve all tasks. Therefore it is necessary to experiment with different algorithms and choose the most appropriate one [1].

Some studies have also applied other methods to improve RL controls. For example, when faced with sparse reward spaces where it is difficult for the controller to learn an effective control strategy. Course-based reinforcement learning[19] and imitation learning[11] can alleviate this problem.

For robotic fish swarm control, many researchers still applied the above single-agent reinforcement learning algorithms for training, only a few mentioned the multi-agent reinforcement learning algorithms. Zhang et al.[12] explored varies multi-agent reinforcement learning algorithms in the robotic fish swarm control task, such as C2VDN, VDN and MADDPG algorithms. The utilization of multi-agent reinforcement learning algorithms is suitable for the scenarios involving multiple fish, and further research is required to enhance the application of these algorithms in the robotic fish swarm control.

3 Application of RL Methods in Robotic Fish

With the rapid development of reinforcement learning, there has been a growing body of research on the application of reinforcement learning in the past decade. With the key words of reinforcement learning and robot fish, total of 44 papers were retrieved, among which 15 focus on the swarm control of robotic fish, while 29 explore the single control of robotic fish. Swarm control involves the multiple robotic fish, such as controlling the fish to swim in a certain formation, or adjusting the movement of the fish to obtain the most energy-efficient swimming mode in the tail vortex of the leading fish. There are many kinds of individual control tasks, such as controlling the robot fish to swim at a specified speed, controlling the fish to swim to a specified position, or adjusting the pose of the fish, etc.

Different robotic fish mechanisms and control tasks were mentioned in these papers. It is not easy to come up with a distinct all-compassing taxonomy of all RL algorithms for robotic fish manipulation. To cover more works, we propose categorizing these papers into single and swarm control. Fig.5 shows how we categorize these tasks by separating the number of controlled agents.

In the context of applying reinforcement learning control to robotic fish, computer simulation have predominantly been conducted. However, the big gap between the dynamics model of robotic fish under simulation and the real environment can not be ignored. Further investigation is required to determine whether reinforcement learning can outperform alternative control methods in real underwater environment. Table 1 lists some typical tasks and their characteristics for both categories.

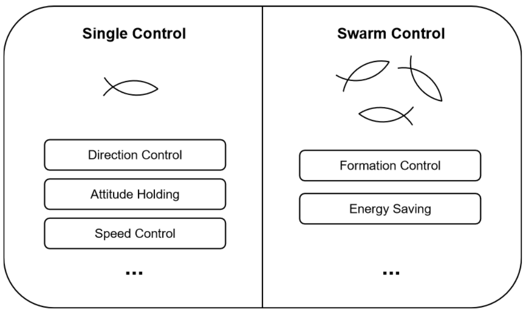

Fig. 5: Category of control task with RL-based methods.

3.1 Single Control

Tasks such as trajectory tracking control, speed control, attitude control, and direction control fall under the domain of motion control in robotic fish systems, which are also commonly addressed in the literature reviewed. In trajectory tracking task, by adjusting the control parameters governing swimming angle and position, the fish can closely follow the specified trajectory. Attitude control means controlling robotic fish swimming to the specified position and holding a certain attitude without a predefined path. Zhang et al.[10] constructed a learning framework for motion control of robotic fish from simulation to reality. Initially, the neural network-based control model of robotic fish was trained in the simulation environment until the optimal performance achieved, and subsequently it was put into CFD software environment to continue training. The resulting controller model that achieved optimal control effects in both simulation environments was directly transferred to the physical robot fish for executing tasks involving Bezier curve path tracking and pose adjustment with success.

Speed control refers to the fish can swim at a desired speed by adjusting its swimming strategy. Rajendran et al.[20] utilized a kind of artificial muscle, super-coiled polymers (SCP), as the fish body structure, and analyzed its dynamic model based on the three-joint structure of the fish body. Rajendran developed a one-dimensional dynamic model as the simulation environment for reinforcement learning training, and achieved controlled swimming at specific speed in computer simulation.

Direction control of robotic fish involves manipulating the fish to perform forward/backward and turning motions. The influence of water flow during fish swimming is a crucial factor that cannot be overlooked. Zhu et al.[6] employed CFD approach, immersed bounding-lattice Boltzmann method (IB-LBM), to construct a simulation environment. The vortical flow in the simulation environment would affect the swimming behavior of the fish, which needed to learn the effective navigation strategy for reaching specific destination points in the flow field. By utilizing CFD to create the environment, the performance of robotic fish in real world is better to understand and optimize. In this simulation setup, the fish employed tail flapping as its actions. After training with RL algorithm, the final simulation effect showed that the fish learned to swim in the specified direction through subtle tail movements.

In addition to the control tasks mentioned above, there are also tasks such as robotic fish cruising that demonstrated the efficiency of reinforcement learning methods. Yan et al.[2] designed a robotic shark which could modify the waveform of body undulation by CPG model. Within a pool, the robotic fish was required to cruise without touching the walls and bottom of the pool. To facilitate this, Yan constructed an expert experience buffer, and derived an initial control policy through imitation learning. Subsequently, reinforcement learning training was employed to update the policy. Finally, the robotic fish acquired a more efficient strategy for task completion. The utilization of imitation learning accelerated the convergence speed.

Table 1: Typical control task of robotic fish with RL approach.

Years	Reference	Task	RL Algorithm	Computer simulation	Practical application
Single Control					
2023	Duraisamy et al.[17]	Speed control	TD3	✔	✔
2023	Yan et al.[2]	Navigation control	DDPG+IL	✖	✔
2022	Youssef et al.[1]	Goal tracking	PPO	✖	✔
2022	Zheng et al.[9]	Attitude holding	DDPG	✔	✔
2022	Dong et al.[14]	Gliding motion optimization	DDPG	✔	✔
2021	Yu et al.[43]	Goal tracking	DDPG	✖	✔
2020	Zhang et al.[13]	Path following	A2C	✔	✔
2006	Shen et al.[28]	Obstacle avoidance	Q-learning, SARSA	✔	✖
Swarm Control					
2023	Sun et al.[21]	Formation control	D3QN	✔	✖
2022	Yu et al.[40]	Self-organization research	DQN	✔	✖
2021	Li et al.[8]	Energy saving	DQN	✔	✔
2021	Zhang et al.[12]	Formation control	C2VDN	✔	✔
2018	Verma et al.[3]	Energy saving	Unknown	✔	✖
2016	Yu et al.[16]	Cooperative behavior learning	Q-learning	✖	✔
2014	Gazzola et al.[23]	Formation control	Q-learning	✔	✖

3.2 Swarm Control

The swarm control of robotic fish involves multiple robotic fish. A common task in swarm control is formation control task, where the fish are required to swim in the specified direction or the desired formation trajectory. Compared to the single control of robotic fish, fish in swarm control are more sensitive to vortical flow generated by other fish. And researchers adopted the swarm control methods to investigate whether fish can acquire strategies for minimizing energy loss in uniform flow.

The control task of Zhang's[12] work is making three robot fish swim in a circle formation, and each robot fish swimming was controlled using CPG model. Since the swarm control task involves the training of multiple agents, Zhang chose various multi-agent reinforcement learning algorithms for comparison experiments, and finally selected an optimal strategy obtained by C2VDN algorithm to deploy to physical robot fish without additional engineering fine-tuning. At the beginning, the robotic fish were placed at any position in the pool. After a period of time, the fish swam circle formation and continued to swim in this formation. Sun et al.[21] constructed a simulation environment through CFD. The swarm of three fish swung their tails in the environment, and was required to swim with keeping a triangular formation. Sun adopted D3QN as the RL algorithm for this task. In order to accelerate the training, the imitation learning was added to the process of RL training, and the action of the leader fish in the triangle formation could be selected by the follower fish as an expert action. The final experimental results showed that the fish maintained stable formation.

In swarm control, fish are highly sensitive to vortex street caused by the swimming of other fish. The study of group robotic fish control can help to save energy through adjusting swimming mode. Researchers primarily focus on the physical properties of the water flow and their impact on the fish bodies. Novati et al.[22] used remeshed vortex methods to solve the Navier-Stokes equations in velocity-viscosity form, and created a simulation environment for RL training. In Novati's simulation experiment, two fish swam forward in a leader-follower formation, which altered swimming mode by adjusting swing frequency. The leader fish continuously generated vortex street and influenced the follower fish. Through RL training, the follower learned the energy-efficient swimming strategy within the leader fish's vortex street compared to its pre-training performance. Similarly, Verma et al.[3] and Gazzola et al.[23, 24] also obtained the similar conclusion in simulation environment.

In addition to training the robotic fish in a virtual environment, direct training in the real world can work without prior knowledge of hydrodynamics. Fu et al.[25] placed two robotic fish in a low-turbulence water tunnel following a leader-follower formation. By employing RL method, the energy consumption of the following fish was used as the reward signal, and the parameters of the CPG equations of the following fish were constantly changed to optimize the swimming posture. Finally, an optimal control strategy for saving energy was achieved.

4 Discussion

At present, reinforcement learning has demonstrated favorable results across various tasks, showcasing its promising potential as a novel control method for this domain, but there are limitations that necessitate further exploration:

1) The application of reinforcement learning in the field of robotic fish control is still in the exploratory stage, and has not formed a mature and stable mode.
2) Reinforcement learning requires a large amount of data, thereby entailing extensive interaction data with the environment during robotic fish training, which is costly in real-world environments. Consequently, constructing simulation environments or enhancing data collection efficiency, learning efficacy, and training success rates in the real environment are potential solutions which need to be explored.
3) The utilization of a simulation environment can enhance the learning efficiency of robotic fish and mitigate risks during the training process. However, due to unpredictable differences between the simulation environments and the real environments, numerous challenges arise when transferring the controllers trained in the simulation to the real-world scenarios. Therefore,

it is imperative to explore effective sim to real transfer method.

4) The design and control effectiveness of reinforcement learning methods for robotic fish are directly influenced by the degrees of freedom in their structure and motion. It is of great significance to utilize different structural configurations of robotic fish to apply reinforcement learning control. Currently, most robotic fish employing reinforcement learning methods are rigid body structures driven by servos, with limited research on actuators other than servos. Nowadays, there exist a variety of actuators for robotic fish with rigid body structures. Different actuators lead to the control strategies of robotic fish different, and also impact the design and control strategies of the corresponding reinforcement learning. Moreover, while electrical rigid actuators exhibit good controllability, there are few degrees of freedom in motion and fall short in replicating motion performance of bionic objects. Only a few studies involved soft structures. The rigid and flexible composite fish body structure is closer to real fish, and its motion performance is closer to the bionic object; however, it possesses significantly different dynamic laws compared to the rigid structure. As an adaptive strategy, reinforcement learning methods can complete tasks without requiring complex dynamics modeling for soft robots. Nevertheless, only a handful of studies have explored the application platform of reinforcement learning control using soft structures as well; hence further exploration into diverse structural designs and materials for robotic fish is warranted.
5) An excellent and easily usable simulation software can help more researchers to deploy RL methods in their robots, like $Mojoco$ and $IssacGym$ in robotic arm with RL methods. But in robotic fish field, no enough accurate and widely used software for researcher to train his or her agent. Therefore, it's helpful to develop software combined RL training process and environment. To better simulate the uncertainty of real water, some special RL methods can be used as references, such as Domain Randomization.

5 Conclusion

Reinforcement learning methods have garnered significant attention in recent years, and offer a potential solution for controlling robotic fish with unclear dynamic models. In this article, we present the survey that focuses on design details of RL methods in robotic fish field, and describe existing promising achievements. However, multiple challenges remain in this field. To the best of our knowledge, we have discussed these challenges and possible solutions.

While nature fish may not comprehend the physical laws of hydrodynamics, this does not hinder fish from their mastery of swimming in the sea. Through continuous interaction, perception and learning processes within water environments, fish develop extraordinary swimming abilities. By emulating the biological "body" through mechanical structure and employing reinforcement learning method to acquire behavioral logic via environment interactions, robotic fish hold promising prospects for attaining biological "intelligence".

References

[1] S. Youssef, M. Soliman, M. Saleh et al,, Design and control of soft biomimetic pangasius fish robot using fin ray effect and reinforcement learning, *Scientific Reports*, 12(1):21861, 2022.

[2] S. Yan, Z. Wu, J. Wang, Y. Huang, M. Tan, and J. Yu, Real-World Learning Control for Autonomous Exploration of a Biomimetic Robotic Shark, *IEEE Trans. on Industrial Electronics*, 70(4): 3966-3974, 2023.

[3] S. Verma, G. Novati, and P. Koumoutsakos, Efficient collective swimming by harnessing vortices through deep reinforcement learning, *proceedings of the national academy of sciences of the united states of America*, 115(23): 5849-5854, 2018.

[4] Y. Zhu, J. Pang, T. Gao and F. Tian, Learning to school in dense configurations with multi-agent deep reinforcement learning, *Bioinspiration & Biomimetics*, 18(1): 2022.

[5] Y. Zhu and J. Pang, A numerical simulation of target-directed swimming for a three-link bionic fish with deep reinforcement learning, in *Proceedings of the Institution of Mechanical Engineers Part C-Journal of Mechanical Engineering Science*, 2023: 2450-2460.

[6] Y. Zhu, J. Pang, and F. Tian, Point-to-Point Navigation of a Fish-Like Swimmer in a Vortical Flow With Deep Reinforcement Learning, *Frontiers in Physics*, 10: 2022.

[7] Y. Zhu, F. Tian, J. Young, J. Liao, and J. Lai, A numerical study of fish adaption behaviors in complex environments with a deep reinforcement learning and immersed boundary-lattice Boltzmann method, *Scientific Reports*, 11(1):1691, 2021.

[8] L. Li, D. Liu, J. Deng, M. Lutz, and G. Xie, Fish can save energy via proprioceptive sensing, *Bioinspiration & Biomimetics*, 16(5): 2021.

[9] J. Zheng, T. Zhang, C. Wang, M. Xiong, and G. Xie, Learning for Attitude Holding of a Robotic Fish: An End-to-End Approach With Sim-to-Real Transfer, *IEEE Trans. Robot*, 38(2): 1287-1303, 2022.

[10] T. Zhang et al., From Simulation to Reality: A Learning Framework for Fish-Like Robots to Perform Control Tasks, *IEEE Trans. Robot*, 38(6): 3861-3878, 2022.

[11] T. Zhang et al., Leveraging Imitation Learning on Pose Regulation Problem of a Robotic Fish, *IEEE Trans. on Neural Networks and Learning Systems*, 2022.

[12] T. Zhang et al., Decentralized Circle Formation Control for Fish-like Robots in the Real-world via Reinforcement Learning, in *IEEE International Conference on Robotics and Automation*, 2021: 8814-8820.

[13] T. Zhang, R. Tian, C. Wang, and G. Xie, Path-following Control of Fish-like Robots: A Deep Reinforcement Learning Approach, in *21st IFAC World Congress on Automatic Control - Meeting Societal Challenges*, 2020: 8163-8168.

[14] H. Dong, Z. Wu, Y. Meng, M. Tan, and J. Yu, Gliding Motion Optimization for a Biomimetic Gliding Robotic Fish, *IEEE-ASME Transactions on Mechatronics*, 27(3): 1629-1639, 2022.

[15] J. Liu, Z. Liu, Z. Wu, J. Yu, Three-Dimensional Path Following Control of an Underactuated Robotic Dolphin Using Deep Reinforcement Learning, in *IEEE International Conference on Real-time Computing and Robotics*, 2020: 315-320.

[16] J. Yu, C. Wang, and G. Xie, Coordination of Multiple Robotic Fish With Applications to Underwater Robot Competition, *IEEE Trans. on Industrial Electronics*, 63(2): 1280-1288, 2016.

[17] P. Duraisamy, M. Santhanakrishnan, and A. Rengarajan, Design of Deep Reinforcement Learning Controller Through Data-assisted Model for Robotic Fish Speed Tracking, *Journal of Bionic Engineer*, 20(3): 953-966, 2023.

[18] T. Costa, A. Laan, F. Heras, et al., Automated Discovery of

Local Rules for Desired Collective-Level Behavior Through Reinforcement Learning, *Frontiers in Physics*, 8, 2020.

[19] P. Chivkula, C. Rodwell, and P. Tallapragada, Curriculum-based reinforcement learning for path tracking in an underactuated nonholonomic system, in *2nd Modeling, Estimation and Control Conference*, 2022: 339-344.

[20] S. Rajendran, F. Zhang, Learning based speed control of soft robotic fish, in *11th Annual Dynamic Systems and Control Conference*, 2018.

[21] Y. Sun, C. Yan, X. Xiang, H. Zhou, D. Tang and Y. Zhu, Towards end-to-end formation control for robotic fish via deep reinforcement learning with non-expert imitation, *Ocean Engineering*, 271: 2023.

[22] G. Novati, S. Verma, D. Alexeev, D. Rossinelli et al., Synchronisation through learning for two self-propelled swimmers, *Bioinspiration & Biomimetics*, 12(3): 2017.

[23] M. Gazzola, B. Hejazialhosseini, and P. Koumoutsakos, reinforcement learning and wavelet adapted vortex methods for simulations of self-propelled swimmers, *Siam Journal on Scientific Computing*, 36(3): B622-B639, 2014.

[24] M. Gazzola, A. Tchieu, D. Alexeev, A. de Brauer, and P. Koumoutsakos, Learning to school in the presence of hydro dynamic interactions, *Journal of Fluid Mechanics*, 789: 726-749, 2016.

[25] R. Fu, L. Li, C. Xu, and G. Xie, Studies on Energy Saving of Robot Fish Based on Reinforcement Learning, *Acta Scientiarum Naturalium Universitatis Pekinensis*, 55(3): 405-410, 2019.

[26] J. Liu, H. Hu, D. Gu, A hybrid control architecture for autonomous robotic fish, in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2006: 312-+.

[27] J. Liu, H. Hu, D. Gu, RL-based optimisation of robotic fish behaviours, in *6th World Congress on Intelligent Control and Automation*, 2006:3992-+.

[28] Z. Shen, Z. Cao, M. Tan, and S. Wang, Control of obstacle avoidance of biomimetic robot fish based on reinforcement learning, *High Technology Letters*, 16(12): 1253-1258, 2006.

[29] Z. Shen, M. Tan, Z. Cao, S. Wang, and Z. Hou, Obstacle avoidance learning for biomimetic robot fish, in *7th International Conference on Fuzzy Logic and Intelligent Technologies in Nuclear Science*, 2006: 719-+.

[30] J. Shao, L. Wang, Cooperation of multiple fish-like microrobots based on reinforcement learning, in *1st IEEE Symposium on Artificial Life*, 2006: 348-+.

[31] L. Lin, H. Xie, D. Zhang, and L. Shen, Supervised Neural Q_learning based Motion Control for Bionic Underwater Robots, *Journal of Bionic Engineer*, 7: S177-S184, 2010.

[32] J. Wang, J. Kim, Optimization of Fish-like Locomotion using Hierarchical Reinforcement Learning, in *12th International Conference on Ubiquitous Robots and Ambient Intelligence*, 2015: 465-469.

[33] K. Gustavsson, L. Biferale, A. Celani, and S. Colabrese, Finding efficient swimming strategies in a three-dimensional chaotic flow by reinforcement learning, *Eur Phys J E Soft Matter*, 40(12): 2017.

[34] J. Liu, Z. Liu, Z. Wu, J. Yu, Three-Dimensional Path Following Control of an Underactuated Robotic Dolphin Using Deep Reinforcement Learning, in *IEEE International Conference on Real-time Computing and Robotics*, 2020: 315-320.

[35] H. Yang, W. Huang, and F. Ao, Simulation on self-organization behaviors of fish school based on reinforcement learning, *Journal of National University of Defense Technology*, 42(1): 194-202, 2020.

[36] H. Deng, P. Burke, D. Li, B. Cheng, Design and Experimental Learning of Swimming Gaits for a Magnetic, Modular, Undulatory Robot, in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2021: 9562-9568.

[37] Y. Jiao, F. Ling, S. Heydari, N. Heess, J. Merel, and E. Kanso, Learning to swim in potential flow, *Physical Review Fluids*, 6(5): 2021.

[38] L. Yan, X. Chang, N. Wang, R. Tian, L. Zhang, and W. Liu, Learning how to avoid obstacles: A numerical investigation for maneuvering of self-propelled fish based on deep reinforcement learning, *International Journal for Numerical Methods in Fluids*, 93(10): 3073-3091, 2021.

[39] S. Yan, Z. Wu, J. Wang, M. Tan, and J. Yu, Efficient Cooperative Structured Control for a Multijoint Biomimetic Robotic Fish, *IEEE-ASME Trans. on Mechatronics*, 26(5): 2506-2516, 2021.

[40] H. Yu, B. Liu, C. Wang, X. Liu, X. Lu, and H. Huang, Deep-reinforcement-learning-based self-organization of freely undulatory swimmers, *Physical Review E*, 105(4): 2022.

[41] N. Van Dong, V. Dinh Quoc, D. Van Tu, N. Huy Hung, and N. Tan Tien, Reinforcement learning-based optimization of locomotion controller using multiple coupled CPG oscillators for elongated undulating fin propulsion, *Mathematical Biosciences and Engineering*, 19(1): 738-758, 2022.

[42] L. Yan, X. Chang, R. Tian, N. Wang, L. Zhang, and W. Liu, A numerical simulation method for bionic fish self-propelled swimming under control based on deep reinforcement learning, in: *Proceedings of the Institution of Mechanical Engineers Part C-Journal of Mechanical Engineering Science*, 2020: 3397-3415.

[43] J. Yu, Z. Wu, X. Yang, Y. Yang, and P. Zhang, Underwater Target Tracking Control of an Untethered Robotic Fish With a Camera Stabilizer, *IEEE Tran. on Systems Man Cybernetics-Systems*, 51(10): 6523-6534, 2021.

[44] Y. Zhu, J. Pang, and F. Tian, Stable Schooling Formations Emerge from the Combined Effect of the Active Control and Passive Self-Organization, *Fluids*, 7(1): 2022.

[45] J. Hwangbo et al, Learning agile and dynamic motor skills for legged robots, *Science Robot*, 4(26): 2019.

[46] G. Sartoretti, W. Paivine, Y. Shi, Y. Wu, and H. Choset, Distributed Learning of Decentralized Control Policies for Articulated Mobile Robots, *IEEE Trans. Robot*, 35(5): 1109-1122, 2019.

[47] P. Duraisamy, R. Sidharthan, and M. Santhanakrishnan, Design, Modeling, and Control of Biomimetic Fish Robot: A Review, *Journal of Bionic Engineering*, 16(6): 967-993, 2019.

[48] H. Nguyen and H. La, Review of Deep Reinforcement Learning for Robot Manipulation, in *2019 Third IEEE International Conference on Robotic Computing*, 2019: 590-595.

[49] F. Agostinelli, G. Hocquet, S. Singh, and P. Baldi, From Reinforcement Learning to Deep Reinforcement Learning: An Overview, *Lect Notes Artif Int*, 11100: 298-328, 2018.

[50] R. Martin-Martin, M. Lee, R. Gardner, S. Savarese, J. Bohg, and A. Garg, Variable Impedance Control in End-Effector Space: An Action Space for Reinforcement Learning in Contact-Rich Tasks, in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2019: 1010-1017.

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

[52] V. Mnih, et al., Human-level control through deep reinforcement learning, *Nature*, 518: 529-533, 2015.

[53] V. Mnih, et al., Asynchronous methods for deep reinforcement learning, in *International Conference on Machine Learning*, 2016: 1928-1937.

[54] T. Lillicrap et al., Continuous control with deep reinforcement learning, in *Proc.Int.Conf.Learn.Representations*, 2016.

[55] R. Sutton and A. Barto, Reinforcement Learning: An Introduction. Cambridge: MIT-Press, 2018.