2026 IEEE International Conference on Robotics and Automation (ICRA 2026)
June 1-5, 2026. Vienna, Austria

Agile and Controllable Omnidirectional Fast-start Maneuvers of Robotic Fish via Bio-inspired Reinforcement Learning

Xu Huang*, Xiaozhu Lin*, *Graduate Student Member, IEEE*, Xiaopei Liu,
and Yang Wang†, *Member, IEEE*

***Abstract*— Fast-start maneuvers—exemplified by the C-start in fish—represent a highly agile and very attractive locomotor strategy that requires precise multi-joint coordination under conditions of unsteady fluid dynamics, and has evolved through extensive predator–prey interactions in natural environments. Replicating such maneuvers in robotic fish is challenging due to strong fluid–structure nonlinearities, instantaneous dynamics, and complex vortex interactions. Prior approaches were limited by their dependence on specialized materials, lack of active controllability, incompatibility with mechanical structures, and inability to generate sufficient forward propulsion. Here, we propose a deep reinforcement learning method for multi-joint robotic fish that embeds key biological features of C-start maneuvers—burst acceleration, rapid directional adjustment, and two-stage bend-and-stretch motion—into the reward and observation design. By training in a physically consistent, high-performance Computational Fluid Dynamics (CFD) solver, the agent autonomously discovers effective launch strategies without requiring explicit models or real fish data. The resulting policies not only reproduce C-start-like motions and achieve fully controllable directional fast-starts, but also significantly expand the maneuvering potential of robotic fish, enabling higher velocities, greater displacement, and more agile motion than state-of-the-art methods. This biologically inspired and generalizable method demonstrates the promise of integrating biological principles into reinforcement learning to unlock advanced, high-acceleration capabilities in multi-joint aquatic robots.**

I. Introduction

Robotic fish have attracted considerable attention in recent years due to their unique advantages, such as high propulsion efficiency, environmental adaptability, and stealth capability [1]. In particular, robotic fish propelled by bio-inspired bodies and/or caudal fins (BCF) have seen significant progress over the past decade, achieving notable advancements in electromechanical design [2], [3], underwater perception [4], [5], and locomotion control [6], [7]. Nevertheless, there remains a striking gap between robotic fish and natural fish in terms of agility in swimming. A typical example is "fast-starts"—brief and rapid acceleration maneuvers employed by fish during predator-prey interactions, whose performance is crucial to the success of fish predation and their survival [8], [9]. Harper and Blake [10] reported that the peak velocity of northern pike during fast-starts can reach 6.25 m/s. However, at present, the best fast-start performance achievable by robotic fish of the same type—through researching and mimicking the characteristic of C-starts (where a fish's body rapidly bends into "C" shape to achieve high acceleration and rapid turning)—only achieves a peak velocity of about 1.5 m/s [11]. This performance not only pales in comparison to natural fish, but is also achieved by mechanical devices that passively release stored energy. Such an approach is fundamentally limited, as it is uncontrollable, relies on special materials, and suffers from other constraints that hinder broad applicability. Traditional perspectives attribute this gap primarily to differences between robotic fish and real fish in aspects such as materials, morphology, actuation, and number of joints, while often overlooking the gap in locomotion strategies. In this paper, from the perspective of control strategies, we conduct a study on how to further enhance the fast-start capability of existing classical BCF-propelled robotic fish configuration by leveraging reinforcement learning and a physically-consistent high-performance CFD simulation environment [12], which is expected to not only significantly improve maneuverability and responsiveness of robotic fish in high-agility underwater tasks, but also exemplifies the significance of control in biomimetic robotics research.

The study of fast-start maneuvers in robotic fish is inherently a challenging nonlinear problem, in which the strong nonlinearities primarily arise from fluid–structure interactions and multi-joint coordination, as well as complex unsteady fluid dynamics dominated by transient vortices and wake effects [14]. Precisely because of this, there are relatively few research outcomes that have properly addressed these challenges, with typical examples including: A bio-inspired robotic fish using spine snap-through buckling to achieve fast-start maneuvers and high accelerations [11]. However, as mentioned before, it was implemented with specialized elastic materials, is uncontrollable, has extremely limited maneuverability, and is not compatible with conventional robotic fish designs. To some extent, directional controllability during C-start maneuvers was achieved in a four-joint robotic fish via closed-loop control [14], yielding large turning angles and high turning rates. However, this approach failed to effectively convert the substantial angular velocity into forward speed during the maneuver partly due to the lack of systematic optimization. As a result, the motion of the robotic fish generated by this approach more closely resembles spinning in place rather than mimicking the biological fast start, which enables real fish to rapidly

* denote Equal contribution, † denote Corresponding author.

This work was supported by the National Natural Science Foundation of China under Grant 62503329.

Xu Huang, Xiaozhu Lin, Xiaopei Liu and Yang Wang are with the School of Information Science and Technology, ShanghaiTech University, Shanghai, China {huangxu2024, linxzh, liuxp, wangyang4}@shanghaitech.edu.cn

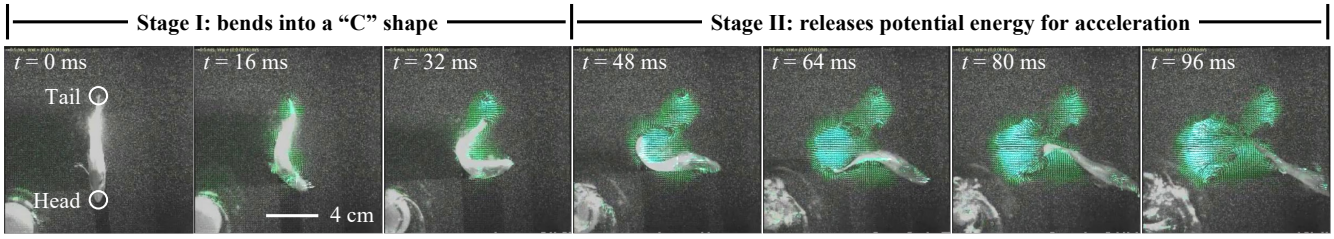


Fig. 1. **C-start maneuver of a typical bluegill sunfish showing the hydrodynamic flows at the mid-body level (green velocity vectors).** Time-resolved digital particle image velocimetry (DPIV) at 1000 fps was used to capture the flow patterns from a ventral view over 100 ms. Timestamps indicate the elapsed time from the onset of the maneuver, and the scale bar denotes the body length (BL). Head and tail positions are highlighted by circles for clarity. The original image of the natural sunfish was processed from the supplementary video provided by Tytell et al. [13]

propel themselves in a desired direction. It is worth noting that fast-start maneuvers in fish have been a popular topic in biology. For instance, Mandralis et al. [15] applied reinforcement learning to identify energy-efficient escape patterns in larval fish, revealing C-start and burst-and-glide motions that maximize escape distance under energy constraints. While these studies are highly inspiring, they are unfortunately not directly applicable to robotic fish. This is primarily because they generally rely on continuum-based models, which are incompatible with discrete actuators employed in robotic systems. Moreover, such studies often overlook mechanical constraints, actuation limits, and real-world hydrodynamic disturbances. Taken together, these studies reveal the key challenges in reproducing fast-start maneuvers in robotic fish. Existing approaches are limited by their reliance on specialized or hand-made materials, insufficient active controllability, poor robustness, and inability to achieve high forward speeds.

In this work, we propose a Deep Reinforcement Learning (DRL)-based control method for multi-joint BCF robotic fish to achieve agile and controllable omnidirectional fast-start maneuvers. Our approach explicitly leverages biological insights: key motion features from real fish fast-starts—such as rapid burst acceleration [10], instantaneous directional adjustment [16], and two-stage bend-and-stretch motions [17]—are embedded into the reward and observation design. By structuring the learning objective around these biologically inspired cues, the agent is guided to discover effective turning and propulsion strategies without manual trajectory design. DRL then efficiently handles the high-dimensional continuous state and action spaces [7], [18], training the agent in a physically consistent high-performance CFD solver [12] that provides realistic fluid–structure interaction data. As a result, the learned policies reproduce C-start-like behaviors, achieve fully directional controllable fast-starts, and generate motion patterns closely matching those of biological fish. Compared with state-of-the-art methods [14], our approach yields lower directional errors, higher maximum forward velocities, and larger displacement projections, demonstrating that embedding biological principles into reinforcement learning can effectively produce agile, high-acceleration maneuvers with minimal manual design effort.

Overall, the distinctive features of this work are threefold:

1) Biologically Inspired: The method incorporates key features of fish fast-starts into reward and observation design, allowing robotic fish to autonomously learn efficient fast-start maneuvers.
2) Simple Implementation: The method does not require explicit dynamic models or real fish data, making it straightforward to apply to multi-joint robotic fish.
3) Agile and Omnidirectional: The resulting control strategies enable high-velocity, omnidirectional maneuvers in multi-joint robotic fish, with motion patterns broadly resembling fast-start behaviors.

II. Methodology

In this section, we propose a biologically inspired reinforcement learning method for generating agile, omnidirectional fast-start maneuvers in multi-joint robotic fish. The method incorporates key motion features from real fish C-starts into the reward function and trains the agent within a physically consistent, high-performance CFD simulator that captures fluid–body interactions.

A. Bio-inspired DRL Framework

We formulate the fast-start task of robotic fish as a Markov Decision Process (MDP) defined by (S,A,r,D,P), where S is the state space, A the action space, r the reward function, D the initial state distribution, and P the state transition probability. At each step, the agent selects an action a_t according to its policy π, receives a reward r_t, and transitions to a new state s_{t+1}. The goal is to learn an optimal policy π^* that maximizes the expected cumulative reward: $\pi^* = \arg\max_\pi \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r_t\right]$, where $\gamma \in [0,1)$ is the discount factor and T is the episode length.

1) Bio-inspired Reward Shaping: Weihs [19] described fast-start maneuvers in fish as three stages: Stage 1, a unilateral bend forming a C-shape (preparatory); Stage 2, a contralateral tail flip generating thrust (propulsive); and Stage 3, a variable braking or coasting phase [20]. Figure 1 illustrates a typical C-start maneuver of a bluegill sunfish, showing the body configuration over time from a ventral view. Stage 1 corresponds to the initial turning of the fish's head and body toward the escape direction, while Stage 2 corresponds to the rapid tail flip generating forward acceleration. Image processed from video provided by Tytell et al. [13].

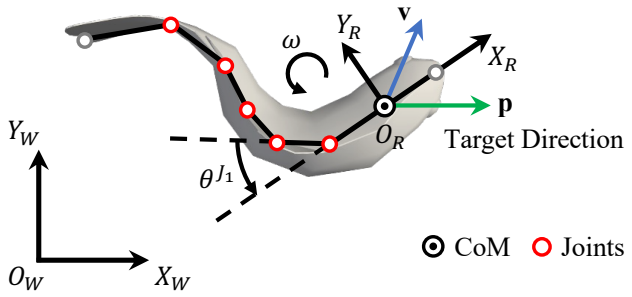


Fig. 2. **Schematic of robotic fish fast-start states and coordinate frames.** Illustration of the robotic fish's joint positions, center of mass, joint-link structure, target direction, and the coordinate frames used to represent these quantities in the simulation environment.

To incorporate these biological objectives into reinforcement learning, we simplify the C-start into two stages: Stage I (turning) and Stage II (acceleration). The reward function follows this sequence: Stage I encourages rapid reduction of directional error for turning, while Stage II promotes strong acceleration along the target vector. This bio-inspired transition from reorientation to acceleration, defined using the coordinate frames and CoM in Fig. 2, is directly mapped to stage-specific learning signals.

a) Stage I (Turning): To drive rapid reorientation, the Stage I reward encourages reduction of the direction error:

$$r_t^{\mathrm{I}} = w_1\big(|\phi_{t-1}| - |\phi_t|\big), \tag{1}$$

where ϕ_t is the angular difference between the head orientation and the escape direction at time t, and w_1 controls the strength of directional alignment.

The agent transitions to Stage II when ϕ_t changes sign ($\phi_t \cdot \phi_{t-1} < 0$), indicating that the heading has crossed the target direction, as in biological C-starts.

b) Stage II (Acceleration): Once aligned, the reward shifts focus to propulsion:

$$r_t^{\mathrm{II}} = w_2\,(\mathbf{v}\cdot\mathbf{p}), \tag{2}$$

where $\mathbf{v}$ is the fish velocity, $\mathbf{p}$ is the unit vector along the escape direction, and w_2 scales the propulsion reward.

c) Joint Velocity Penalty: To ensure biologically plausible smoothness, a penalty term discourages excessive joint velocities:

$$r_t^{\text{penalty}} = -w_0\|\dot{\mathbf{J}}_t\|^2, \tag{3}$$

where $\dot{\mathbf{J}}_t$ are joint angular velocities and w_0 regulates the penalty strength.

d) Total Episode Reward: The episode reward integrates stage-specific rewards, the smoothness penalty, and a terminal displacement term:

$$R = \sum_{t=0}^{T}\left(\begin{cases} r_t^{\mathrm{I}}, & \text{Stage I,} \\ r_t^{\mathrm{II}}, & \text{Stage II} \end{cases} + r_t^{\text{penalty}}\right) + w_3\big((\mathbf{X}_T - \mathbf{X}_0)\cdot\mathbf{p}\big). \tag{4}$$

where $\mathbf{X} = [x, y] \in \mathbb{R}^2$ is the position of the robot in the global frame and w_3 regulates the corresponding coefficient.

This reward design grounds RL in biological insights by decoupling the fast-start maneuver into two functional stages. The weights $[w_0, w_1, w_2, w_3] = [0.005, 5, 10, 10]$ were empirically tuned to balance agility, accuracy, and mechanical stability. Specifically, w_1 and w_2 govern the trade-off between reorientation speed and terminal propulsion: while a higher w_1 facilitates rapid heading correction, it may cause directional overshooting; conversely, w_2 prioritizes the subsequent burst acceleration. The penalty w_0 is critical for suppressing high-frequency oscillations, ensuring biologically plausible motions that minimize mechanical stress. Because the Stage I reward is capped and the Stage II reward scales with velocity, the agent is naturally incentivized to complete the reorientation quickly and maximize forward thrust. This structure effectively yields agile, omnidirectional maneuvers that combine precise directional control with high propulsion efficiency.

TABLE I
HYPERPARAMETERS OF SOFT ACTOR-CRITIC (SAC)

Parameters	Value
discount factor (γ)	0.99
replay buffer size	10^6
batch size	256
learning rate	3×10^{-4}
target smoothing coefficient (τ)	0.005
learning step	1
entropy target	-3

2) State Space: At each time step t, the RL agent observes the environment state s_t, which consists of self-perception and task-relevant variables, defined as

$$S_t = [\mathbf{p}, \omega, \mathbf{v}, \mathbf{J}, \dot{\mathbf{J}}, \tau_t, \sigma_t], \tag{5}$$

where $\mathbf{p} = [p_x, p_y] \in \mathbb{R}^2$ denotes the desired escape direction in the robot's body frame, $\omega \in \mathbb{R}$ and $\mathbf{v} = [v_x, v_y] \in \mathbb{R}^2$ represent the angular and linear velocities of the robot, $\mathbf{J} = [\theta_1, \theta_2, \theta_3, \theta_4, \theta_5] \in \mathbb{R}^5$ and $\dot{\mathbf{J}} = [\dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3, \dot{\theta}_4, \dot{\theta}_5] \in \mathbb{R}^5$ are the joint angles and joint angular velocities, $\tau_t \in [0, 1]$ is the normalized time progress within the current stage, and $\sigma_t \in \{0, 1\}$ is the stage switch indicator, where $\sigma_t = 0$ for Stage I and $\sigma_t = 1$ for Stage II.

3) Action Space: At each time step t, the neural network controller observes the current state s_t and outputs an action $a_t = [\mathbf{J}_{\text{des}}] \in \mathbb{R}^5$, where J_{des}^i denotes the desired angle of joint i. These commands are applied directly to the robot joints, without relying on lower-level mechanisms such as central pattern generators (CPGs) or predefined motion trajectories which allows the policy to autonomously learn non-periodic, transient fast-start maneuvers [6].

The action space design offers two key advantages. First, providing full freedom in joint movements enables the agent to discover efficient, non-periodic strategies that exploit complex fluid-dynamic interactions. Second, this end-to-end joint control eliminates the need for manually designed low-level controllers or CPGs, simplifying the architecture and allowing the policy to adapt instantaneously to varied escape demands.

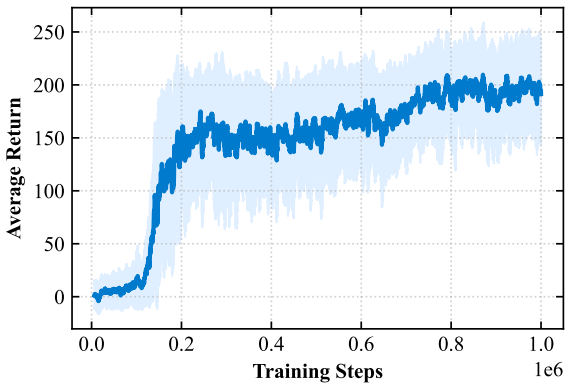


Fig. 3. **Training reward curves.** The average return (total reward) was smoothed using a sliding window of 50 episodes. The solid line denotes the mean return, and the shaded region indicates one standard deviation around the mean.

TABLE II
SIMULATION KEY PARAMETERS

Parameters	Value
CFD domain size (L×W×H, m)	3.6 × 3.6 × 0.5
Simulation timestep (s)	0.004
Control timestep (s)	0.02
Fish robot body length (m)	0.32
Fish robot mass (kg)	1.1
Joint range(degree)	±60
Joint velocity range(degree/s)	±300

B. CFD Simulator

RL requires extensive interactions between the agent and the environment to learn complex policies from scratch. Therefore, a fast and accurate simulation platform is essential. A major challenge for robotic fish lies in the complex dynamics of the surrounding fluid. To address this, we employ a novel three-dimensional fluid–structure interaction simulator [12][1], which builds upon an efficient lattice Boltzmann solver and incorporates a dynamically moving local domain that tracks the agent. The fluid dynamics are modeled by the following unsteady, isothermal, weakly compressible Navier–Stokes (NS) equations:

$$\begin{aligned} &\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0, \\ &\frac{\partial (\rho \mathbf{u})}{\partial t} + \nabla \cdot (\rho \mathbf{u}\mathbf{u}) = -\nabla p + \nabla \cdot \sigma + \mathbf{F}, \\ &p = \rho R T_0, \end{aligned} \tag{6}$$

where $\mathbf{u}$, ρ, and p denote the fluid velocity, density, and pressure fields, respectively; $\mathbf{F} = \rho a$ is the external force field, with a representing its corresponding acceleration. T_0 is the constant environmental temperature, R is the specific gas constant, t denotes time, and σ is the shear stress tensor, defined by the linear constitutive law. This design provides physically consistent, high-performance simulation of hydrodynamic interactions experienced by robotic fish. Using this simulator, 1 second of physical time can be reproduced in approximately 2 seconds on a standard personal computer, enabling efficient RL training for agile, omnidirectional fast-start maneuvers. Moreover, the simulator has demonstrated successful transfer of learned policies to real robotic fish [6], supporting real-world deployment. For further details, see [21].

[1]Please refer to `https://collisionmodel.com` for more details and updates (accessed Mar. 6, 2026).

C. Training Details and Result

We employ the Soft Actor-Critic (SAC) algorithm to train fast-start policies for the robotic fish. SAC adds an entropy regularization term, controlled by temperature α, to balance exploration and exploitation, improving sample efficiency—especially important in computationally intensive CFD simulations. The main hyperparameters are listed in Table I.

At each episode, the fish is placed at a fixed position with a random orientation, and the desired escape direction is sampled within $[-\pi, \pi]$ relative to its heading. Episodes terminate when the maximum step $T_{\max} = 50$ is reached, with a control interval of 0.02 s. The policy network (Actor) is an MLP $[17(\text{ReLU}), 256(\text{ReLU}), 256(\text{ReLU}), 5(\text{Tanh})]$, mapping states to five joint target angles. The Critic network (Q-network) is $[22(\text{ReLU}), 256(\text{ReLU}), 256(\text{ReLU}), 1]$, estimating action values. Fig. 3 shows the learning curve of the training process, illustrating that the cumulative reward increases steadily and converges after approximately 8×10^5 steps. This indicates that the SAC agent effectively learns stable and high-performance fast-start maneuvers within the proposed simulation environment. The entire training process(1×10^6 steps) was completed in approximately 4 hours on a workstation equipped with an Intel i9-12900K CPU and an NVIDIA GeForce RTX 4090GPU.

III. EXPERIMENTAL RESULTS

In this section, we present simulation experiments assessing the performance of our method in a fluid simulator [12] that captures key physical interactions. The evaluation includes a comprehensive comparison of our method and state-of-the-art (SOTA) controller proposed by Su et al. [14] (hereafter referred to as the baseline), as well as analysis of vortex structures in key action frames. First, we investigate the temporal dynamics of kinematic parameters during fast-start maneuvers from a quiescent state to evaluate the agent's capacity for rapid acceleration. Second, we statistically examine performance across multiple target directions to verify robustness and omnidirectional capability. Finally, the body motion snapshots of our method, the baseline method, and natural sunfish are also analyzed to reveal flow structures and bio-inspired mechanisms underlying the observed performance advantages.

A. Simulation Setup

The robotic fish model in the CFD environment consists of six rigid links connected by seven joints, with the central five joints actively actuated. The lengths and masses of the links are $L_1, \ldots, L_6$ and $M_1, \ldots, M_6$, respectively. The skeleton motion of the fish-like robot is driven by articulated

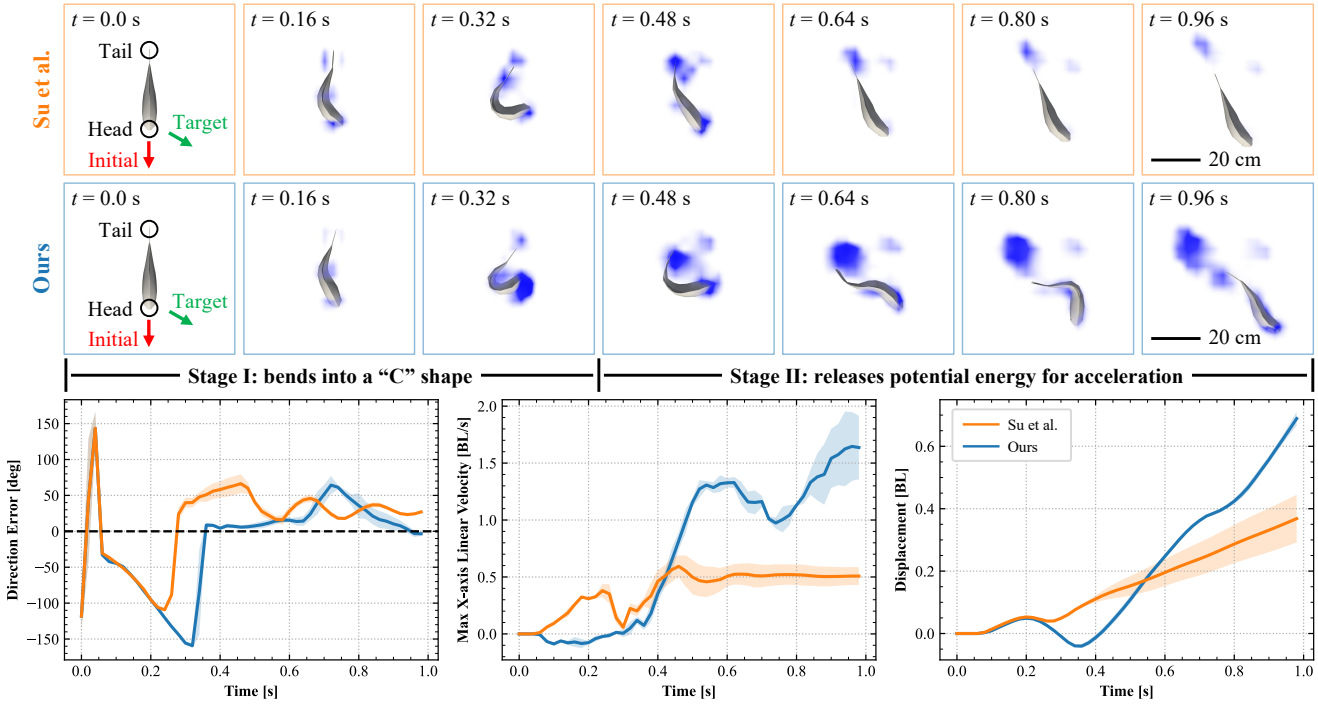


Fig. 4. **Process comparison of 60° target fast-start maneuvers between ours and baseline method.** Top: snapshot sequences of fast-start maneuvers, where the baseline is shown in yellow and our method in blue. Snapshots include the initial heading, head and tail markers, scale bar, and timestamps, with stage divisions indicated below. Bottom: temporal evolution of key parameters, where the forward speed is normalized by the body length (BL), where the solid line denotes the mean and the shaded area represents one standard deviation.

rigid body dynamics [22]. Given a set of joint angles, the surface shape is determined by linear blend skinning [23], allowing smooth mesh deformation and capturing fluid–structure interactions. All active joints are torque-controlled in the simulation. A first-order PD controller converts the desired joint positions J_{des} from the RL agent into torques:

$$\tau = k_p(J_{\text{des}} - J) + k_d(0 - \dot{J}), \tag{7}$$

where J and $\dot{J}$ are the current joint positions and velocities. This setup enables realistic actuation of the links and their interaction with the surrounding fluid. By capturing multi-joint dynamics and fluid–structure interactions in a physically consistent manner, the simulator serves as a reliable platform for validating fast-start maneuvers.

Table II summarizes the key simulation parameters settings used in this study. In the CFD simulation environment, the robot is initialized at the domain origin with zero rotation. The surrounding fluid is modeled with a density of 1000 kg/m³ and a kinematic viscosity of 1×10^{-6} m^2/s. These settings define the baseline physical properties of both the robot and the fluid in the simulation, as well as the initial configuration and actuation limits of the robot. Based on these settings, two types of experiments are designed.

B. Dynamics of Agile Fast-Start Maneuvers

In this experiment, the robotic fish starts from rest in a straight posture with all joint angles at zero and the head oriented along the positive x-axis (subject to a random deviation within $\pm 2°$). The target direction is set to 60° to the left of the initial heading. Each 1 s simulation is repeated five times to record the temporal evolution of the directional error, the body-fixed forward velocity, and the displacement along the target direction.

As illustrated in Fig. 4, snapshots at 0.16 s intervals reveal that our method generates a larger and more coherent flow region behind the fish compared to the baseline. This indicates more effective momentum transfer and higher reactive thrust. Conversely, the baseline produces smaller, more diffused flow structures, resulting in weaker propulsion and slower velocity growth. Quantitative analysis confirms these observations: while our method's directional error initially decreases slightly slower than the baseline, it quickly achieves a smaller steady-state error, demonstrating precise alignment. Furthermore, the forward velocity rises more rapidly and reaches a significantly higher peak (1.8 BL/s vs. 0.5 BL/s), leading to faster displacement along the target direction and a more agile escape maneuver. The initial jump in directional error is attributed to defining the velocity vector as the instantaneous heading. Although sensitive at low speeds, this metric provides a more faithful representation of the actual swimming direction than head orientation during large-scale body oscillations. Overall, these results highlight that our method successfully combines precise directional control with enhanced propulsion, enabling agile and robust fast-start performance.

C. Evaluation of Omnidirectional Fast-Start Maneuvers

The proposed method was evaluated across a wide range of target directions, from 15° to 165° in 15° increments, with five repeated trials conducted for each target direction.

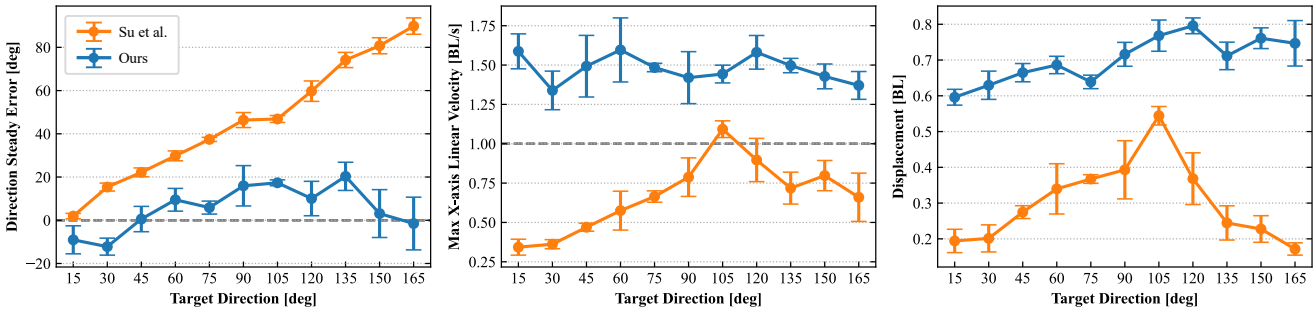


Fig. 5. **Statistical comparison of our method and the baseline across all directions.** Comparison of steady-state direction error, maximum forward velocity, and displacement projection across target directions from 15° to 165° in 15° increments, with baseline in yellow and our method in blue; each target direction was evaluated in 5 repeated trials, and error bars represent the standard error of the mean (SEM).

TABLE III

MEAN PERFORMANCE METRICS ACROSS TARGET DIRECTIONS (OURS / BASELINE)

Kinematic Variables	**Method**	**30°**	**45°**	**60°**	**90°**	**120°**	**135°**	**150°**
Max Angular Velocity [deg/s]	Su et al.	684.1±2.2	690.7±13.2	**699.1±12.5**	518.7±13.4	489.0±15.1	467.9±4.7	433.6±3.6
	Ours	**707.5±4.5**	**765.1±6.0**	691.0±8.2	**665.1±7.4**	**771.5±19.9**	**525.3±9.3**	**681.2±76.3**
Max Acc. along Goal [BL/s^2]	Su et al.	22.6±1.9	20.8±0.2	21.3±0.5	12.6±1.1	7.6±0.5	5.3±0.4	8.5±0.5
	Ours	**26.3±1.2**	**30.1±0.9**	**33.5±2.5**	**27.4±1.6**	**22.7±0.5**	**20.6±1.6**	**17.4±1.1**
Stage I Duration [s]	Su et al.	0.140	0.180	**0.200**	0.240	0.300	0.332	0.396
	Ours	**0.140**	**0.180**	0.224	**0.240**	**0.280**	**0.328**	**0.336**
Stage I Max Speed [BL/s]	Su et al.	0.5±0.0	0.7±0.0	0.7±0.0	0.7±0.1	0.4±0.0	0.3±0.0	0.3±0.0
	Ours	**1.4±0.1**	**1.5±0.0**	**1.7±0.1**	**1.6±0.1**	**1.5±0.1**	**1.6±0.1**	**1.3±0.1**

For every trial, we recorded the steady-state direction error, as well as the maximum forward velocity in the body-fixed frame.

As illustrated in Fig. 5 and Table III, our method consistently maintains directional errors within approximately 20° across all evaluated angles, demonstrating reliable control even during large-angle maneuvers. The RL agent achieves peak forward velocities of ≈ 1.5 BL/s, with initial bursts consistently exceeding 1 BL/s. This performance substantially surpasses the baseline, which exhibits lower startup speeds (≈ 0.7 BL/s) and higher variability across different headings. Quantitative analysis further substantiates these advantages: our method produces significantly higher maximum angular velocities and stronger accelerations, reaching 33.5 BL/s^2 at 60° (compared to 21.3 BL/s^2 for the baseline). Additionally, Stage I maximum speeds (1.3–1.7 BL/s vs. 0.3–0.7 BL/s) indicate more effective reactive propulsion. In summary, these results validate the agile, omnidirectional fast-start proficiency of the bio-inspired RL approach. While peak velocities remain below those of biological specimens due to simplified actuation and limited degrees of freedom, the method achieves precise directional alignment, high propulsive efficiency, and uniform escape performance. These attributes highlight its superiority in scenarios requiring omnidirectional maneuvering within simulated robotic constraints.

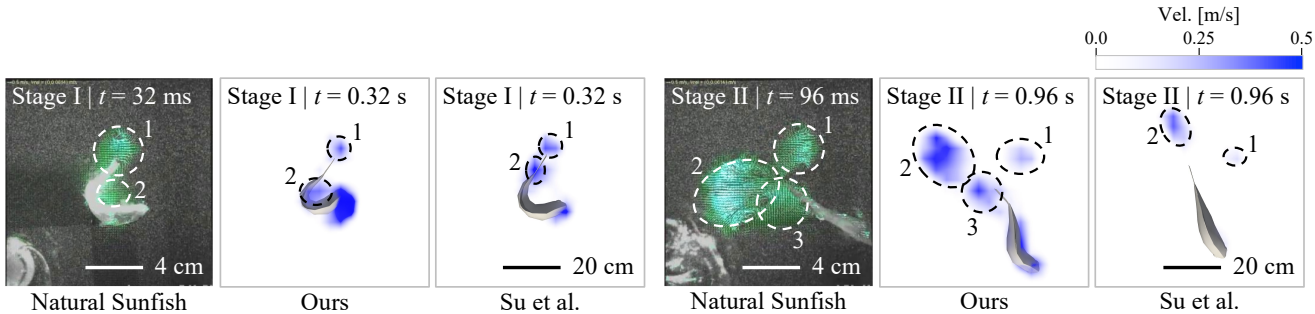
D. Vortex Structures during Fast-Start Maneuvers

To understand why our RL-based strategy achieves agile fast-start maneuvers, we analyze the flow structures generated in a representative case with a target direction 45°. As shown in Fig. 6, natural sunfish exhibit a characteristic vortex pattern: the turning stage generates two strong vortices near the mid-body and tail, followed by three well-formed tail vortices during acceleration, which reflect efficient momentum transfer. Our method is explicitly shaped to reproduce such dynamics and indeed closely matches the natural sequence: vortices emerge at nearly identical locations during the initial turn, and three prominent tail vortices, including one shed from the body side, form during propulsion. In contrast, the baseline controller departs from this pattern, producing misaligned vortices during turning and only two weaker tail vortices during acceleration, thereby missing the body-shed vortex.

The structural resemblance between our method's flow and that of natural sunfish underscores the efficacy of embedding biological principles within the reward design. Specifically, the three well-formed vortices characteristic of our strategy enhance backward jetting, thereby increasing reactive thrust. This leads to superior forward acceleration and greater displacement toward the target compared to the baseline, whose fragmented vortex structures yield diminished propulsion and reduced maneuverability. Ultimately, these results demonstrate that our bio-inspired approach successfully captures biologically consistent vortex dynamics, effectively translating natural swimming strategies into agile robotic fast-starts.

IV. CONCLUSIONS

In this work, we presented an RL-based method for generating fast-start maneuvers in a multi-joint robotic fish. By integrating a two-stage reward design inspired by biological



(a) Stage I (b) Stage II

Fig. 6. **Comparison of vortex structures generated during motion.** (a) Stage I and (b) Stage II, showing the tail fluid of a real fish (left), our method (middle), and the baseline method (right). Each panel includes timestamps, scale bars, and stage divisions. Colorbars indicate fluid velocity increasing from white (low) to blue (high), and dashed lines highlight prominent vortex structures. The original image of the natural sunfish was processed from the supplementary video provided by Tytell et al. [13]

behavior and training within a physically consistent, high-performance CFD simulator, the agent learned to rapidly turn toward the target direction and efficiently propel forward while maintaining smooth multi-joint coordination. Experiments in multiple target directions showed that the learned policy achieves superior directional accuracy, forward speed, and displacement compared to the baseline controllers. This study demonstrates that RL can effectively synthesize transient, non-periodic, and adaptive escape behaviors in articulated robotic fish, offering a promising approach for developing agile bio-inspired swimming robots. Future work includes transferring learned policies to real-world platforms, exploring the benefits of improving intelligent materials and structure design, and generalizing policies to more complex scenarios such as flow disturbances.

REFERENCES

[1] S. Yan, Z. Wu, J. Wang, Y. Feng, L. Yu, J. Yu, and M. Tan, "Recent advances in design, sensing, and autonomy of biomimetic robotic fish: A review," *IEEE/ASME Transactions on Mechatronics*, 2024.

[2] S. Yan, Z. Wu, J. Wang, S. Li, M. Tan, and J. Yu, "Towards unusual rolled swimming motion of a bioinspired robotic hammerhead shark under negative buoyancy," *IEEE/ASME Transactions on Mechatronics*, vol. 29, no. 3, pp. 2253–2265, 2023.

[3] Q. Cao, R. Wang, S. Huang, T. Zhang, B. Yin, M. Tan, and S. Wang, "Energy efficient swimming: Exploring an intermittent swimming gait for robotic fish via deep reinforcement learning," *IEEE/ASME Transactions on Mechatronics*, 2025.

[4] Y. Feng, Z. Wu, J. Wang, J. Gu, F. Yu, J. Yu, and M. Tan, "Decentralized multirobotic fish pursuit control with attraction-enhanced reinforcement learning," *IEEE Transactions on Industrial Electronics*, 2025.

[5] 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 Transactions on Industrial Electronics*, vol. 70, no. 4, pp. 3966–3974, 2022.

[6] X. Lin, X. Liu, and Y. Wang, "Learning agile swimming: An end-to-end approach without cpgs," *IEEE Robotics and Automation Letters*, 2025.

[7] T. Zhang, R. Tian, H. Yang, C. Wang, J. Sun, S. Zhang, and G. Xie, "From simulation to reality: A learning framework for fish-like robots to perform control tasks," *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3861–3878, 2022.

[8] P. Domenici and R. W. Blake, "The kinematics and performance of fish fast-start swimming," *Journal of Experimental Biology*, vol. 200, no. 8, pp. 1165–1178, 1997.

[9] E. D. Tytell and G. V. Lauder, "The c-start escape response of polypterus senegalus: bilateral muscle activity and variation during stage 1 and 2," *Journal of Experimental Biology*, vol. 205, no. 17, pp. 2591–2603, 2002.

[10] H. R. Frith, "Energetics of fast-starts in northern pike, esox lucius," Ph.D. dissertation, University of British Columbia, 1990.

[11] T. M. Currier, S. Lheron, and Y. Modarres-Sadeghi, "A bio-inspired robotic fish utilizes the snap-through buckling of its spine to generate accelerations of more than 20g," *Bioinspiration & Biomimetics*, vol. 15, no. 5, p. 055006, 2020.

[12] W. Song, H. Zhang, Y. Wang, and X. Liu, "Creating fluid-interactive virtual agents by an efficient simulator with local-domain control," *ACM Transactions on Graphics (TOG)*, vol. 44, no. 4, pp. 1–19, 2025.

[13] E. D. Tytell and G. V. Lauder, "Hydrodynamics of the escape response in bluegill sunfish, lepomis macrochirus," *Journal of Experimental Biology*, vol. 211, no. 21, pp. 3359–3369, 2008.

[14] Z. Su, J. Yu, M. Tan, and J. Zhang, "Implementing flexible and fast turning maneuvers of a multijoint robotic fish," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 1, pp. 329–338, 2013.

[15] I. Mandralis, P. Weber, G. Novati, and P. Koumoutsakos, "Learning swimming escape patterns for larval fish under energy constraints," *Physical Review Fluids*, vol. 6, no. 9, p. 093101, 2021.

[16] W. J. Stewart, A. Nair, H. Jiang, and M. J. McHenry, "Prey fish escape by sensing the bow wave of a predator," *Journal of Experimental Biology*, vol. 217, no. 24, pp. 4328–4336, 2014.

[17] J. Wakeling, "Biomechanics of fast-start swimming in fish," *Comparative Biochemistry and Physiology Part A: Molecular & Integrative Physiology*, vol. 131, no. 1, pp. 31–40, 2001.

[18] S. Yan, Z. Wu, J. Wang, M. Tan, and J. Yu, "Efficient cooperative structured control for a multijoint biomimetic robotic fish," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 5, pp. 2506–2516, 2020.

[19] D. Weihs, "The mechanism of rapid starting of slender fish," *Biorheology*, vol. 10, no. 3, pp. 343–350, 1973.

[20] B. Ahlborn, D. G. Harper, R. W. Blake, D. Ahlborn, and M. Cam, "Fish without footprints," *Journal of Theoretical Biology*, vol. 148, no. 4, pp. 521–533, 1991.

[21] X. Lin, W. Song, X. Liu, X. He, and Y. Wang, "Exploring learning-based control policy for fish-like robots in altered background flows," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 2338–2345.

[22] R. Weinstein, J. Teran, and R. Fedkiw, "Dynamic simulation of articulated rigid bodies with contact and collision," *IEEE Transactions on Visualization and Computer Graphics*, vol. 12, no. 3, pp. 365–374, 2006.

[23] N. Magnenat-Thalmann, R. Laperrière, and D. Thalmann, "Joint-dependent local deformations for hand animation and object grasping," in *Proceedings on Graphics interface'88*, 1989, pp. 26–33.