Navigation of a three-link microswimmer via deep reinforcement learning

Yuyang Lai [1], Sina Heydari [2], On Shun Pak [2,3,*] and Yi Man [1,†]
[1]*Department of Mechanics and Engineering Science at College of Engineering, Peking University, Beijing 100871, People's Republic of China*
[2]*Department of Mechanical Engineering, Santa Clara University, Santa Clara, California 95053, USA*
[3]*Department of Applied Mathematics, Santa Clara University, Santa Clara, California 95053, USA*

(Received 6 November 2024; accepted 5 May 2025; published 16 June 2025)

Motile microorganisms develop effective swimming gaits to adapt to complex biological environments. Translating this adaptability to smart microrobots presents significant challenges in motion planning and stroke design. In this work, we explore the use of reinforcement learning (RL) to develop stroke patterns for targeted navigation in a three-link swimmer model at low Reynolds numbers. Specifically, we design two RL-based strategies: one focusing on maximizing velocity (velocity-focused strategy) and another balancing velocity with energy consumption (energy-aware strategy). Our results demonstrate how the use of different reward functions influences the resulting stroke patterns developed via RL, which are compared with those obtained from traditional optimization methods. Furthermore, we showcase the capability of the RL-powered swimmer in adapting its stroke patterns to perform different navigation tasks, including tracing complex trajectories and pursuing moving targets. Taken together, this work highlights the potential of reinforcement learning as a versatile tool for designing efficient and adaptive microswimmers capable of sophisticated maneuvers in complex environments.

DOI: 10.1103/9msg-hgqn

I. INTRODUCTION

Locomotion at low Reynolds numbers is a fascinating subject, as the interaction between microorganisms and their environment generates propulsion in ways fundamentally different from macroscopic motion [1–3]. Microorganisms navigate their viscous environments through specialized mechanisms, such as the undulating flagella of sperm cells [4], the rotating helical flagella of bacteria [5], and the coordinated ciliary movements of paramecia [6]. Inspired by these natural strategies, various microswimmers have been designed for applications such as drug delivery [7–9], self-assembly [10,11], and targeted therapy [12]. A core challenge in the design of microswimmers is the development of effective stroke patterns or motion planning: What body deformations can achieve the desired locomotion? Unlike microorganisms, which can adapt their gaits based on environmental cues and functional needs, most current microswimmers possess a single mode of motion and can only operate in simple, controlled environments [7,13–17]. Addressing this challenge requires not only an understanding of the biomechanics of microbial movement but also insights into how their detailed structures and sensory systems coordinate to achieve their goals, making the modeling process inherently complex [18–20].

Model-free reinforcement learning (RL) offers a promising approach for stroke design and motion planning in microswimmers. Recent computational and experimental studies have demonstrated

*Contact author: opak@scu.edu
†Contact author: yiman@pku.edu.cn

the potential of RL in studying biophysical problems at low Reynolds numbers and designing intelligent microswimmers [21–26]. Within the RL framework, microswimmers learn from experience through trial and error without relying on physical knowledge of the system. This allows for the discovery of novel locomotion strategies that traditional modeling approaches may not easily uncover. For example, RL has enabled microswimmers to achieve targeted navigation, adapting their movements in response to complex environmental cues and disturbances, ensuring robust performance even in dynamic and unpredictable fluid environments [24,27–32]. Studies have shown that microswimmers can optimize their swimming strategies to achieve specific goals, such as maximizing speed or efficiency, by adjusting their stroke patterns accordingly [24–26]. Additionally, RL has been successfully applied to scenarios involving multiple microswimmers, facilitating coordinated behaviors such as pursuit-evasion dynamics and collective navigation, which are critical for applications like targeted drug delivery and environmental sensing [25,33]. These advancements demonstrate how reinforcement learning can effectively address the challenges associated with microswimmer design, offering a powerful tool for developing efficient and intelligent microrobots capable of performing sophisticated tasks in complex biological environments [30,34,35].

In this work, we consider a three-link swimmer, one of the simplest microswimmer models capable of generating propulsion at low Reynolds numbers. We utilize RL to explore the development of stroke patterns for targeted navigation. We design two strategies—one focusing on maximizing velocity (velocity-focused strategy) and another balancing velocity with energy consumption (energy-aware strategy). We examine the stroke patterns developed through RL based on different reward functions. Our results underscore the effectiveness and versatility of RL in developing stroke patterns to meet various performance goals, demonstrating the potential for RL as a tool to design locomotory gaits of microswimmers. We also showcase the capability of the RL-powered microswimmer in performing complex navigation tasks in scenarios relevant to its potential biomedical applications.

This paper is structured as follows. In Sec. II, we introduce the three-link swimmer model, detailing its degrees of freedom and its dynamics at low Reynolds numbers. Section III describes the RL framework we employed, including the design of the two strategies: the velocity-focused strategy and the energy-aware strategy. We outline the neural network architecture and the reward functions tailored for each strategy. In Sec. IV, we present the results of our simulations, analyzing the swimmer's performance under both strategies. We compare the stroke patterns developed through RL with those from previous studies, highlighting similarities and differences. Additionally, we demonstrate the RL framework's capability to develop complex stroke patterns for tracing a star-shaped trajectory and navigating toward moving targets. We conclude this work with remarks on its limitations in Sec. V.

II. MODEL OF A THREE-LINK SWIMMER

The three-link swimmer possesses the minimal degrees of freedom required for self-propulsion in a low-Reynolds-number environment [36]. This system consists of three identical rigid links, each with a radius a and a length $l = L/3$, where L represents the total length of the swimmer [see Fig. 1(a)]. The locomotion of the swimmer is constrained to two dimensions, described using the Cartesian coordinates $(\mathbf{e}_1, \mathbf{e}_2)$. The left end of each link is denoted by $\mathbf{x}_i = (x_i, y_i)$, and its orientation by $\mathbf{t}_i$. The angle between $\mathbf{t}_i$ and $\mathbf{e}_1$ is represented by θ_i. The swimmer's hinges allow for free rotation, with the angles between adjacent links denoted as α_1 and α_2. By actuating these angles, the links interact with the surrounding fluid, resulting in net propulsion. To avoid close proximity, the angles α_1 and α_2 are restricted to the range $[-2\pi/3, 2\pi/3]$.

The position of any point on link i is denoted by $\mathbf{X}_i = \mathbf{x}_i + s\mathbf{t}_i$, where s represents the distance along the link from its left end. The local velocity at $\mathbf{X}_i$ is given by

$$\mathbf{u}_i = \dot{\mathbf{x}}_i + s\dot{\theta}_i\mathbf{n}_i, \tag{1}$$

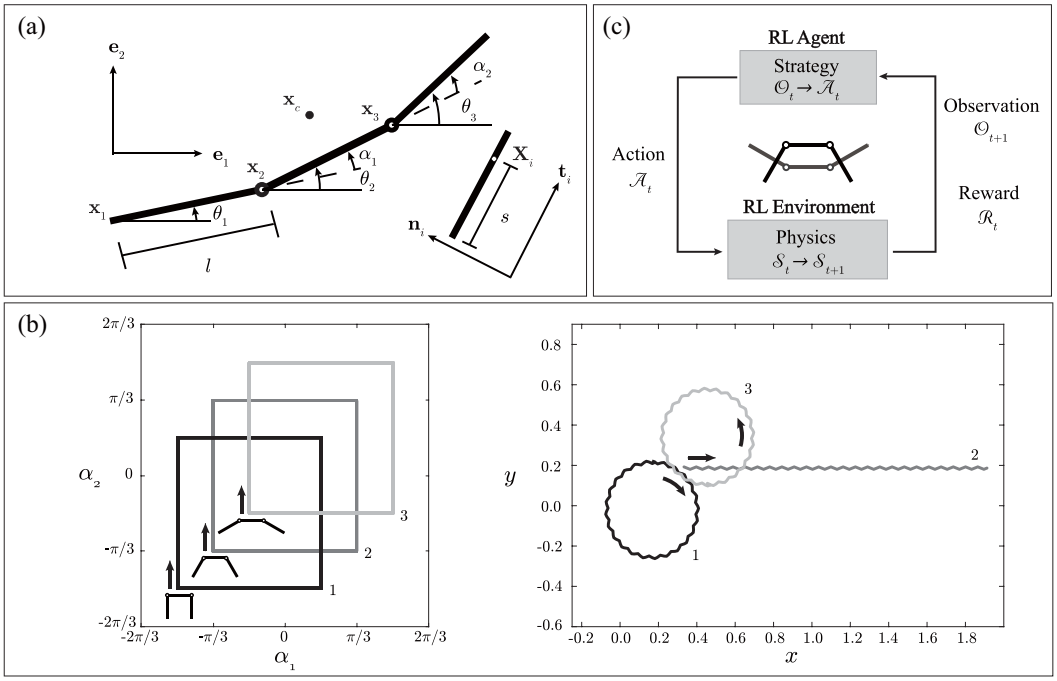

FIG. 1. (a) Model of three-link swimmer. It consists of three rigid links of equal length, which are connected by two hinges, allowing rotation to adjust the relative angles α_i ($i = 1, 2$). The swimmer's geometric centroid, denoted $\mathbf{x}_c$, serves as the reference point for its motion. (b) Three basic stroke patterns of the three-link swimmer. Left: stroke patterns in phase space; right: corresponding trajectories of the geometric centroid in physical space. The initial configurations for these movements are shown in the left panel. (c) Framework of model-free reinforcement learning.

where $\mathbf{n}_i$ represents the unit vector normal to link i. Based on the resistive force theory, the local hydrodynamic force is proportional to the local velocity [37]. Consequently, the local force is calculated as follows:

$$\mathbf{f}_i = -(C_\parallel \mathbf{t}_i\mathbf{t}_i + C_\perp \mathbf{n}_i\mathbf{n}_i) \cdot \mathbf{u}_i, \tag{2}$$

where $C_\parallel = 2\pi\mu/[\ln(L/a) - 1/2]$ and $C_\perp = 4\pi\mu/[\ln(L/a) + 1/2]$ are the drag coefficients [37], and μ is the dynamic viscosity of the fluid. Integrating along link i, the total force and hydrodynamic torque are

$$\mathbf{F}_i = \int_0^l \mathbf{f}_i \, \mathrm{d}s, \quad \mathbf{M}_i = \int_0^l \mathbf{X}_i \times \mathbf{f}_i \mathrm{d}s. \tag{3}$$

For low-Reynolds-number locomotion, the total hydrodynamic force and torque on the swimmer should vanish, namely

$$\sum_{i=1}^{3} \mathbf{F}_i = \mathbf{0}, \quad \sum_{i=1}^{3} \mathbf{M}_i = \mathbf{0}. \tag{4}$$

Moreover, the motion of the swimmer has kinematic constraints (here, $i = 1, 2$):

$$\mathbf{x}_{i+1} - \mathbf{x}_i = l\mathbf{t}_i, \quad \theta_{i+1} - \theta_i = \alpha_i. \tag{5}$$

In presenting our results, we scale all lengths by the total length of the swimmer, L. We assume a characteristic time scale, T_0, which corresponds to the actuation rate of the angle between

neighboring links. The associated force scale is defined as $C_\perp L^2/T_0$. As a result, the dimensionless quantities are defined as $\mathbf{x}_i = L\bar{\mathbf{x}}_i$, $\overline{\alpha_j} = T_0\dot{\alpha}_j$, $\gamma = C_\parallel/C_\perp$, where $i = 1, 2, 3$ and $j = 1, 2$. In this study, we consider a slender swimmer ($a \ll L$) with $\gamma = 1/2$. To simplify the notations, we omit the overbars hereafter and refer only to dimensionless quantities. Combining Eqs. (4) and (5), the swimmer's motion is described by a system of linear equations:

$$\boldsymbol{H}(\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{\Theta}) \begin{pmatrix} \dot{\boldsymbol{X}} \\ \dot{\boldsymbol{Y}} \\ \dot{\boldsymbol{\Theta}} \end{pmatrix} = \boldsymbol{q}, \tag{6}$$

where $\boldsymbol{X} = [x_1, x_2, x_3]^\top$, $\boldsymbol{Y} = [y_1, y_2, y_3]^\top$, $\boldsymbol{\Theta} = [\theta_1, \theta_2, \theta_3]^\top$, while $(\dot{\boldsymbol{X}}, \dot{\boldsymbol{Y}}, \dot{\boldsymbol{\Theta}})$ are their derivative with respect to time t. The vector $\boldsymbol{q}$ is the function of the actuation rates of the angle between neighboring links $\dot{\alpha}_1, \dot{\alpha}_2$. See Supplemental Material [38] for the components of $\boldsymbol{H}$ and $\boldsymbol{q}$.

All instantaneous configurations of the swimmer can be represented by a point in the two-dimensional (α_1, α_2) phase space. Thus, all periodic stroke patterns of the swimmer can be depicted as a single closed curve in this space. In Fig. 1(b), we illustrate three stroke patterns in the phase space (left panel) along with the corresponding trajectories of the swimmer's geometric centroid in the physical space (right panel). The classical Purcell's stroke pattern is shown in gray lines. In this pattern, only one arm moves at a time, maintaining symmetry with joint angles ranging from $-\pi/3$ to $\pi/3$. This symmetric stroke results in the swimmer moving straight along the horizontal direction. We modify Purcell's stroke pattern by allowing the joint angles to vary asymmetrically between $-\pi/2$ and $\pi/6$, as illustrated by the light gray lines. This asymmetry causes the swimmer to move along a clockwise circular trajectory. Similarly, if the joint angles vary between $-\pi/6$ and $\pi/2$, shown by the black lines, the swimmer moves along a counterclockwise circular path.

III. TARGETED NAVIGATION VIA REINFORCEMENT LEARNING

A. RL framework for targeted navigation

We use an RL framework to train the swimmer in swimming parallel along a certain target direction θ_T [Fig. 1(c)]. The state of the system, $\mathcal{S} \in (\mathbf{x}_1, \theta_1, \theta_2, \theta_3)$, is specified by the coordinate of the swimmer's one end $\boldsymbol{x}_1$ and link orientations $\theta_1, \theta_2, \theta_3$. The observation, $\mathcal{O} \in (\cos\theta_d, \sin\theta_d, \alpha_1, \alpha_2)$, is extracted from the state, where $\theta_d = \theta_2 - \theta_T$ is the difference between the second link's orientation and the target direction. The term $(\cos\theta_d, \sin\theta_d)$ is introduced to ensure continuity in the orientation space, as each component remains within $[-1, 1]$. This ensures that our data will not overflow, thereby preventing the continuity of the values from being disrupted when taking θ_d modulo 2π. The agent in the RL framework utilizes an Actor-Critic neural network architecture to decide the swimmer's actions based on the observations. Specifically, for each action step, the swimmer senses its observation $\mathcal{O}$ and, through the Actor network, determines the action $\mathcal{A} \in (\dot{\alpha}_1, \dot{\alpha}_2)$ by calculating the angular velocities for rotating its two hinges.

We design different reward functions to evaluate the success of the swimmer's actions in achieving targeted navigation. Two types of objective criteria are established for control. The first objective focuses on velocity toward the target direction, which we refer to as the VFS. The criterion here is the distance traveled by the swimmer along the target direction within a specific time period. Specifically, the reward function for the VFS is defined as follows:

$$\mathcal{R}_k = b\left(\mathbf{x}_{c_{k+1}} - \mathbf{x}_{c_k}\right) \cdot \mathbf{p}, \tag{7}$$

where k represents the ordinal number of the training step, and $\mathbf{x}_{c_k}$ denotes the geometric centroid of the swimmer at the kth training step. The targeted orientation is denoted as $\mathbf{p} = \cos\theta_T \mathbf{e}_1 + \sin\theta_T \mathbf{e}_2$. The parameter b is a positive scaling factor introduced to adjust the magnitude of the reward signal. A larger value of b increases the reward's magnitude, which can accelerate the convergence rate of training by encouraging larger updates during gradient descent. However, if b is set too high, it may

lead to numerical instability due to excessively large gradient steps. Therefore, b should be chosen carefully to balance the trade-off between faster convergence and stable learning dynamics.

The second objective is to achieve an EAS, which aims to realize targeted navigation while penalizing energy consumption. We consider the total rate of work done by the swimmer on the fluid:

$$\Phi = \sum_{i=1}^{3} \Phi_i, \tag{8}$$

where Φ_i refers to the rate of work done by the ith link and can be computed as follows:

$$\Phi_i = \int_0^{1/3} -\mathbf{f}_i \cdot \mathbf{u}_i \, \mathrm{d}s, \tag{9}$$

where $\mathbf{f}_i$ and $\mathbf{u}_i$ are given by Eqs. (1) and (2).

In the actual training process, we calculate the work done by the swimmer during the kth training step, defining it as

$$W_k = \int_{t_k}^{t_{k+1}} \Phi \, \mathrm{d}t, \tag{10}$$

where $\mathbf{t}_k$ denotes the initial time of the kth training step. By incorporating an energy penalty, we design the reward function for the EAS as

$$\mathcal{R}_k = b\big(\mathbf{x}_{c_{k+1}} - \mathbf{x}_{c_k}\big) \cdot \mathbf{p} - cW_k, \tag{11}$$

where c is a positive weight introduced to penalize mechanical power consumption during each action step. A larger c increases the emphasis on reducing energy expenditure, which can lead to higher swimming efficiency. However, if c is set too high, the swimmer may prioritize conserving energy over progressing toward the target, resulting in decreased accuracy in navigating along the desired direction or even failure to reach the target.

B. Training process

We employ the proximal policy optimization (PPO) algorithm to train the swimmer to navigate along a specified target direction, θ_T. The algorithm is adapted from [22,24] (see Supplemental Material [38] for more details). Without loss of generality, we set the target direction to be parallel to the x-direction, corresponding to a target angle of $\theta_T = 0$. To fully explore the observation space $\mathcal{O}$, $(\theta_1, \theta_2, \theta_3)$ in the initial swimmer state $\mathcal{S}$ are randomized at the beginning of each episode. The training process is divided into N_E episodes, each consisting of N_s action steps. A sufficiently large number of episodes and action steps is necessary to ensure the convergence of the training results and smoothness of the swimmer's movements. In the reward functions, we set the coefficients to $b = 6$ and $c = 3$. Our numerical experiments show that choosing a value of $b < 6$ increases the convergence time, though the training results remain similar to when $b = 6$. However, increasing b beyond 6 may cause numerical instability due to larger gradient steps, resulting in deviations from the target direction during navigation. The coefficient c is a positive weight that penalizes mechanical power consumption, which may be expected to reduce performance in terms of displacement toward the target direction or the ability to reorient toward it. When c exceeds 3, we observe a significant asymmetry in the stroke patterns, rendering the navigation strategy ineffective. This occurs because the swimmer prioritizes energy conservation over advancing toward the target, leading to a decrease in navigation accuracy (refer to the Supplemental Material [38] for more details on the effects of these parameters).

In Fig. 2, we compare the progression of rewards versus the number of training episodes for both the VFS and the EAS reward functions. Here, the reward $\mathcal{R} = \sum_{k=1}^{N_s} \mathcal{R}_k$ denotes the cumulative reward obtained over all action steps within a single episode. It can be observed that while both

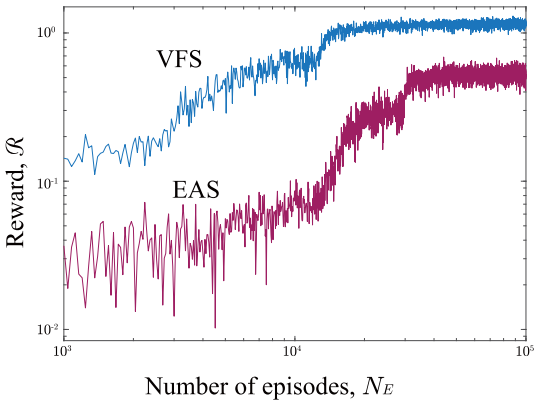

FIG. 2. Convergence of reward functions for the velocity-focused strategy (VFS, blue line) and the energy-aware strategy (EAS, purple line). Each training episode contains a fixed number of action steps $N_s = 200$. The reward $\mathcal{R} = \sum_{k=1}^{N_s} \mathcal{R}_k$ denotes the cumulative reward obtained over all action steps within a single episode.

training processes eventually converge, the EAS requires more episodes to do so. Specifically, the VFS rewards converge around 15 000 episodes, whereas the EAS rewards take approximately 40 000 episodes to converge. This slower convergence in the EAS can be attributed to the added complexity of its reward function, which incorporates not only the displacement in the target direction but also an energy penalty. We set a sufficiently large number of episodes ($N_E = 100\,000$) to ensure convergence of the reward function while maintaining a manageable training time. Similarly, a sufficiently large number of action steps per episode ($N_s = 200$) is set to yield a high success rate of navigation while keeping training time minimal (see Supplemental Material [38] for more details on the effect of N_s on the success rate.)

IV. RESULTS AND DISCUSSION

A. Stroke patterns and motion dynamics

In Fig. 3, we illustrate the swimming trajectories based on the VFS and EAS. The initial configuration of the swimmer is set as $\mathbf{x}_1 = (1, 0)$ and $\theta_1 = \theta_2 = \theta_3 = \pi/3$. The swimmer, following both strategies, is allowed to move for 1500 steps. The trajectories of the stroke patterns in the phase plane are shown in Figs. 3(a) and 3(c), while the corresponding trajectories of the geometric centroid of the swimmer in the physical space are shown in Figs. 3(b) and 3(d).

We observe that in both cases, the swimmer successfully achieves targeted navigation and swims horizontally. The trajectories can be divided into two stages: steering and translation. The steering is the process of the swimmer adjusting its direction, while the translation reflects the swimmer moving steadily along a given direction. In the phase space shown in Figs. 3(a) and 3(c), the phase point circles clockwise, with the swirling center gradually approaching the origin of the phase plane. The stroke pattern eventually converges to a symmetric closed loop, indicating straight locomotion. In the physical space shown in Figs. 3(b) and 3(d), the swimmer gradually turns clockwise and ultimately swims horizontally. The transient process represents the steering stage, while the converged straight motion represents the translation stage. The converged stroke patterns of VFS and EAS are visibly different. The VFS trajectory in the phase space is more rectangular, while the EAS trajectory is more rounded.

Since the position of the swimmer's centroid oscillates during motion, we need to define an averaged orientation to establish a criterion for convergence. Observing that one period of the swimmer's motion contains about 70 steps, we choose to smooth the trajectories by averaging over this period. Specifically, for each step i (where $i > 35$), we calculate the average position by

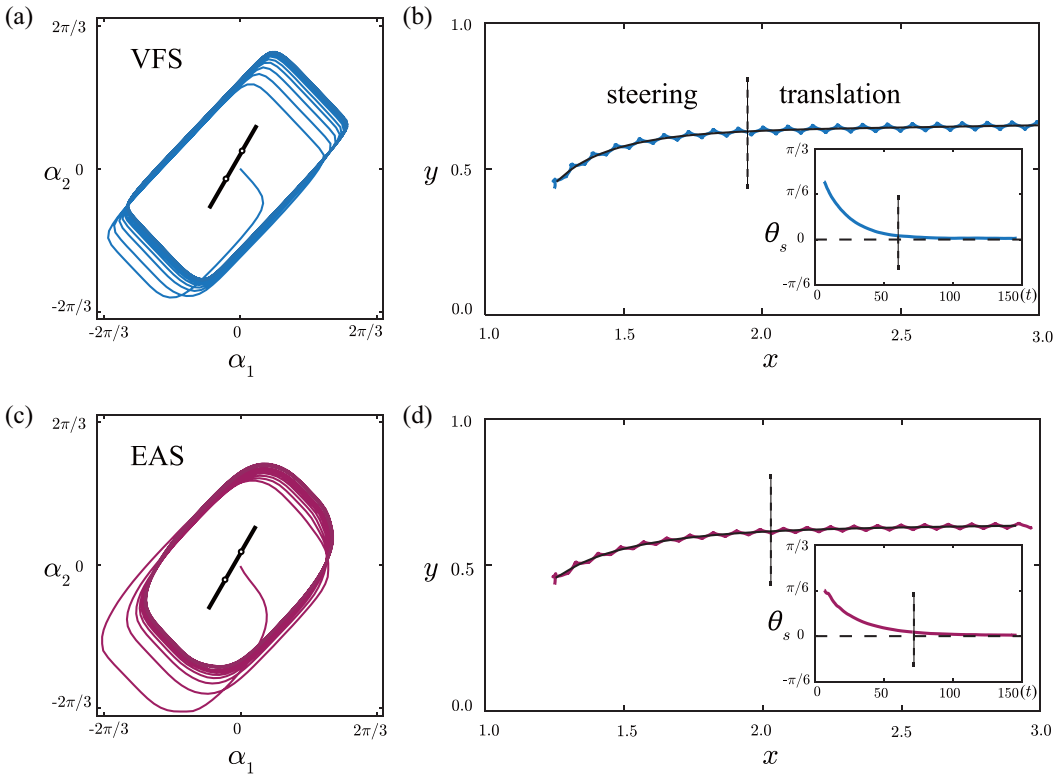

FIG. 3. Learning to swim along $\theta_T = 0$ with VFS and EAS. (a),(c): Stroke patterns in the phase plane. (b),(d): Trajectories of the geometric centroid and the smoothed path. The initial state is set as $\mathbf{x}_1 = (1, 0)$, $\theta_1 = \theta_2 = \theta_3 = \pi/3$. The insets in (b),(d) display the evolution of the swimmer's averaged orientation, θ_s, over time. In (b),(d), the black lines represent the smoothed path of the swimmer's motion, with the black dashed line used to distinguish the steering and translation stages. The blue lines indicate the VFS results, while the purple lines show the EAS results.

considering the positions from 35 steps before to 35 steps after step i. This means we average the positions from step $i - 35$ to step $i + 35$, effectively smoothing over one full period of motion. The smoothed path is shown as the black solid lines in Figs. 3(b) and 3(d). Next, we calculate the slope angle θ_s of the smoothed path to determine the averaged orientation of the swimmer. To do this, we compute the finite differences between consecutive smoothed positions to obtain the local slope at each point. By analyzing θ_s, we can assess how effectively the swimmer is aligning its motion with the desired target direction, thereby establishing a criterion for convergence. In the insets of Figs. 3(b) and 3(d), we show the convergence of the averaged orientation, θ_s. In addition, we use θ_s to precisely distinguish between the steering and translation stages. If $|\theta_s| > 2.5^\circ$, the trajectory segment is classified as steering; otherwise, it is classified as translation. Based on this classification, we use a dashed line to separate the two stages.

B. Swimming speed and efficiency

Building upon the results that demonstrate both strategies effectively achieve targeted navigation, we proceed to quantitatively distinguish the VFS and EAS. By calculating the swimming speed along the target direction and the swimming efficiency during the steering and translation stages—based on the smoothed motion—we quantify the differences between the two strategies.

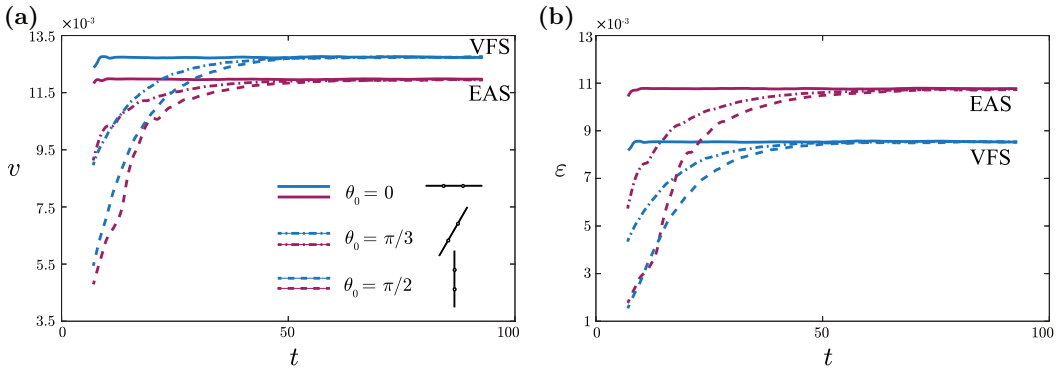

FIG. 4. Velocity along the target direction (a) and swimming efficiency (b) over time. Velocity and efficiency are calculated based on the smoothed paths. In each case, straight initial configurations with $\theta_2 = 0$ (solid line), $\pi/3$ (dash-dotted line), and $\pi/2$ (dashed line) are considered.

In Fig. 4(a), we demonstrate that the translation stage is independent of the initial configurations. We simulate the dynamics resulting from the VFS and EAS with initial configurations $\theta_2 = \theta_0 = 0, \pi/3, \pi/2$ and $\alpha_1 = \alpha_2 = 0$. In both VFS and EAS, the horizontal speed, denoted v, converges to the same value. For the VFS, the horizontal speed converges to approximately 0.01284, while for the EAS, the steady speed is slightly slower at about 0.01176.

To evaluate the swimming efficiency, we adopt a definition by Lighthill [37] and Purcell [39]. At a given time, we calculate the rate of work done by the swimmer on the fluid, denoted as $\Phi(t)$, using Eq. (8). As a reference motion, we consider towing the swimmer in its straightened configuration ($\alpha_1 = \alpha_2 = 0,\ \theta_2 = 0$) along the horizontal direction at velocity $v(t)$. The rate of work for the towing problem is calculated as

$$\Phi_0 = \gamma v^2. \tag{12}$$

The swimming efficiency, ε, is then defined as the ratio of the rates of work:

$$\varepsilon = \frac{\Phi_0}{\Phi}. \tag{13}$$

By calculating the swimming efficiency, we observe that in both VFS and EAS, the efficiency criterion ε converges to consistent values despite different initial configurations. According to Fig. 4(b), for the VFS, the efficiency converges to approximately 0.854%, while for the EAS, a higher efficiency of about 1.077% is achieved. These results demonstrate that, regardless of the initial configuration, both strategies converge to their respective steady efficiency levels.

During the translation stage, the swimmer begins to move steadily by repeating the same stroke pattern. In Figs. 5(a) and 5(b), we plot the converged stroke patterns for both VFS and EAS in the phase space using solid lines. Tam *et al.* [40] investigated the optimal stroke patterns for the three-link swimmer, focusing on two cases: velocity optimal (VO) and efficiency optimal (EO). They numerically optimized the periodic functions of α_1 and α_2 using gradient search. In Figs. 5(a) and 5(b), we reproduce the stroke patterns of VO and EO in [40] using dashed lines. We compare these optimal stroke patterns with those obtained through our reinforcement learning approach. It is intriguing to observe that, although the RL-generated stroke patterns are not identical to the optimized patterns from [40], they exhibit similar features. For instance, the stroke pattern from the VFS shares similarities with the VO pattern, being more rectangular in the phase space. Meanwhile, the stroke pattern from the EAS resembles the EO pattern, appearing more rounded.

In Figs. 5(c) and 5(d), we further quantitatively compare the results from our RL strategies with those from [40] by calculating the average velocity along the target direction $\langle v \rangle_t$ and the swimming efficiency over one stroke cycle $\langle \varepsilon \rangle_t$. Specifically, we consider four cases: the velocity-optimal (VO)

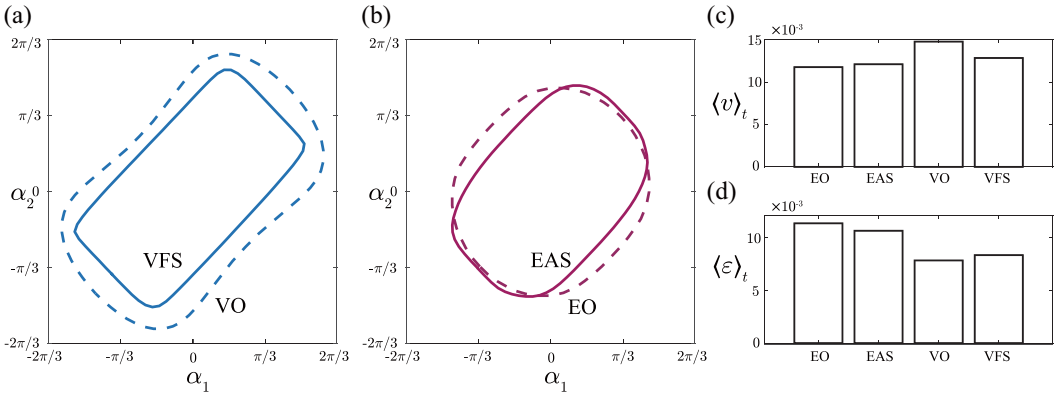

FIG. 5. Comparison between the models from optimizations and the strategies obtained through RL. (a): Stroke patterns with VO (velocity optimal) and VFS (velocity-focused strategy). (b): Stroke patterns with EO (efficiency optimal) and EAS (energy-aware strategy). (c): Comparison of average velocity along the target direction for all four strategies. (d): Comparison of average swimming efficiency for all four strategies. The results of VO and EO are reproduced from Ref. [40].

and the efficiency-optimal (EO) stroke patterns from optimizations, the velocity-focused strategy (VFS), and the energy-aware strategy (EAS) from RL.

For the average velocity, the VO achieves the highest value, followed by the VFS, EAS, and EO. Quantitatively, the VFS achieves over 80% of the average velocity of the VO, indicating that the RL-generated VFS closely approximates the velocity performance of the optimal stroke pattern. In terms of swimming efficiency, the EO from optimization attains the highest efficiency, followed by the EAS, VFS, and VO. Notably, the EAS again captures over 80% of the efficiency achieved by the EO, which demonstrates that the RL-generated EAS effectively balances energy consumption while maintaining reasonable propulsion.

These results highlight that, although the stroke patterns obtained through RL are not identical to the optimal ones, they exhibit similar features and achieve comparable performance levels. This underscores the capability of RL in developing effective stroke patterns that align with specific objectives, such as maximizing velocity or efficiency, without explicitly programming these optimal solutions. Overall, the RL approach demonstrates a strong ability to capture key characteristics of optimal swimming gaits identified by traditional optimization methods.

C. Complex navigation tasks

Finally, we showcase the swimmer's capability to trace complex paths and navigate toward moving targets. In Fig. 6, we task the swimmer with tracing a star-shaped trajectory. Notably, the hydrodynamic calculations required to design the stroke patterns for such complex paths can become intractable as the complexity increases. Here, rather than explicitly programming the swimmer's stroke patterns, we only select target points ($\mathbf{x}_{T_i}, i = 1, 2, \ldots, 10$) as landmarks and require the swimmer to navigate using its own strategy. The target direction at time step $k+1$ is given by $\theta_{T_{k+1}} = \arg(\mathbf{x}_{T_i} - \mathbf{x}_{c_k})$. Starting from the initial state $\mathbf{x}_1 = (1, 0)$ with link orientations $\theta_1 = \theta_2 = \theta_3 = 0$, the swimmer, equipped with the VFS model, is assigned the next target point $\mathbf{x}_{T_{i+1}}$ once its centroid is within a certain threshold (set to 0.001 here) from $\mathbf{x}_{T_i}$. The navigation strategy enables the swimmer to adjust its swimming gaits to navigate several wide (e.g., around point 3) and sharp angles (e.g., around point 4) in tracing the star-shaped trajectory (see Supplemental Movie 1 [38]).

Next, we demonstrate the RL-powered swimmer's capability to navigate toward a dynamic target, characterized by its position $\mathbf{x}_T$, orientation $\mathbf{p}_T$, and intrinsic speed v_T. In addition, we consider

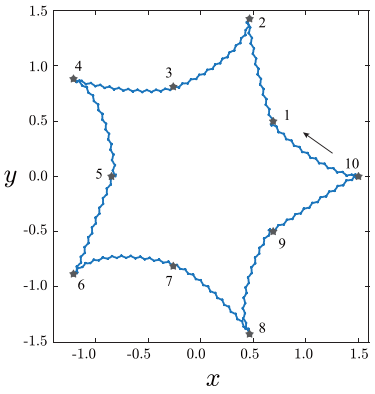

FIG. 6. Three-link swimmer traces a star-shaped trajectory. The trajectory of the swimmer's geometric centroid is represented by blue lines. Initialized in a straight configuration with $\theta_1 = \theta_2 = \theta_3 = 0$, the swimmer is provided with a sequence of target points (1–10), where it chases one target point (gray stars) at a time. The black arrows indicate the intended direction of the swimmer's movement.

scenarios where the target's movement is influenced by random fluctuations due to Brownian motion, characterized by a diffusivity D. This target undergoes purely translational diffusion in two dimensions, described by independent Brownian motions in the x- and y-directions. Specifically, each action step satisfies $\langle \delta x^2 \rangle = \langle \delta y^2 \rangle = 2D\delta t$, where δt denotes the duration of an action step, and δx and δy are the displacements in the x- and y-directions within one action step. This combination of directed movement and random motion of the target introduces additional complexity to the swimmer's navigation task. We note that all these quantities—position, orientation, speed, and diffusivity—are nondimensionalized using the characteristic length, time, and force scales defined earlier in Sec. II.

In Fig. 7, we present three scenarios where the swimmer navigates toward moving targets with different intrinsic speeds. The swimmer utilizes the VFS to adjust its stroke patterns based on the current observed direction of the moving target relative to its own position. Navigation is

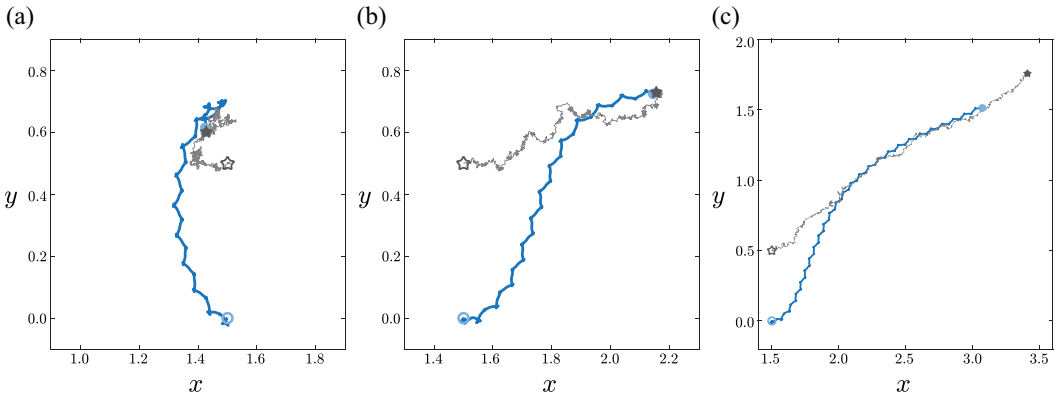

FIG. 7. Three-link swimmer (VFS, blue dots) navigates toward diffusing targets (gray stars) with different speeds. (a): $v_T/v_m = 0$. (b): $v_T/v_m = 0.5$. (c): $v_T/v_m = 1$; v_T is the target's speed and v_m is the maximum speed achieved by the VFS. The trajectory of the swimmer's geometric centroid is shown in blue lines, and the trajectory of the target is shown in gray lines. The swimmer is initialized as a straight shape with $\theta_1 = \theta_2 = \theta_3 = 0$, and the target is oriented at 30° relative to the horizontal axis. The diffusivity is set to $D = 5 \times 10^{-5}$.

achieved by continuously sensing the target's location and adapting its movements to minimize the distance between the swimmer and the target. In the simulations, the swimmer's initial state is set as $\mathbf{x}_1 = (1, 0)$ with link orientations $\theta_1 = \theta_2 = \theta_3 = 0$. The target starts from the initial position $\mathbf{x}_T = (1.5, 0.5)$ and has an initial orientation of 30° relative to the horizontal axis. The diffusivity of the target is set to $D = 5 \times 10^{-5}$. We consider targets with three different intrinsic speeds: $v_T/v_m = 0$, 0.5, and 1, where v_m denotes the maximum speed achieved by the VFS. We define capture as the event when the distance between the swimmer and the target becomes less than a predefined threshold of 0.001. Once the swimmer is within this distance of the target, it is considered to have successfully captured the target.

In Fig. 7(a), we present the scenario where the swimmer (represented by the blue dot) navigates toward the target (gray star) undergoing pure Brownian motion (i.e., $v_T = 0$). Despite the target's random motion, the swimmer effectively adjusts its motion based on the observed direction of the target relative to its own position and navigates toward the moving target. We observe that the swimmer's centroid follows a relatively smooth trajectory compared with the randomly fluctuating path of the target. The swimmer eventually captures the moving target (see Supplemental Movie 2 [38]). When the target has an intrinsic speed that is half that of the swimmer (i.e., $v_T = v_m/2$) in addition to its random motion, the swimmer is still able to continuously adapt its stroke patterns to pursue and successfully capture the moving target (see Supplemental Movie 3 [38]). In Fig. 7(c), we push the limits further by examining the scenario where the moving target's intrinsic speed is increased to match that of the swimmer (i.e., $v_T = v_m$). Under this challenging condition, the swimmer is unable to capture the target but is still able to closely follow its trajectory (see Supplemental Movie 4 [38]). Taken together, these results demonstrate the capability of the RL-powered swimmer to navigate toward a target moving at a significant fraction of its own speed.

V. CONCLUDING REMARKS

In this work, we presented a reinforcement learning (RL) approach to enable the navigation of a three-link swimmer at low Reynolds numbers. While a prior study demonstrated limited locomotion of a three-link swimmer with discrete action spaces [26], the deep RL-powered swimmer presented here leverages continuous action spaces to learn complex stroke patterns for effective swimming and navigation toward a target direction. We examined how different reward functions—one that rewards only the swimmer's velocity toward the target and another that also accounts for energy consumption—lead to the development of distinct stroke patterns. We note that energetic cost has been incorporated into the reward function in previous predator-prey contexts [25]. In contrast, our work focuses on the propulsion performance of a widely studied three-link swimmer, enabling direct benchmarking of RL-derived strategies against those obtained from prior optimization-based approaches. With different reward functions, we observed that the RL-derived stroke patterns exhibit qualitative features similar to the optimal solutions identified in previous optimization studies [40]. Quantitatively, the strategies developed by RL are at least 80% as effective as the optimal solutions in terms of both propulsion velocity and energetic efficiency. The performance gap may be attributed to the fundamental difference in methodology: prior optimization-based approaches typically impose a single-period optimization of stroke kinematics and explicitly search for an optimal periodic gait under well-defined parameters, the RL framework applies no such constraints *a priori*. Instead, the RL agent is free to discover any control strategy that achieves forward motion, without assuming periodicity or a fixed stroke duration. While additional stroke constraints may be incorporated into the RL framework, such approaches inherently impose a preferred structure on the solution. In contrast, the simpler reward functions used here allow the RL agent to more autonomously develop its own strategies, providing a flexible alternative for complex scenarios where effective stroke patterns are not well understood or where the setup may be dynamically changing. Lastly, we demonstrated the swimmer's ability to autonomously adapt its stroke patterns to navigate in any target direction, enabling it to trace complex trajectories and pursue moving

targets (e.g., mimicking swimming bacteria or circulating tumor cells). These capabilities serve as proof of concept for scenarios relevant to potential biomedical applications.

We remark on several limitations of the current study and discuss potential directions for future research. First, we use a three-link swimmer here as a simple example to demonstrate the RL approach. We anticipate that increasing the degrees of freedom by incorporating additional links will enable a multilink swimmer to perform more complex maneuvers and further enhance propulsion performance. Second, in demonstrating the swimmer's ability to pursue a moving target, we neglect the hydrodynamic interactions between the swimmer and the target. Incorporating these interactions in future work could reveal new features in the strategies identified by RL. Lastly, the presence of obstacles or flow perturbations in complex biological environments would also impact the swimmer's navigation. Addressing these challenges would pave the way for developing intelligent microswimmers with more robust navigation capabilities.

ACKNOWLEDGMENTS

Y. Lai acknowledges partial support from the National Natural Science Foundation of China (NSFC) through the Fundamental Research Project for Undergraduates (Grant No. 123B1034). O. S. Pak acknowledges partial support from the National Science Foundation (NSF) under Grants No. CBET-2323046 and No. CBET-2419945. Y. Man acknowledges partial support from the NSFC under Grant No. 12372258.

[1] E. Lauga and T. R. Powers, The hydrodynamics of swimming microorganisms, Rep. Prog. Phys. **72**, 096601 (2009).

[2] J. Elgeti, R. G. Winkler, and G. Gompper, Physics of microswimmers—single particle motion and collective behavior: a review, Rep. Prog. Phys. **78**, 056601 (2015).

[3] J. M. Yeomans, D. O. Pushkin, and H. Shum, An introduction to the hydrodynamics of swimming microorganisms, Eur. Phys. J.: Spec. Top. **223**, 1771 (2014).

[4] L. J. Fauci and R. Dillon, Biofluidmechanics of reproduction, Annu. Rev. Fluid Mech. **38**, 371 (2006).

[5] E. Lauga, Bacterial hydrodynamics, Annu. Rev. Fluid Mech. **48**, 105 (2016).

[6] B. Párducz, Ciliary movement and coordination in ciliates, Int. Rev. Cytol. **21**, 91 (1967).

[7] W. Gao, D. Kagan, O. S. Pak, C. Clawson, S. Campuzano, E. Chuluun-Erdene, E. Shipton, E. E. Fullerton, L. Zhang, E. Lauga *et al.*, Cargo-towing fuel-free magnetic nanoswimmers for targeted drug delivery, Small **8**, 460 (2012).

[8] H. Ceylan, I. C. Yasa, O. Yasa, A. F. Tabak, J. Giltinan, and M. Sitti, 3D-printed biodegradable microswimmer for theranostic cargo delivery and release, ACS Nano **13**, 3353 (2019).

[9] L. Zhang, J. J. Abbott, L. Dong, K. E. Peyer, B. E. Kratochvil, H. Zhang, C. Bergeles, and B. J. Nelson, Characterizing the swimming properties of artificial bacterial flagella, Nano Lett. **9**, 3663 (2009).

[10] G. Grosjean, G. Lagubeau, A. Darras, M. Hubert, G. Lumay, and N. Vandewalle, Remote control of self-assembled microswimmers, Sci. Rep. **5**, 16035 (2015).

[11] U. K. Cheang and M. J. Kim, Self-assembly of robotic micro- and nanoswimmers using magnetic nanoparticles, J. Nanopart. Res. **17**, 145 (2015).

[12] T.-Y. Huang, M. S. Sakar, A. Mao, A. J. Petruska, F. Qiu, X.-B. Chen, S. Kennedy, D. Mooney, and B. J. Nelson, 3D printed microtransporters: Compound micromachines for spatiotemporally controlled delivery of therapeutic agents, Adv. Mater. (Deerfield Beach, Fla.) **27**, 6644 (2015).

[13] W. Hu, G. Z. Lum, M. Mastrangeli, and M. Sitti, Small-scale soft-bodied robot with multimodal locomotion, Nature (London) **554**, 81 (2018).

[14] C. Ohm, M. Brehmer, and R. Zentel, Liquid crystalline elastomers as actuators and sensors, Adv. Mater. **22**, 3366 (2010).

[15] B. Dai, J. Wang, Z. Xiong, X. Zhan, W. Dai, C.-C. Li, S.-P. Feng, and J. Tang, Programmable artificial phototactic microswimmer, Nat. Nanotechnol. **11**, 1087 (2016).
[16] S. Palagi, Soft microrobots based on photoresponsive materials, Mechanically Responsive Materials Soft Robotics 327 (2020).
[17] A. von Rohr, S. Trimpe, A. Marco, P. Fischer, and S. Palagi, Gait learning for soft microrobots controlled by light fields, in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (IEEE, 2018), pp. 6199–6206.
[18] X. Nassif, S. Bourdoulous, E. Eugène, and P.-O. Couraud, How do extracellular pathogens cross the blood–brain barrier? Trends Microbiol. **10**, 227 (2002).
[19] J. P. Celli, B. S. Turner, N. H. Afdhal, S. Keates, I. Ghiran, C. P. Kelly, R. H. Ewoldt, G. H. McKinley, P. So, S. Erramilli *et al.*, Helicobacter pylori moves through mucus by reducing mucin viscoelasticity, Proc. Natl. Acad. Sci. **106**, 14321 (2009).
[20] S. A. Mirbagheri and H. C. Fu, Helicobacter pylori couples motility and diffusion to actively create a heterogeneous complex medium in gastric mucus, Phys. Rev. Lett. **116**, 198101 (2016).
[21] A. C. H. Tsang, P. W. Tong, S. Nallan, and O. S. Pak, Self-learning how to swim at low Reynolds number, Phys. Rev. Fluids **5**, 074101 (2020).
[22] Y. Jiao, F. Ling, S. Heydari, N. Heess, J. Merel, and E. Kanso, Learning to swim in potential flow, Phys. Rev. Fluids **6**, 050505 (2021).
[23] J. Qiu, N. Mousavi, K. Gustavsson, C. Xu, B. Mehlig, and L. Zhao, Navigation of micro-swimmers in steady flow: The importance of symmetries, J. Fluid Mech. **932**, A10 (2022).
[24] Z. Zou, Y. Liu, Y.-N. Young, O. S. Pak, and A. C. Tsang, Gait switching and targeted navigation of microswimmers via deep reinforcement learning, Commun. Phys. **5**, 158 (2022).
[25] G. Zhu, W.-Z. Fang, and L. Zhu, Optimizing low-Reynolds-number predation via optimal control and reinforcement learning, J. Fluid Mech. **944**, A3 (2022).
[26] K. Qin, Z. Zou, L. Zhu, and O. S. Pak, Reinforcement learning of a multi-link swimmer at low Reynolds numbers, Phys. Fluids **35**, 032003 (2023).
[27] S. Colabrese, K. Gustavsson, A. Celani, and L. Biferale, Flow navigation by smart microswimmers via reinforcement learning, Phys. Rev. Lett. **118**, 158004 (2017).
[28] J. K. Alageshan, A. K. Verma, J. Bec, and R. Pandit, Machine learning strategies for path-planning microswimmers in turbulent flows, Phys. Rev. E **101**, 043110 (2020).
[29] E. Schneider and H. Stark, Optimal steering of a smart active particle, Europhys. Lett. **127**, 64003 (2019).
[30] S. Muiños-Landin, A. Fischer, V. Holubec, and F. Cichos, Reinforcement learning with artificial microswimmers, Sci. Rob. **6**, eabd9285 (2021).
[31] Y. Yang, M. A. Bevan, and B. Li, Micro/nano motor navigation and localization via deep reinforcement learning, Adv. Theor. Simul. **3**, 2000034 (2020).
[32] L. Yang, J. Jiang, F. Ji, Y. Li, K.-L. Yung, A. Ferreira, and L. Zhang, Machine learning for micro- and nanorobots, Nat. Mach. Intell. **6**, 605 (2024).
[33] Y. Liu, Z. Zou, O. S. Pak, and A. C. Tsang, Learning to cooperate for low-Reynolds-number swimming: a model problem for gait coordination, Sci. Rep. **13**, 9397 (2023).
[34] L. Amoudruz and P. Koumoutsakos, Independent control and path planning of microswimmers with a uniform magnetic field, Adv. Intell. Syst. **4**, 2100183 (2022).
[35] M. R. Behrens and W. C. Ruder, Smart magnetic microrobots learn to swim with deep reinforcement learning, Adv. Intell. Syst. **4**, 2200023 (2022).
[36] E. M. Purcell, Life at low Reynolds number, Am. J. Phys. **45**, 3 (1977).
[37] S. J. Lighthill, *Mathematical biofluiddynamics* (SIAM, Philadelphia, 1975).
[38] See Supplemental Material at http://link.aps.org/supplemental/10.1103/9msg-hgqn for additional details on the dynamic model of the three-link swimmer; the PPO framework used to train the swimmer's control policy; the impact of training parameters on the swimmer's performance; and supplemental movies illustrating complex navigation tasks. The Supplemental Material also contains Refs. [37,41,42].
[39] E. M. Purcell, The efficiency of propulsion by a rotating flagellum, Proc. Natl. Acad. Sci. **94**, 11307 (1997).

[40] D. Tam and A. E. Hosoi, Optimal stroke patterns for Purcell's three-link swimmer, Phys. Rev. Lett. **98**, 068105 (2007).
[41] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, arXiv:1412.6980.
[42] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, Proximal policy optimization algorithms, arXiv:1707.06347.