Navigation in three-dimensional vortical flows with egocentric observations based on imitation learning

*Zewei Xia, Chen Fang, Wenjie Hao, Zhi Wang, Chao Xu, Weiwei Che, Qi Gao, Shengze Cai**

Zewei Xia, Chen Fang, Wenjie Hao, Zhi Wang, Chao Xu, Shengze Cai
College of Control Science and Engineering, Zhejiang University, Hangzhou, China
Email Address: shengze_cai@zju.edu.cn
Weiwei Che
College of Information Science and Engineering, Northeastern University, Shenyang, China
Qi Gao
School of Aeronautics and Astronautics, Zhejiang University, Hangzhou, China

Keywords: *Autonomous navigation, Reinforcement learning, Imitation learning, Flow simulation, Swimmer*

Achieving high-precision autonomous navigation for robots in complex dynamic flow environment is a pivotal yet challenging task across broad applications, such as oceanographic surveys and aerial reconnaissance, where robots are required to resist intense vortex disturbances while maintaining stable locomotion. Although deep reinforcement learning (DRL) has been successfully applied in such task, existing methods either rely on global coordinates or body coordinates combined with sensor arrays (for measuring flow velocity gradients), neither of which is particularly practical. To address these limitations, a "teacher-student" DRL framework based on imitation learning is proposed. The teacher strategy is trained via the PPO algorithm using global geocentric observations, while the student imitates the teacher's actions relying solely on egocentric observations. By doing this, the student policy achieves a high success rate in navigation and exhibits strong generalization in unseen scenarios beyond the training environment, without external aids such as satellite positioning or flow gradient sensors. In addition, unlike the existing methods which are validated in a two-dimensional flow field, the proposed learning framework is experimented in complex three-dimensional flows for the first time. This work provides a viable practical solution for autonomous navigation of underwater robots, aerial vehicles and micro-swimmers in diverse real-world environments.

1 Introduction

Achieving high-precision navigation for autonomous robots in complex time-varying flow fields stands as a pivotal yet challenging task across broad applications, ranging from long-term ocean surveying carried out by underwater robots, or low-altitude reconnaissance and material delivery by unmanned aerial vehicles (UAVs) [1, 2, 3, 4]. In such scenarios, robots are required to determine appropriate paths through media perturbed by intense vortices generated from ocean currents or airflows, while sustaining stable locomotion in dynamic flow environments — much like their biological counterparts, such as aquatic swimmers and aerial flyers [5, 6, 7].

Conventional path planning approaches have been extensively investigated and deployed for robotic systems operating in time-varying complex flow fields. For instance, time-optimal control schemes have been applied to guide swimmers navigating in two dimensional turbulent flow environments [8, 9]. Nevertheless, trajectory planning based on optimal control theory typically necessitates a priori knowledge of the full flow field and its temporal dynamics [10, 11, 12, 13], which is practically unfeasible for real-time control of

underwater or aerial robots. While alternative control methodologies (e.g., adaptive control [14, 15] and model predictive control [16, 17]) are viable for motion planning and navigation in partially known flow fields, they still fall short of practical demands when flow information is limited [18]. Therefore, developing efficient, robust, and engineerable navigation strategies for marine and aerial robots bears considerable practical significance.

In recent years, deep reinforcement learning (DRL) has emerged as a promising alternative for this research topic. By continuously interacting with flow environments to learn control policies, DRL enables robots to achieve path planning with near-optimal performance in unknown or partially known flow fields [19, 20, 21, 22, 23]. For example, DRL has been successfully applied to glider soaring task, where it senses and exploits natural convection flows in both simulated scenarios and real-world applications [24, 25]. More specifically, in the context of navigation within turbulent or vortical flows, DRL has demonstrated remarkable potential in generating efficient path planning strategies that are comparable to those derived from traditional optimal control methods [26, 27, 28, 29, 30, 31, 32, 33]. These applications have verified that DRL-based swimmers or flyers can achieve effective navigation through unsteady flow fields relying solely on local measurements - a feature that renders such methods highly analogous to bio-inspired algorithms.

However, existing studies exhibit notable limitations that hinder their direct translation to engineering applications in real-world complex scenarios. In terms of environmental cues, while integrating deep reinforcement learning (DRL) with local flow velocity measurements has yielded successful navigation outcomes [26, 27], these works typically assume that robots or swimmers rely on inertial information—such as absolute position, velocity, and orientation defined in a geocentric coordinate system—to navigate in unsteady flow fields. Given that autonomous agents operate independently of external reference frames, more practical and on-board perceivable environmental cues are expected. More importantly, such geocentric environmental cues severely constrain the generalization capability of the trained DRL policies: the learned navigation strategies can only function effectively in the training flow scenarios and perform completely inadequately when transferred to unfamiliar environments [26], rendering them unsuitable for real-world deployment.

To mitigate these limitations, DRL integrated with egocentric observations in a body-centric coordinate system is recently investigated [34], a characteristic highly analogous to the ethological concept of "Umwelt" [35, 36]. It has been demonstrated that DRL strategy leveraging local and egocentric observations achieves successful navigation in unsteady flows and exhibits enhanced generalization performance in unfamiliar conditions beyond the training environment. Nevertheless, a key finding in [34] indicates that reliable egocentric navigation requires sensing not only local flow velocities but also local flow gradient information. We note, however, that measuring accurate local flow gradients necessitates the deployment of multiple distributed flow velocity sensors, which is not very practical for micro-robotic platforms (e.g., robotic fish or micro-UAVs). This is primarily due to the fact that flow gradients in real-world scenarios are typically weak and contaminated by measurement noise, rendering accurate detection challenging in a small scale. This then raises a fundamental question: *can effective egocentric navigation be achieved using DRL algorithms that rely solely on local flow velocity measurements?*

In this work, we aim to address this question by proposing a "teacher-student" DRL framework based on imitation learning. Following the work of [34], our objective is to de-

velop a navigation strategy for aquatic swimmers which can perceive target positions and flow motions in their egocentric body coordinate systems, eliminating the need for the global flow direction and inertial reference frame information. To this end, we first train a "teacher" strategy through the DRL-PPO algorithm, which leverages global geocentric observations to navigate efficiently in unsteady flow. Subsequently, in the identical flow environment, we train a "student" strategy that takes only egocentric observations as input, to imitate the action decisions of the teacher policy. Ultimately, the resulting "student" policy achieves high-precision navigation in complex flow fields without relying on global coordinate information (as used in [26]) and without additional observation data (e.g., flow gradients required in [34]). In addition, such a student policy can be transferred to unseen flow environments and achieve successful navigation without further training.

Notably, this work also represents the first attempt to investigate such DRL-based navigation tasks in three-dimensional (3D) flow fields. Prior DRL-based navigation methods have been exclusively validated in two-dimensional (2D) flow domains [26, 34, 27], which are only able to characterize planar velocity distributions and vortex evolution processes. These 2D models, however, fail to capture key 3D flow features such as the vertical oscillation of vortex cores [16, 37], rendering them inadequate for meeting the multi-degree-of-freedom motion requirements of underwater vehicles and low-altitude UAVs operating in real-world 3D spatial flow fields. This limitation is eliminated in this work, which paves a more practical path for autonomous navigation of underwater robots, aerial vehicles and micro-swimmers in diverse real-world environments.

2 Methodology

2.1 Problem Setup

The objective in this paper is to achieve efficient point-to-point navigation for a single autonomous agent in a 3D time-varying flow field. The agent starts from an initial position within the flow field, traverses an unsteady wake, and finally reaches the target position. Navigation is deemed successful if the agent arrives at the target position and is considered a failure if the agent moves out of the flow field domain or exceeds the time limit.

Here, we first establish the kinematic model of the agent. The velocity of the self-propelled agent is denoted as $\mathbf{U}_{swim}(\mathbf{x}, t) \in \mathcal{R}^3$, where $\mathbf{x} \in \mathcal{R}^3$ and t denote the space and time coordinates, respectively, which are omitted for the sake of brevity hereafter. In addition to representing the velocity vector $\mathbf{U}_{swim}$ by three velocity components, the velocity of the agent can also be represented by its translational speed magnitude U_{swim}, the pitch angle θ and the yaw angle ψ:

$$\mathbf{U}_{swim} = \begin{bmatrix} \cos\psi \cdot \cos\theta \\ \cos\psi \cdot \sin\theta \\ \sin\psi \end{bmatrix} \cdot U_{\text{swim}}. \tag{1}$$

In addition, although the agent's movement does not exert any effect on the flow field, the agent within the flow is subject to the local velocity of the flow field, denoted $\boldsymbol{v}_{\text{flow}} \in \mathcal{R}^3$. To this end, the kinematic of the agent can be simply expressed as follows:

$$\boldsymbol{v} = \mathbf{U}_{swim} + \boldsymbol{v}_{\text{flow}}. \tag{2}$$

Given the specified initial position, the velocity magnitude and the velocity direction of the agent, the position of the agent with discrete time steps is obtained by:

$$P_{t+1} = P_t + \boldsymbol{v}\Delta t, \tag{3}$$

where $t \geq 0$ and Δt is the time step set during the simulation process. Such a model is simple yet reasonable and is commonly adopted in studies of autonomous navigation in dynamic flow environments [26, 34].

More importantly, we note that the swimmer, modeled as a self-propelled agent, is constrained to a propulsion speed less than or equal to the maximum flow velocity in the simulation (i.e., $\mathbf{U}_{swim} \leq \max(\boldsymbol{v}_{\text{flow}})$). For example, in the case of 3D flow past a cylinder, it is assumed that the agent's velocity magnitude is not greater than the freestream (i.e., inlet) flow speed. Such a constraint makes the control and navigation task considerably challenging. Given that the agent's maximum speed is comparable to the freestream velocity, its propulsion capability is inadequate to counteract flow advection directly when operating outside the wake region. Consequently, the agent is forced to drift passively downstream, making it difficult to perform efficient navigation in the flow.

To navigate to a destination across a vortical wake, the swimmer is required to adopt a three-stage strategy. First, it cuts into the wake region; second, it maneuvers through the vortices along a serpentine trajectory to achieve upstream movement, leveraging the weak flow resistance inside the wake; finally, it escapes the wake region by following the downstream flow to reach the target (see Figure 1a). This serpentine movement is also observed in live fish [38] or inanimate self-propelled agents [39, 40], indicating that adaptive locomotion modes are indispensable when interacting with unsteady wakes. In the case where the spatiotemporal evolution of the flow field is fully known, such an ideal swimming path can be obtained based on optimal control approaches. However, in practical applications, underwater robotic or biological navigators often lack access to comprehensive spatiotemporal evolution information of the flow field [26]. On the other hand, reinforcement learning provides a viable solution to this problem: based on local, instantaneous observations of the target flow velocity and relative position, the agent can learn optimal path planning itself. It has been demonstrated that, provided that the observations are based on an inertial frame of reference, the problem of traversing unsteady wakes is controllable and can be solved via partial observations of the flow field. Nevertheless, in this paper, we aim to achieve efficient navigation solely relying on the agent's local observations from an egocentric perspective, without incorporating additional observation information (e.g., the flow gradients required in [34]).

2.2 Geocentric and Egocentric Observations

The geocentric observations are obtained in the Earth-centered frame (Figure 1b). As an inertial reference frame, the Earth-centered frame acts as an objective reference for characterizing the motion state of the agent within the flow field. Geocentric observations encompass three quantities: (1) the relative distance from the agent to the target, $\Delta \boldsymbol{x}_e = (\Delta x_e, \Delta y_e, \Delta z_e)$; (2) the global velocity of the agent, $\boldsymbol{v} = \boldsymbol{U}_{\text{swim}} + \boldsymbol{v}_{\text{flow}} = (v_{e,x}, v_{e,y}, v_{e,z})$; and (3) the heading direction of the agent in the inertial frame, (θ, ψ), with respect to its own velocity $\mathbf{U}_{swim}$. In practice, to obtain these observations, the agent needs to measure raw physical quantities via sensors deployed in its body-fixed frame, and then convert these quantities to the inertial frame using an attitude transformation matrix.

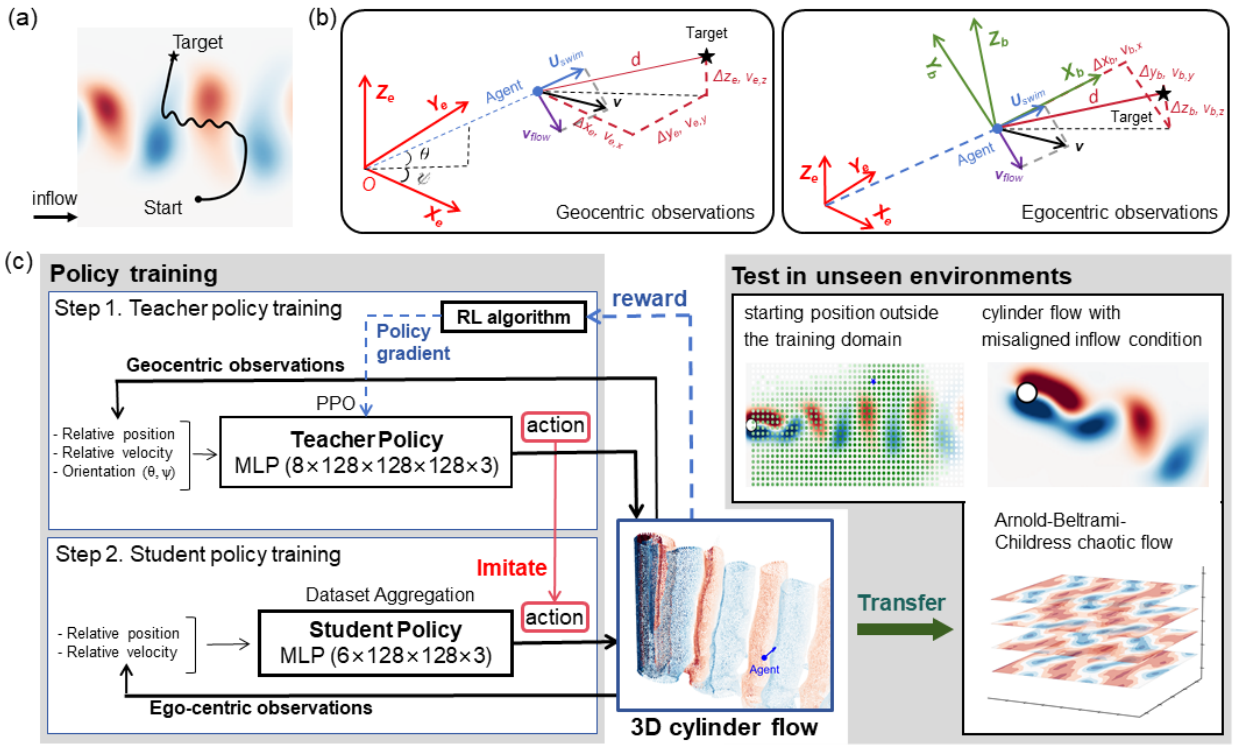

Figure 1: **Problem setup and imitation framework for learning to navigate in complex unsteady flows.** (a) Illustration of a swimmer navigating across the unsteady vortical flow. The serpentine trajectory is obtained as the swimmer's propulsion capability is inadequate to counteract the wake flow. (b) Definitions of geocentric observation and egocentric observation. (c) Training and testing based on the "teacher-student" framework. The teacher policy is trained via reinforcement learning, with access to geocentric observational information. Subsequently, the student policy learns by imitating the teacher action, relying solely on egocentric observational information. The teacher policy and student policy, which are both trained in the 3D cylinder wake flow, are transferred to unfamiliar environments for testing.

The egocentric observations are obtained in its body-fixed frame (Figure 1b), where the x-direction is assigned as the heading direction of the agent (that is, the orientation of $\mathbf{U}_{swim}$). The agent directly acquires perceptual information in its body-fixed frame, without prior knowledge of the free-stream direction or reliance on the inertial frame of reference. Here, the agent observes the relative distance to the target, $\Delta \boldsymbol{x}_b = (\Delta x_b, \Delta y_b, \Delta z_b)$, projected from the vector between the agent and the target (denoted by $\mathbf{d}$ in Figure1b). The velocity of the agent, $\boldsymbol{v} = (v_{b,x}, v_{b,y}, v_{b,z})$, is also observed based on the egocentric frame. Note that orientation angles (θ, ψ) with respect to the Earth-centered frame are no longer required, which eliminates the potential time delays and computational overhead associated with inertial signal evaluation [41]. The design of perceptual information for egocentric observation aligns well with the practical scenario where underwater robots can only perceive local, instantaneous environmental information, avoiding the introduction of global information beyond their physical sensing capabilities.

The distance, velocity, and orientation between the agent and the target are considered as observations in this paper, which are consistent with those used in existing literature [26, 27, 34]. However, there is a minor difference. The velocity observation term in relevant studies refers to the local flow velocity $\boldsymbol{v}_{\text{flow}}$, while that in this paper is the to-

tal velocity of the agent movement with respect to the target. This is motivated by the requirements of practical engineering implementation: measuring the position difference and relative velocity with respect to the target can be achieved merely by sonar systems, while observation of flow velocity requires additional deployment of a three-axis current meter, thus increasing the hardware overhead [42]. However, we argue that these two velocities are convertible to each other based on Equ.(2). To verify this conclusion, we have conducted experiments using the flow velocity as the velocity observation, which can be found in Supporting Information (SI) Section 1. The performances of using these two velocity definitions are similar, demonstrating that our proposed method is not constrained by specific forms of velocity observation and possesses excellent adaptability.

2.3 Imitation Framework for Navigation

2.3.1 Overview of Imitation Framework With Teacher-Student Models

The imitation learning framework employed in this study is illustrated in Figure 1c. Geocentric and egocentric observations correspond to the information acquired from the inertial coordinate frame and the body-centric coordinate frame, respectively. First, the teacher policy is trained using the Proximal Policy Optimization (PPO) algorithm, which takes perceptual information under geocentric observations as input and outputs the corresponding agent actions. Subsequently, the student policy is trained with the Dataset Aggregation (DAGGER) algorithm, which receives egocentric observational information as input and yields actions of the same type as those generated by the teacher policy. During the training phase of the student policy, the agent's actions are fully and autonomously generated by the student policy, while the teacher policy only provides action demonstrations and guidance as a supervisory signal.

All training processes are conducted in the three-dimensional flow field around a circular cylinder, which simulates continuous vortex streets generated in the wake of the cylinder, thereby providing complex flow disturbance conditions that are highly consistent with practical engineering scenarios for policy training. After the teacher policy and student policy complete the basic training and both achieve stable navigation performance in the training flow environment, further transfer tests are carried out to verify their navigation performance when confronted with unseen states during the training phase (see Figure 1c).

2.3.2 Teacher Policy Learned from Geo-observations

To highlight that the teacher policy is trained with geocentric observations, we refer to it as "Teacher (-geo)" throughout the paper. The teacher (-geo) is trained based on the PPO algorithm of reinforcement learning, which adopts a classical actor-critic architecture. The actor network outputs the mean value of the action distribution as the policy decision according to the input state, while the critic network learns the value function to evaluate the quality of the current state.

The navigation problem can be formulated as a Markov Decision Process (MDP), in which the probability distribution of future states depends only on the current state and the action selected at the current time step, and is independent of the historical states and actions prior to the current state. This MDP is composed of a state space S, an action space A, a reward function R, and a state transition probability P. The objective of the

MDP is to enable the agent to maximize the long-term cumulative reward starting from the current state via rational action selection.

The state space of the teacher model is defined as

$$S = \{\Delta x_e, \Delta y_e, \Delta z_e, v_{e,x}, v_{e,y}, v_{e,z}, \theta, \psi\} \tag{4}$$

where the vector $\{\Delta x_e, \Delta y_e, \Delta z_e\}$ denotes the position differences between the agent and the target observed in the geocentric inertial coordinate system, $\{v_{e,x}, v_{e,y}, v_{e,z}\}$ represents the velocity of the agent in the geocentric inertial coordinate system, $\{\theta, \psi\}$ indicates the direction of the agent's velocity vector $\mathbf{U}$ in the geocentric inertial coordinate system, which also describes the pose relationship between the agent-fixed body coordinate system and the world coordinate system. The acquisition of these observation quantities has been discussed in Section 2.2 and illustrated in Figure 1b.

The action space for the teacher model is defined as

$$A = \{U_{\text{swim}}, \dot{\theta}, \dot{\psi}\} \tag{5}$$

where U_{swim} denotes the magnitude of the agent's own velocity, $\dot{\theta}$ is the pitch angular velocity of the agent (in units of rad/s) with a range of $\left[-\frac{\pi}{4}, \frac{\pi}{4}\right]$ rad/s, and $\dot{\psi}$ is the yaw angular velocity of the agent with a range of $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ rad/s. The action space can uniquely define the motion of the agent, as specified in Equations (1-2).

The following requirements are considered in the design of the reward function. It must simultaneously satisfy the goals of reaching the navigation destination and achieving efficient navigation. In addition, the reward function should continuously change with each navigation time step. The reward function at the n-th time step can be defined as:

$$r_n = -\gamma_n \cdot \Delta t + \alpha \cdot (d(p_n, p_{\text{goal}}) - d(p_{n-1}, p_{\text{goal}})) + r_{\text{success}} \tag{6}$$

where $d(p_n, p_{\text{goal}})$ denotes the Euclidean distance between the agent and the target at the n-th time step. The first part of the reward function penalizes the navigation time, fulfilling the requirement of minimizing navigation duration. Furthermore, a normalized attenuation coefficient γ_n that decreases as the distance between the agent and the target reduces is incorporated, thereby enabling the agent to reduce its focus on efficiency and concentrate on navigation accuracy at the end of the trajectory. The second term calculates the distance difference between the agent and the target over two consecutive time steps, incentivizing the agent to approach the target. A sparse reward r_{success} is granted to the agent when the distance between the agent and the target is within a certain small scale (which is 0.3 in this paper). Assuming a complete motion trajectory of the agent requires N time steps, the total reward function is expressed as follows:

$$r_{\text{total}} = \sum_{n=1}^{N} r_n = -\sum_{n=1}^{N} \gamma_n \cdot T_{\text{total}} + \alpha \cdot D + r_{\text{success}} \tag{7}$$

where T_{total} denotes the total navigation time of the agent, and D represents the distance between the starting point and the target point.

2.3.3 Student Policy Learned from Ego-observations

As the student policy is trained with egocentric observations, we refer to it as "Student (-ego)" hereafter. The student (-ego) is trained via the DAGGER algorithm, where the state

input of the student model is defined as

$$S = \{\Delta x_b, \Delta y_b, \Delta z_b, v_{b,x}, v_{b,y}, v_{b,z}\} \tag{8}$$

where $\{\Delta x_b, \Delta y_b, \Delta z_b\}$ denotes the positional differences between the agent and the target observed in the body-fixed coordinate system, and $\{v_{b,x}, v_{b,y}, v_{b,z}\}$ represents the velocity of the agent in the body-fixed coordinate system. Figure 1b illustrates the acquisition of these observation quantities.

On the other hand, the action space of the student model, namely the output of the model, is defined identically to that of the teacher model in Equation 5. We note that as the action is controlled by the angular velocities rather than the angles, the action is independent to the coordinate system.

As for the training of DAGGER algorithm, the student (-ego) generates actions based on its egocentric observational inputs and interacts with the environment to collect interaction data. Meanwhile, the trained teacher model synchronously acquires the geocentric observations corresponding to the current state and generates actions [43] that can lead to successful navigation. After sufficiently exploring the flow environment, the data pairs of egocentric student observations and expert actions are collected into the experience buffer, achieving the cumulative expansion of the training dataset. This mechanism expands the coverage of training data through the continuous interaction between the student (-ego) and the environment, and mitigates the distribution shift problem in imitation learning by virtue of expert corrective guidance.

The student (-ego) takes the aim of minimizing the mean squared error (MSE) between the predicted actions and the teacher actions as the optimization objective, with the Adam optimizer employed for parameter updates. The loss function is simply defined as:

$$\mathcal{L}(\phi) = \frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{d} \left(\hat{a}_{i,j}(\phi) - a^*_{i,j} \right)^2 \tag{9}$$

where ϕ denotes the network parameters of the student (-ego), N is the number of samples in the training batch, and d is the output dimension of the student (-ego), which is equal to 3; $\hat{a}_{i,j}(\phi)$ and $a^*_{i,j}$ represent the output value by the student model and the demonstration value generated by the expert model, respectively.

In addition to the student (-ego) model, which is trained by imitating the teacher (-geo) action, we introduce an independent baseline (-ego) model for comparison.

The **baseline (-ego)** model is trained and interacted with the environment based on PPO algorithm, which is designed to verify whether egocentric navigation policies can be achieved based on reinforcement learning without additional measurements. The difference between the baseline (-ego) and the teacher (-geo) lies in the observation terms. Intuitively, the observations of the teacher (-geo) are derived from the geocentric reference frame, whereas those of the baseline (-ego) are obtained from the egocentric reference frame. Meanwhile, the distinction between the baseline (-ego) and the student (-ego) is as follows: the former is an unsupervised independent reinforcement learning paradigm that attempts to inherently acquire navigation strategies by itself, essentially serving as a direct validation of the feasibility of the *egocentric observation + DRL* framework. In contrast, the latter operates within the privileged learning framework, taking the output of the teacher (-geo) as the supervision signal and distilling the global optimization logic into local decision-making via the DAGGER algorithm.

3 Results

3.1 Flow Environment and Training Process

We consider an autonomous swimmer navigating through the unsteady wake of a three-dimensional flow past a circular cylinder. The flow field around the cylinder is generated by computational fluid dynamics (CFD) simulations of an incompressible fluid at a Reynolds number of 300. In such a flow field, the incoming flow generates a continuous array of vortex street behind the cylinder, posing significant challenges for navigation across the wake flow. The dimensionless incoming flow velocity is set to 1.0, and the cylinder diameter is 1.0. The agent is assigned a maximum free swimming speed of 1.0, which renders it unable to reach the target point directly in a straight path across the wake flow.

As shown in Figure 2a, we define two cuboid regions in the flow field, where the coordinate range of Region 1 is $(x, y, z) \in [8, 12] \times [2, 4] \times [3, 7]$, and that of Region 2 is $[8, 12] \times [-4, -2] \times [3, 7]$. During training, for each navigation task, an initial position and a target position are randomly generated in Region 1 and Region 2, respectively, ensuring that the agent must navigate through the vortex wake region downstream of the cylinder, which increases the complexity and stochasticity of the navigation problem. The initial orientation of the agent for each navigation trial is also randomly assigned, leading to greater diversity in the initial states of the navigation tasks.

We adopt the PPO algorithm to train the baseline (-ego) and the teacher (-geo) in the three-dimensional flow past a circular cylinder. Each of the two policies was independently trained five times, with each training run containing 3×10^7 iterations. From the reward evolution shown in Figure 2b, the learning processes of the two policies can be divided into an initial synchronous stage and a late divergent stage. At the beginning, the average episode reward curves of the teacher (-geo) and the baseline (-ego) overlap highly and remain at a low level, where the learning of the agent for both policies focuses on acquiring low-level motion control capabilities. As the training time steps further increase, the average episode reward of the teacher (-geo) shows a continuous rising trend, which tends to stabilize after the training time steps reach 2×10^7; and the reward curves of the five independent training runs present an extremely high consistency of convergence. on the contrary, the average episode reward of the baseline (-ego) plateaus and stays at a small reward value, leading to weak navigation performance.

Subsequently, we use teacher (-geo) model to supervise the training of the egocentric observation-based student (-ego), which was also independently trained for five times in total to migrate randomness of the network parameters. Figure 2b shows the variation in the loss of the student (-ego) model during the training process. Within the first five iterations of DAGGER, the loss rapidly decreases from an initial value of 0.8 to 0.15, after which the loss curve gradually becomes smooth and flat.

3.2 Navigation Performance in the Training Environment

We evaluate the navigation performance of the three aforementioned policies, with 10,000 independent episodes tested for each policy. In summary, the teacher (-geo) achieves a navigation success rate of 94.69%, with an average time of 26.92 s for successful navigation tasks, while the baseline (-ego) fails to complete any navigation task in all test episodes. The student (-ego) reaches a navigation success rate of 94.44%, and the average time for

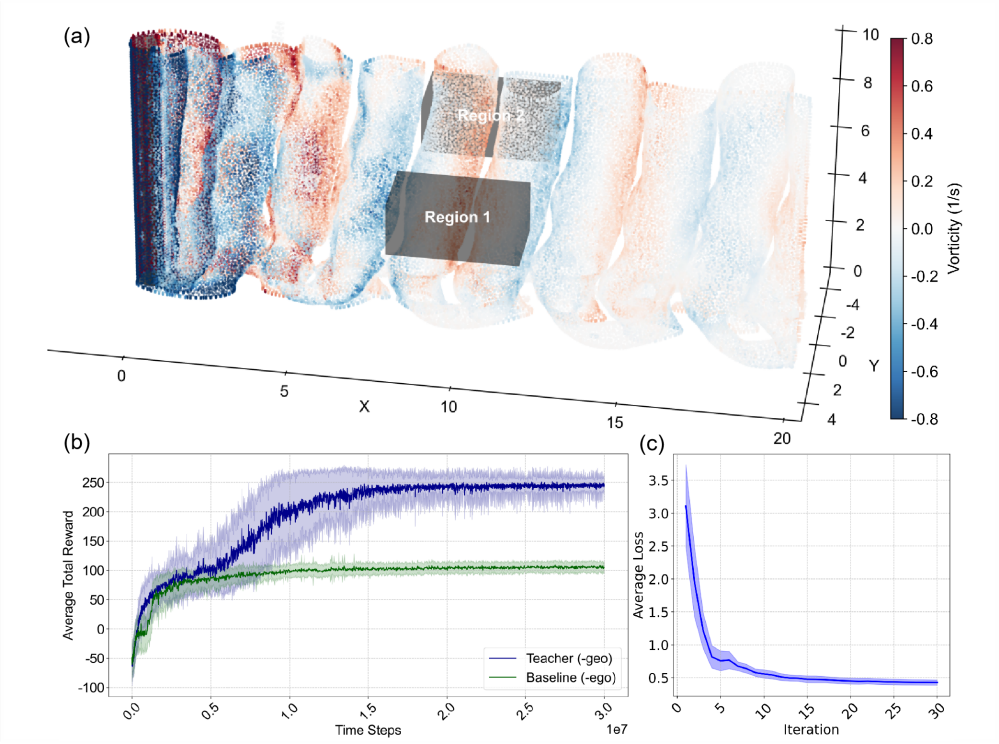

Figure 2: **Flow environment settings and learning progress during training.** (a) 3D flow around a cylinder with the z-component of vorticity color-coded. During training, the start and target points are randomly generated within the masked regions. (b) Reward functions of the teacher (-geo) and baseline (-ego) during training. (c) Loss function of the student (-ego) during training. Five independent trainings are implemented for each policy. Dark solid lines in (b-c) represent the mean values over five runs, while light shaded areas denote the standard deviations.

successful navigation was 25.31 s. It can be seen that the navigation performance of the student (-ego) in the training environment is essentially consistent with that of the teacher (-geo), as it imitates the action of teacher model during training.

To visualize more details, an example of the navigation trajectories of three policies are presented in Figure 3 as well as Movie 1 in SI. By analyzing the three-dimensional motion trajectories in Figure 3a, it can be clearly observed that the well-trained policies exhibit three prominent phases during the navigation process: entrainment into the wake flow, swirling upstream within the vortex region, and detachment from the wake flow to move toward the target area utilizing the downstream flow. In contrast, the baseline (-ego) fails to perform the swirling upstream maneuver in the vortex flow. For a more intuitive visualization of the navigation trajectories, we extract the XOY cross-section which contains the starting point, and project the corresponding trajectories on this 2D plane, as demonstrated in Figure 4b. It can be seen that after entraining into the wake flow, both the teacher (-geo) and the student (-ego) consciously avoid regions with high flow velocity magnitude and move upstream along the wake flow in a serpentine motion pattern. Conversely, the baseline (-ego) strategy prioritizes identifying the shortest path to the target, thus moving upstream and proceeding directly toward the target point upon entering the wake region. Consequently, its inadequate self-propulsion capability renders it entirely incapable of reaching the destination. This finding corroborates the fact that a naive DRL-

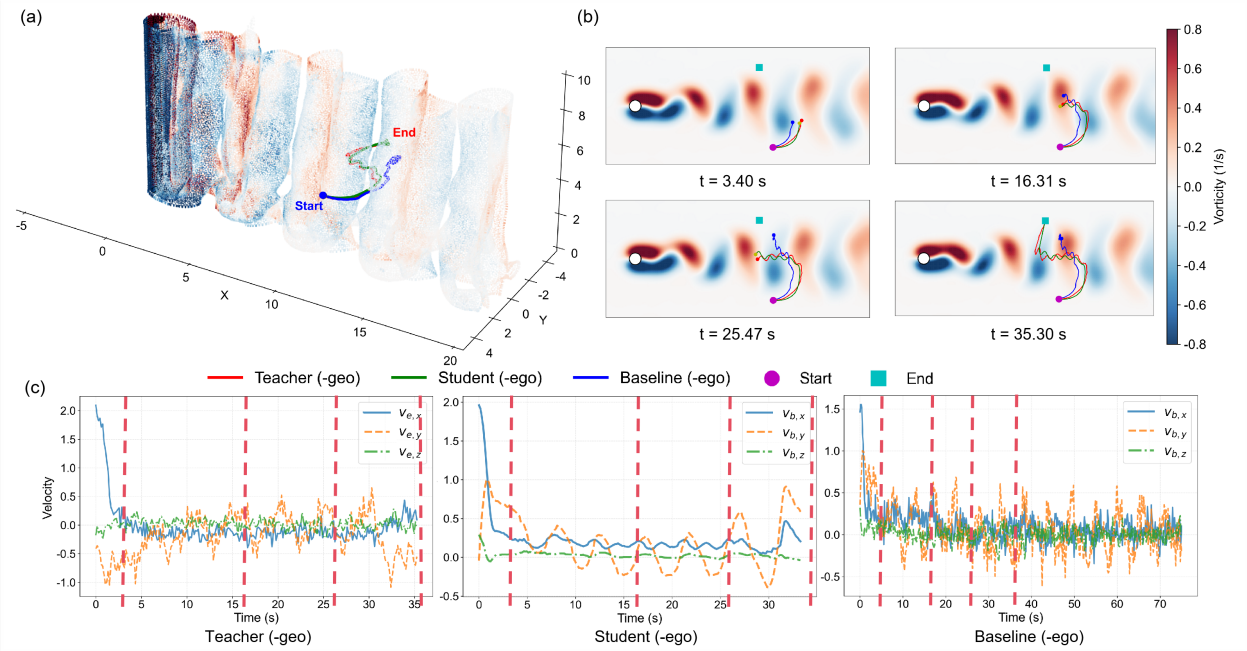

Figure 3: **Navigation performance of different RL policies in 3D cylinder flow.** (a) Navigation trajectories of the teacher (-geo), baseline (-ego) and student (-ego) models. The background flow field is presented as isosurfaces of Q-criterion, color-coded by the z-component of vorticity. (b) Navigation trajectories at different time steps. The z-component of vorticity on the XOY cross-section at the initial position is plotted as the background. (c) Velocity variations of different policies during navigation. The vertical red dashed lines correspond to the four time instants of the navigation process in (b).

PPO model cannot achieve egocentric navigation relying solely on velocity measurements without supplementary flow velocity gradient information—a result highly consistent with the conclusion reported in [34]. However, we further demonstrate that successful egocentric navigation using only velocity measurements is attainable via our proposed imitation learning framework, thus highlighting the superiority of our work over existing methods.

The underlying cause of this phenomenon can be analyzed from the perspective of velocity observation in different coordinate systems. For the agent to achieve upstream locomotion, the velocity observation term $v_{e,x}$ under geocentric observation must take a negative value (i.e., $v_{e,x} < 0$). However, the agent's maximum propulsive speed is equivalent to the free-stream flow velocity, allowing it to only overcome the incoming flow and achieve upstream locomotion inside the wake where the flow velocity is small. For the teacher (-geo) model, $v_{e,x}$ is directly observed in the state space, enabling the straightforward identification of whether the agent is located in a low-speed flow region. As illustrated in Figure 3(c, left), when the agent with teacher (-geo) policy moves upstream along a zigzag trajectory upon entering the wake, i.e., $t \in (5, 30)$ s, $v_{e,x}$ remains predominantly negative.

In contrast, the egocentric agent measures the velocity in its body frame, where $v_{b,x}$ denotes the velocity component along the agent's propulsion direction and cannot directly reflect the magnitude of the ambient flow velocity. In other words, a positive $v_{b,x}$ means that the agent is moving forward relative to its own propulsion orientation, but this does not guarantee upstream motion relative to the incoming flow. As demonstrated in Figure 3(b), the baseline (-ego) consistently moves straight toward the target upon entering the

wake; yet its propulsion capability is weaker than the incoming flow velocity, meaning it cannot navigate upstream along a zigzag trajectory and is ultimately trapped in the vortices. As shown in Figure 3(c, right), $v_{b,x}$ of the baseline (-ego) model oscillates continuously around zero after entering the wake, which indicates the agent is unable to identify the flow velocity and advance steadily along its intended propulsion direction.

Surprisingly, we find that $v_{b,x}$ of the student (-ego) remains consistently positive throughout the navigation trajectory (Figure 3c, middle), demonstrating that the actual locomotion path is always aligned with the agent's intended propulsion direction. This result illustrates that the student (-ego) develops a decision-making logic to identify effective propulsion regions by imitating the behavior of the teacher (-geo) model, ultimately moving upstream inside the wake and achieving successful navigation. Moreover, the observation terms of the student (-ego) exhibit a high degree of smoothness compared to those of the other two policies (see Figure 3c). This can be attributed to the DAGGER training for the student (-ego) model, which minimizes the mean squared error between its actions and those of the teacher (-geo). Consequently, the actions outputted by student (-ego) are smoothed to some extent. This result indicates that the effective behavioral strategies of the teacher (-geo) model are retained, while spurious fluctuations arising from global parameter adaptation are filtered out, further highlighting the superiority of our work.

In summary, we demonstrate that the student (-ego) model which imitates the teacher (-geo) can achieve successful egocentric navigation in the training flow environment, i.e., the 3D flow past a cylinder. Subsequently, to evaluate the generalization capability of the student (-ego) and the teacher (-geo), we transfer both policies to unfamiliar flow environments for testing without any further training and tuning.

3.3 Transferring Navigation Policies to An Extended Domain

To validate the adaptability of both the teacher (-geo) and student (-ego) models to unseen initial conditions within the training environment, we conduct a systematic test wherein the agent is tasked with navigating from random starting positions across the entire computational domain — including regions not explored during training — to a fixed target at coordinates $(11.5, -3.5, 4.5)$. The full domain is partitioned into 1000 subregions, and the navigation success rate for each subregion is calculated when the agent's starting position falls within its bounds. Each policy is evaluated over 50,000 episodes to ensure that the test data adequately covers all subregion and guarantees statistical robustness.

By comparing the two policies in Figure 4(a-b), we observe that the student (-ego) model demonstrates superior generalization performance to the teacher (-geo) model in the upper region of the wake flow. When initialized at starting positions on the same side of the wake flow as the target point, the teacher (-geo) model largely loses its navigational capability. Although the student (-ego) model achieves superior performance across the full computational domain, it exhibits the same limitation when initialized directly upstream of the target. Intuitively, an agent positioned directly upstream of the target should be able to reach the destination via downstream advection, even a naive navigator that moves straight toward the target while completely disregarding the flow field. A moderately sophisticated navigator, equipped with knowledge of the background uniform flow, could further leverage this flow direction to reach the target from a broader range of upstream positions. However, both policies fail to retain this intuitive navigational capability to a notable extent.

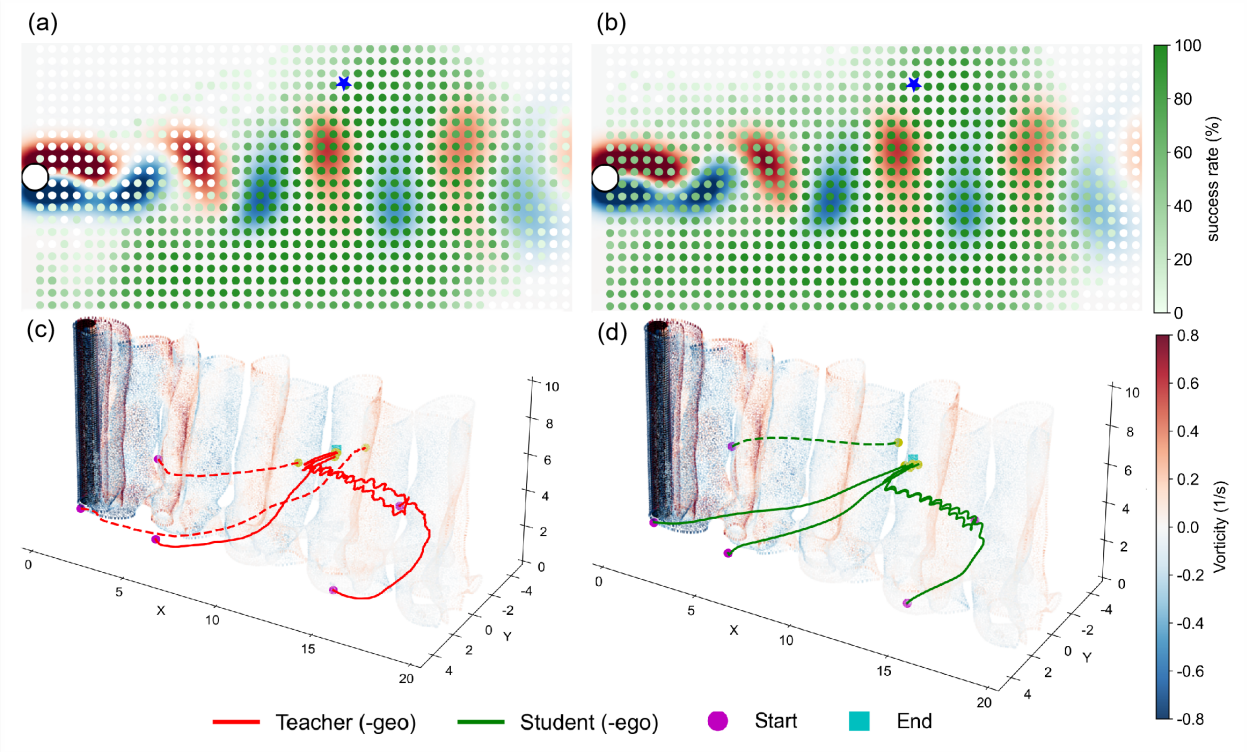

Figure 4: **Performance of transferring navigation policies to an extended domain.** The agent is tasked with navigating from random starting positions across the entire computational domain to a fixed target point. (a-b) Navigation success rates of teacher (-geo) and student (-ego) models, respectively. The background displays the XOY cross-section with z-component vorticity color-coded, overlaid on the domain discretized into 1000 subregions. Green color coding indicates the policy's success rate for departures from each subregion, and asterisks mark the fixed target point. (c-d) Representative navigation trajectories initiated from five typical initial positions in the flow field, where solid lines represent successful trajectories and dashed lines represent failed ones.

We attribute this performance shortfall to the training constraints imposed on the policies. Given that the agent is consistently trained to navigate from starting positions along the positive Y-axis positions (in Region 1 of Figure 2) to target points along negative Y-axis (in Region 2), the policies develop a form of locomotion inertia in their navigation behavior. To validate this hypothesis and address this limitation, we conduct additional training experiments in the Supporting Information, where the starting points were randomly sampled from either region 1 or region 2, and the target points were randomly generated from the other one. The student (-ego) trained under this modified setting achieves remarkably high navigation performance within the entire domain. More details can be found in SI Section 2.

In addition, we observe from Figure 4(a-b) and SI that both policies exhibit degraded navigation performance in the downstream region. This performance decline is primarily attributed to the further decay of vortices in the downstream, which gives rise to elevated flow velocities within the wake and thus hinders the agent from moving upstream against the wake flow. Furthermore, the agent is positioned in close proximity to the flow field boundary in this region, increasing the likelihood of colliding with the boundary prior to entering the wake flow—an outcome that is defined as a navigation failure.

Examples of the navigation trajectories are demonstrated in Figure 4(c-d), where we find that the failure cases are concentrated primarily in the upstream region on the same side of the target. When the agent departs from this region, its self-propulsion velocity superimposes with the incoming flow velocity to generate a large resultant velocity. Coupled with the locomotion inertia accumulated during training, this effect makes the agent highly prone to overshooting the target point, ultimately leading to navigation failure via collisions with the flow field boundaries. In contrast, when the agent initiates navigation from the opposite side of the wake, it can exploit the wake flow to move upstream or traverse directly across the wake region, where velocity magnitudes are relatively low.

3.4 Transferring Navigation Policies to Misaligned Cylinder Flow

We impose a rotation of the free-stream direction relative to the z-axis of the inertial frame, which thereby introduces a misalignment between the wake direction and the inertial frame. We then test the navigation performance of both the teacher (-geo) and student (-ego) models across a range of misalignment levels.

Figure 5 illustrates the navigation performance of the two policies in the misaligned cylinder wake flows. It can be observed that the success rate of the teacher (-geo) declines rapidly with the increase in the misalignment between the wake flow and the reference frame, whereas the student (-ego) maintains its high performance at all misalignment levels. With increasing misalignment, the trajectories of the teacher (-geo) gradually become chaotic, and the agent becomes trapped in regions of strong vorticity. In contrast, the trajectories of the student (-ego) maintain a smooth convergence regardless of the variations in the misalignment angle. This result is intuitively demonstrated in Movie 2 in the Supporting Information. As shown in Figure 5(e), the success rate of the teacher (-geo) drops dramatically with increasing misalignment level, while the student (-ego) maintains a consistently high success rate when the flow condition changes.

These results demonstrate that the policy based on egocentric observations can sustain high navigation performance in cylinder wake flows across all inflow directions, as it adapts the agent's behavior according to local flow measurements without information of the inertial frame. In contrast, the policy trained with geocentric observations learns fixed flow field and motion patterns, and thus relies on prior knowledge of the alignment between the wake flow and the geocentric frame for effective navigation. For practical application of the geo-centric policy, the incoming flow direction of the current flow field must be obtained in advance. However, in real-world environments such as ocean circulations and low-altitude turbulent airflows, flow fields exhibit strong dynamic characteristics. The incoming flow direction may undergo instantaneous variations driven by vortex generation, ocean current convergence, or gust disturbances—changes that cannot be pre-acquired via static measurements. Therefore, the teacher (-geo) model without generalization ability is not practically feasible for deployment when the agent operates in diverse conditions.

3.5 Transferring Navigation Policies to Completely Unseen Flow

The Arnold-Beltrami-Childress (ABC) flow is a canonical three-dimensional incompressible time-periodic flow field that yields an exact solution to the Navier-Stokes equations. Its key characteristics are manifested in the spatial entanglement of vortex tubes and the long-term chaoticity of motion trajectories. In contrast to the regular three-dimensional

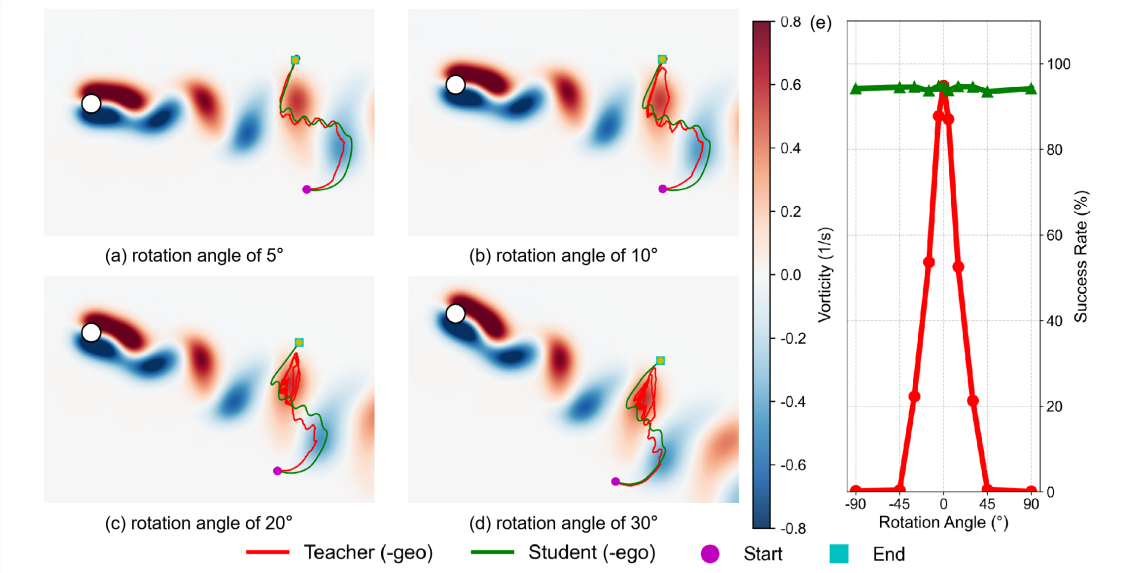

Figure 5: **Transferring navigation policies to misaligned 3D cylinder wake flow.** (a-d) Navigation trajectories under different misalignment levels. The positions of starting and target points relative to the inflow are identical in different cases. (e) Navigation success rates in the cylinder wake flow at different misalignment levels. The success rate of the teacher (-geo) degrades dramatically with increasing misalignment levels, while the student (-ego) maintains a consistently high success rate.

cylinder wake flow, the 3D ABC flow lacks a fixed vortex street structure and a predictable vortex core shedding law. Instead, it forms a dynamic undisturbed vortex system via the coupling effect of flow parameters, making it analogous to complex natural flow environments such as atmospheric convective turbulence, deep-sea ocean circulation, etc. The analytical equations for the ABC flow are provided in the Supporting Information Section 3.

We perform zero-shot transfer tests on the teacher (-geo) and student (-ego) policies in the ABC flow field, with all test parameters presented in dimensionless form. The maximum flow velocity of this flow environment is approximately 1.15. Additionally, the action component U_{swim} output by the policy is scaled by a factor of 0.1, which reduces the agent's maximum propulsive speed from 1 to 0.1 and thus elevates the navigation difficulty. For each navigation task, the start position is randomly generated within the region defined as $(x, y, z) \in [-1.8\pi, -1.5\pi] \times [-0.4\pi, 0.4\pi] \times [-0.1\pi, 0.3\pi]$, while the target position is randomly sampled from the domain $[1.5\pi, 1.8\pi] \times [-0.4\pi, 0.4\pi] \times [-0.1\pi, 0.3\pi]$, requiring the agent to navigate across time-varying and periodic vortices.

We validate the navigational capabilities of the trained policies in this unseen flow field environment. Through 10,000 independent test episodes, we find that the teacher (-geo) model completely fails to achieve successful navigation, while the student (-ego) model attains a navigational success rate of 75.9%. Trajectory visualizations in Fig. 6(a-b) provide an intuitive illustration that the teacher (-geo) model completely loses its navigational capability: it adheres to its pre-learned motion pattern and tends to move upward regardless of the target position. In contrast, while the student (-ego) model exhibits a reduced navigational success rate, it still navigates toward the target position and actively avoids high-velocity flow regions, thus maintaining acceptable navigational performance.

Such a performance gap between the two policies stems primarily from the fact that

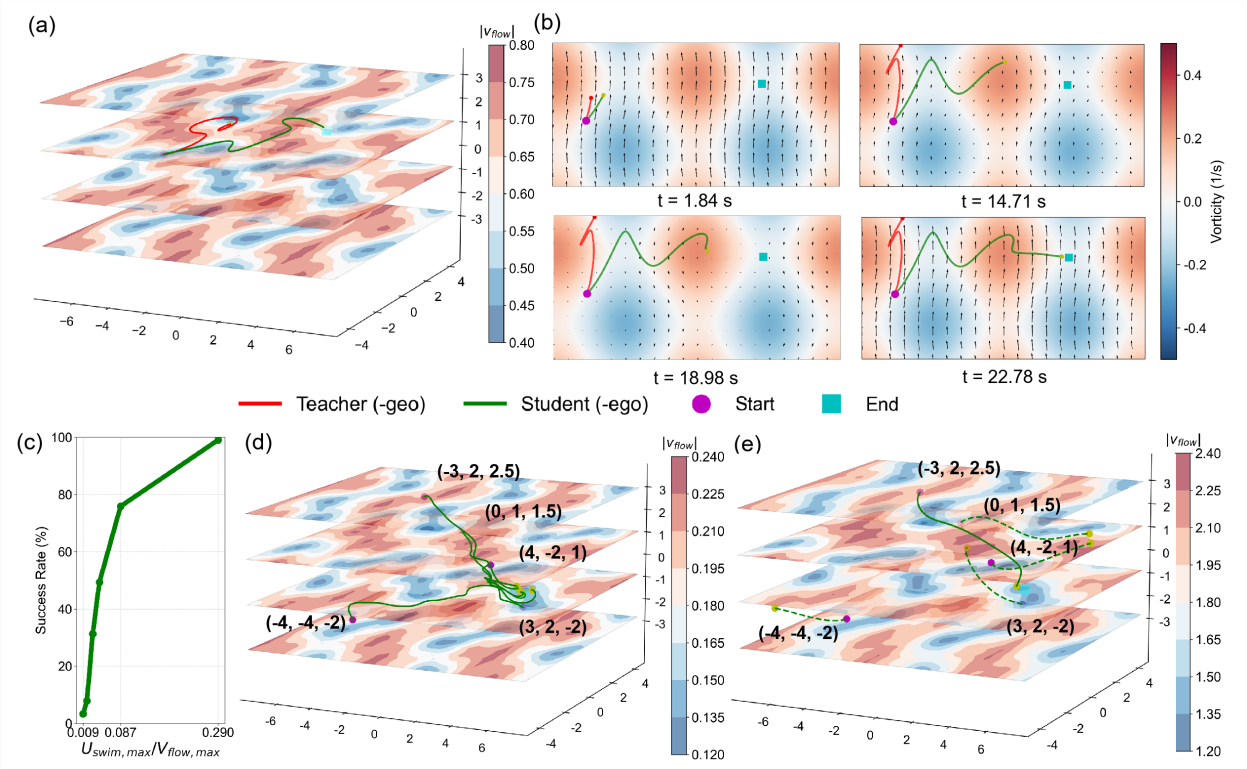

Figure 6: **Trasferring navigation policies to a completely unseen flow environment.** (a) An example of the navigation trajectories of teacher (-geo) and student (-ego), with $U_{\text{swim,max}}/V_{\text{flow,max}} = 0.1/1.15 = 0.087$. Several 2D slices representing the magnitude of flow velocity are demonstrated to highlight the 3D effect of the flow field. (b) Navigation trajectories at different time steps. The z-component of vorticity on the XOY cross-section at the starting point is plotted as the background. (c) Navigation success rate of the student (-ego) model against different ratios of $U_{\text{swim,max}}/V_{\text{flow,max}}$. (d-e) Navigation trajectories from five initial points to a fixed target for the cases of $U_{\text{swim,max}}/V_{\text{flow,max}} = 0.290$ and 0.029, respectively. Solid lines denote successful navigations and dashed lines represent failed ones.

the teacher (-geo) model learns a task-specific strategy tailored to fixed environmental parameters by relying on privileged inertial frame observation information, whereas the student (-ego) model acquires generalized decision-making logic for adapting to the dynamic characteristics of arbitrary flow fields through local perception alone. More specifically, the student (-ego) learns a generalized navigation logic that is independent of specific flow field parameters and grounded solely in real-time local perception. Taking the real-time local states in the body-fixed frame as inputs, the student (-ego) distills the key decision-making experience of the teacher (-geo) via the DAGGER algorithm without replicating the latter's adaptive behaviors tailored to the training flow field parameters. Instead, it extracts a generalized mapping between perceiving the instantaneous flow field states and dynamically adjusting its own actions, establishing a direct link between real-time local perception and adaptive locomotion decisions. This fundamental distinction endows the student (-ego) with superior generalization performance in unfamiliar flow field environments.

In addition, we note that the ratio of $U_{\text{swim,max}}/V_{\text{flow,max}}$, which limits the propulsion ability of the agent, affects the navigation performance. Therefore, we perform an ablation experiment to investigate the performance of student (-ego) by adjusting $V_{\text{flow,max}}$ in the

ABC flow. As shown in Figure 6(c), the navigational success rate decreases gradually as the flow velocity increases (equivalent to a reduction in the ratio $U_{\text{swim,max}}/V_{\text{flow,max}}$). This phenomenon occurs because the chaotic vortex perturbations of the flow field progressively exceed the agent's propulsion control range at higher flow velocities, leaving the agent unable to rapidly modulate its self-propulsion velocity or evade dynamically evolving strong vortex regions. In such scenarios, the agent is prone to behaving as a passive particle advected by the flow (as shown in Figure 6e).

4 Discussion and Conclusion

In this work, We propose a "teacher-student" DRL framework based on imitation learning. Within the same flow field environment, the teacher (-geo) model, which leverages global geocentric observations, is first trained using the PPO algorithm. Subsequently, the student (-ego) model, which only relies on local egocentric observations (a setup more consistent with real-world experimental scenarios), learns to mimic the teacher's behavioral strategies through the dataset aggregation (DAGGER) algorithm. Ultimately, enables high-precision navigation in complex flow fields without the need for global environmental information (e.g., the approach in [26]) or additional flow field observation data (e.g., the flow field gradient employed in [34]).

This proposed egocentric student policy framework is highly consistent with the biological concept of "Umwelt" in nature [35, 36], a principle by which aquatic organisms such as fish and aquatic invertebrates achieve navigation in unsteady flow fields. These organisms forgo global information from an external inertial frame of reference, instead relying solely on egocentric local perception to sense their surrounding environment and make navigational decisions. By design, the egocentric student policy depends only on local and instantaneous environmental information as well, which not only matches the actual perceptual capabilities of miniaturized carriers and aligns with the bionic perceptual logic of aquatic organisms, but also eliminates the hardware overhead incurred by additional sensing equipment.

More importantly, the proposed student (-ego) model exhibits strong generalization capacity for navigation when directly transferred to unfamiliar flow field environments. The teacher (-geo) model, which requires external reference devices such as satellites and compasses to achieve the alignment of a fixed inertial frame, tends to learn task-specific motion pattern tailored to fixed flow field conditions. By comparison, the student (-ego) operates entirely independent of external support and is able to extract generalized navigation logic through imitation learning, making it well-adapted to practical application scenarios lacking stable external references. In our systematic experiments, the student (-ego) maintains exceptional performance even in unseen flow field scenarios, including misaligned cylinder wakes and chaotic Arnold-Beltrami-Childress flow, demonstrating robust generalization beyond the training environment, a property that is critical for real-world deployment.

Our work also presents the first successful attempt of applying DRL to navigation tasks in 3D complex flow fields, breaking the long-standing limitation whereby existing DRL-based navigation studies are predominantly validated in two-dimensional domains. Unlike 2D models which fail to capture key 3D flow characteristics (e.g., vertical oscillation of vortex cores), the proposed framework accommodates the multi-degree-of-freedom

motion requirements of underwater or aerial robots operating in real-world 3D space.

We note that a simplified kinematic model is adopted in this work to describe the motion of the agent, following pioneering studies in this field [26, 33, 34]. Although this simplification helps to demonstrate the main research concept, future work will incorporate more complex locomotion model of the agent to enhance the proposed framework's alignment with real-world practical scenarios. Moreover, building on the strong generalization across diverse flow field scenarios achieved via sole on-board perception, future research can further advance sim-to-real transfer for this framework: by integrating the physical attributes and sensor noise of real-world scenarios into the simulation environment, and combining this with domain adaptation techniques to mitigate the distribution shift between simulation and reality, the trained student (-ego) model can be directly deployed on physical carriers. This thus paves a promising direction for autonomous navigation of underwater robots, aerial vehicles and micro-swimmers operating in diverse real-world environments coupled with complex unsteady flow fields.

Supporting Information
We provide Supporting Information which contains ablation experiments and details of the flow simulation. We also provide the movies corresponding to the main results.

Acknowledgements
This work was supported by the Key Research and Development Program of Zhejiang Province No. 2024C03276(SD2) and Fundamental Research Funds for the Central Universities.

Conflict of Interest
The authors declare no Conflict of interest.

Data Availability Statement
The data that support the findings of this study will be publicly available on Github upon publication.

References

[1] R. N. Smith, Y. Chao, P. P. Li, et al. Planning and implementing trajectories for autonomous underwater vehicles to track evolving ocean processes based on predictions from a regional ocean policy. *International Journal of Robotics Research*, 29(12):1475–1497, 2010.

[2] H. M. P. C. Jayaweera and S. Hanoun. Path planning of unmanned aerial vehicles (uavs) in windy environments. *Drones*, 6(5):101, 2022.

[3] F. Achermann, T. Stastny, B. Danciu, et al. Windseer: real-time volumetric wind prediction over complex terrain aboard a small uncrewed aerial vehicle. *Nature Communications*, 15(1):3507, 2024.

[4] H. Cao, J. Shen, Y. Zhang, Z. Fu, C. Liu, S. Sun, and S. Zhao. Proximal cooperative aerial manipulation with vertically stacked drones. *Nature*, 646:576–583, 2025.

[5] G. Dehnhardt, B. Mauck, and H. Bleckmann. Seal whiskers detect water movements. *Nature*, 394:235–236, 1998.

[6] P. Oteiza, I. Odstrcil, G. Lauder, R. Portugues, and F. Engert. A novel mechanism for mechanosensory-based rheotaxis in larval zebrafish. *Nature*, 547:445–448, 2017.

[7] P. Weber et al. Optimal flow sensing for schooling swimmers. *Biomimetics*, 5:10, 2020.

[8] L. Techy. Optimal navigation in planar time-varying flow: Zermelo's problem revisited. *Intelligent Service Robotics*, 4(4):271–283, 2011.

[9] L. Biferale, F. Bonaccorso, M. Buzzicotti, et al. Zermelo's problem: optimal point-to-point navigation in 2d turbulent flows using reinforcement learning. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 29(10):103138, 2019.

[10] D. Kularatne, S. Bhattacharya, and M. A. Hsieh. Going with the flow: a graph based approach to optimal path planning in general flows. *Autonomous Robots*, 42(6):1369–1387, 2018.

[11] M. Panda, B. Das, B. Subudhi, et al. A comprehensive review of path planning algorithms for autonomous underwater vehicles. *International Journal of Automation and Computing*, 17(3):321–352, 2020.

[12] K. Kartik, S. Zhuoyuan, and S. L. Brunton. Finite-horizon, energy-efficient trajectories in unsteady flows. *Proceedings of the Royal Society A*, 478(2258):20210255, 2022.

[13] T. Larrabee, H. Chao, M. Rhudy, et al. Wind field estimation in uav formation flight. In *2014 American Control Conference*, pages 5408–5413. IEEE, 2014.

[14] I. D. Landau, L. Ljung, and S. S. Sastry. *Adaptive Control*. Springer, 1998.

[15] G. Antonelli, S. Chiaverini, N. Sarkar, and M. West. Adaptive control of an autonomous underwater vehicle: Experimental results on odin. *IEEE Transactions on Control Systems Technology*, 9(5):756–765, 2001.

[16] V. Godavarthi, K. Krishna, S. L. Brunton, et al. Leveraging three-dimensionality for navigation in bluff-body wakes. *Flow*, 5:E8, 2025.

[17] K. Krishna, S. L. Brunton, and Z. Song. Finite-time lyapunov exponent analysis of policy predictive control and reinforcement learning. *IEEE Access*, 2023.

[18] J. Mei, J. N. Kutz, and S. L. Brunton. Observability-based energy efficient path planning with background flow via deep reinforcement learning. In *2023 IEEE Conference on Decision and Control (CDC)*, pages 4364–4371. IEEE, 2023.

[19] J. Schulman, F. Wolski, P. Dhariwal, et al. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[20] J. Lee, J. Hwangbo, L. Wellhausen, et al. Learning quadrupedal locomotion over challenging terrain. *Science Robotics*, 5(47):eabc5986, 2020.

[21] M. G. Bellemare, S. Srinivasan, G. Ostrovski, et al. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature*, 588(7837):77–82, 2020.

[22] V. Mnih, K. Kavukcuoglu, D. Silver, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[23] M. R. Behrens and W. C. Ruder. Smart magnetic microrobots learn to swim with deep reinforcement learning. *Advanced Intelligent Systems*, 4:2200023, 2022.

[24] G. Reddy, A. Celani, T. J. Sejnowski, et al. Learning to soar in turbulent environments. *Proceedings of the National Academy of Sciences*, 113(33):E4877–E4884, 2016.

[25] G. Reddy, J. Wong-Ng, A. Celani, T. J. Sejnowski, and M. Vergassola. Glider soaring via reinforcement learning in the field. *Nature*, 562:236–239, 2018.

[26] P. Gunnarson, I. Mandralis, G. Novati, et al. Learning efficient navigation in vortical flow fields. *Nature Communications*, 12(1):7143, 2021.

[27] Y. Zhang, S. Zheng, C. Xu, et al. Efficient navigation in vortical flows based on reinforcement learning and flow field prediction. *Ocean Engineering*, 327:120937, 2025.

[28] Y. Liu, O. S. Pak, and A. C. H. Tsang. Learning to navigate in chemical fields without a map at low reynolds numbers. *Advanced Science*, 12(24):e10092, 2025.

[29] S. Colabrese, K. Gustavsson, A. Celani, et al. Smart inertial particles. *Physical Review Fluids*, 3(8):084301, 2018.

[30] A. K. Lidtke, D. Rijpkema, and B. Düz. General reinforcement learning control for auv manoeuvring in turbulent flows. *Ocean Engineering*, 309:118538, 2024.

[31] H. Feng, D. Yuan, J. Miao, et al. Efficient navigation of a robotic fish swimming across the vortical flow field. *Journal of Hydrodynamics*, 36:1118–1129, 2024.

[32] L. Amoudruz and P. Koumoutsakos. Independent control and path planning of microswimmers with a uniform magnetic field. *Advanced Intelligent Systems*, 4(3):2100183, 2022.

[33] K. Fukami and K. Taira. Grasping extreme aerodynamics on a low-dimensional manifold. *Nature Communications*, 14(6480), 2023.

[34] Y. Jiao, H. Hang, J. Merel, et al. Sensing flow gradients is necessary for learning autonomous underwater navigation. *Nature Communications*, 16(1):3044, 2025.

[35] J. v. Uexküll. *Umwelt und Innenwelt der Tiere*. Springer, 1909.

[36] G. Dehnhardt, B. Mauck, W. Hanke, and H. Bleckmann. Hydrodynamic trail-following in harbor seals (phoca vitulina). *Science*, 293(5528):102–104, 2001.

[37] Y. Kim, V. Godavarthi, V. Rolandi, et al. Influence of three-dimensionality on wake synchronisation of an oscillatory cylinder. *Journal of Fluid Mechanics*, 1001:A24, 2024.

[38] J. C. Liao, D. N. Beal, G. V. Lauder, and M. S. Triantafyllou. The kármán gait: novel body kinematics of rainbow trout swimming in a vortex street. *J Exp Biol*, 206(6):1059–1073, 2003.

[39] D. N. Beal, F. S. Hover, M. S. Triantafyllou, J. C. Liao, and G. V. Lauder. Passive propulsion in vortex wakes. *Journal of Fluid Mechanics*, 549:385–402, 2006.

[40] J. D. Eldredge and D. Pisani. Passive locomotion of a simple articulated fish-like system in the wake of an obstacle. *Journal of Fluid Mechanics*, 607:279–288, 2008.

[41] I. Masmitja, M. Martin, T. O'Reilly, B. Kieft, and N. Palomeras. Dynamic robotic tracking of underwater targets using reinforcement learning. *Science Robotics*, 8:eade7811, 2023.

[42] Q. Zhang, Z. Zuo, H. Wang, B. Liu, Y. Yilihamu, and L. Wen. Utact: Underwater vision-based tactile sensor with geometry reconstruction and contact force estimation. *Advanced Robotics Research*, page 202500091, 2026.

[43] B. Chen and P. Lin. Deep imitation learning for optimal trajectory planning and initial condition optimization for an unstable dynamic system. *Advanced Intelligent Systems*, 6:2300379, 2024.