communications
physics

ARTICLE

https://doi.org/10.1038/s42005-022-00935-x OPEN

Gait switching and targeted navigation of microswimmers via deep reinforcement learning

Zonghao Zou[1,4], Yuexin Liu[2,4], Y.-N. Young[2]✉, On Shun Pak[1]✉ & Alan C. H. Tsang[3]✉

Swimming microorganisms switch between locomotory gaits to enable complex navigation strategies such as run-and-tumble to explore their environments and search for specific targets. This ability of targeted navigation via adaptive gait-switching is particularly desirable for the development of smart artificial microswimmers that can perform complex biomedical tasks such as targeted drug delivery and microsurgery in an autonomous manner. Here we use a deep reinforcement learning approach to enable a model microswimmer to self-learn effective locomotory gaits for translation, rotation and combined motions. The Artificial Intelligence (AI) powered swimmer can switch between various locomotory gaits adaptively to navigate towards target locations. The multimodal navigation strategy is reminiscent of gait-switching behaviors adopted by swimming microorganisms. We show that the strategy advised by AI is robust to flow perturbations and versatile in enabling the swimmer to perform complex tasks such as path tracing without being explicitly programmed. Taken together, our results demonstrate the vast potential of these AI-powered swimmers for applications in unpredictable, complex fluid environments.

[1] Department of Mechanical Engineering, Santa Clara University, Santa Clara, CA 95053, USA. [2] Department of Mathematical Sciences, New Jersey Institute of Technology, Newark, NJ 07102, USA. [3] Department of Mechanical Engineering, The University of Hong Kong, Pokfulam Road, Hong Kong, China. [4] These authors contributed equally: Zonghao Zou, Yuexin Liu. ✉email: yyoung@njit.edu; opak@scu.edu; alancht@hku.edu

Swimming microorganisms have evolved versatile navigation strategies by switching their locomotory gaits in response to their surroundings[1]. Their navigation strategies typically involve switching between translation and rotation modes such as run-and-tumble and reverse-and-flick in bacteria[2–5], as well as run-stop-shock and run-and-spin in eukaryotes[6,7]. Such an adaptive, multimodal gait-switching ability is particularly desirable for biomedical applications of artificial microswimmers such as targeted drug delivery and microsurgery[8–12], which require navigation towards target locations in biological media with uncontrolled and/or unpredictable environmental factors[13–15].

Pioneering works by Purcell and subsequent studies demonstrated how simple reconfigurable systems with ingenious locomotory gaits can generate net translation and rotation, given the stringent constraints for locomotion at low Reynolds numbers[16]. Yet, the design of locomotory gaits becomes increasingly intractable when more sophisticated maneuvers are required or environmental perturbations are present. Existing microswimmers are therefore typically designed with fixed locomotory gaits and rely on manual interventions for navigation[8,17–21]. It remains an unresolved challenge in developing microswimmers with adaptive locomotory strategies similar to that of biological cells that can navigate complex environments autonomously. Modular microrobotics and the use of soft active materials[22,23] have been proposed to address the challenge.

More recently, the rapid development of artificial intelligence (AI) and its applications in locomotion problems[24–29] have opened different paths towards designing the next generation of smart microswimmers[30,31]. Various machine learning approaches have enabled the navigation of active particles in the presence of background flows[32,33], thermal fluctuations[34,35], and obstacles[36]. As minimal models, the microswimmers are often modeled as active particles with prescribed self-propelling velocities and certain degrees of freedom for speed variation and re-orientation. However, the complex adjustments in locomotory gaits required for such adaptations are typically not accounted for. Recent studies have begun to examine how different machine learning techniques enable reconfigurable microswimmers to evolve effective gaits for self-propulsion[37] and chemotactic repsonse[38].

Here, we combine reinforcement learning (RL) with artificial neural network to enable a simple reconfigurable system to perform complex maneuvers in a low-Reynolds-number environment. We show that the deep RL framework empowers a microswimmer to adapt its locomotory gaits in accomplishing sophisticated tasks including targeted navigation and path tracing, without being explicitly programmed. The multimodel gait switching strategies are reminiscent of that adopted by swimming microorganisms. Furthermore, we examine the performance of these locomotion strategies against perturbations by background flows. The results showcase the versatility of AI-powered swimmers and their robustness in media with uncontrolled environmental factors.

Results and discussion

Model reconfigurable system. We consider a simple reconfigurable system consisting of three spheres with radius R and centers $\mathbf{r}_i$ $(i = 1, 2, 3)$ connected by two arms with variable lengths and orientations as shown in Fig. 1a. This setup generalizes previous swimmer models proposed by Najafi and Golestanian[39] and Ledesma-Aguilar et al. [40] by allowing more degrees of freedom. The interaction between the system and the surrounding viscous fluid is modeled by low Reynolds number hydrodynamics, imposing stringent constraints on the locomotive capability of the system. Unlike the traditional paradigm where the locomotory gaits are prescribed in advance[39–44], here we exploit a deep RL framework to enable the system to self-learn a set of locomotory gaits to swim along a target direction, θ_T. We employ a deep neural network based on the Actor-Critic structure and implement the Proximal Policy Optimization (PPO) algorithm[29,45] to train and update the agent (i.e., AI) in charge of the decision making process (Fig. 1b). The deep RL framework here extends previous studies from discrete action spaces to continuous action spaces[32,35,37,46], enhancing the swimmer's capability in developing more versatile locomotory gaits for complex navigation tasks (see the "Methods" section for implementation details of the Actor-Critic neural network and PPO algorithm).

Hydrodynamic interactions. The interaction between the spheres and their surrounding fluid is governed by the Stokes equation ($\nabla p = \mu \nabla^2 \mathbf{u}, \nabla \cdot \mathbf{u} = 0$). Here, p, μ and $\mathbf{u}$ represent, respectively, the pressure, dynamic viscosity, and velocity field. In this low Reynolds number regime, the velocities of the spheres $\mathbf{V}_i$ and the forces $\mathbf{F}_i$ acting on them can be related linearly as

$$\mathbf{V}_i = \mathbf{G}_{ij}\mathbf{F}_j, \tag{1}$$

where $\mathbf{G}_{ij}$ is the Oseen tensor[47–49] given by

$$\mathbf{G}_{ij} = \begin{cases} \frac{1}{6\pi\mu R}\mathbf{I}, \\ \frac{1}{8\pi\mu|\mathbf{r}_i - \mathbf{r}_j|}(\mathbf{I} + \hat{\mathbf{r}}_{ij}\hat{\mathbf{r}}_{ij}). \end{cases} \tag{2}$$

Here, $\mathbf{I}$ is the identity matrix and $\hat{\mathbf{r}}_{ij} = (\mathbf{r}_i - \mathbf{r}_j)/|\mathbf{r}_i - \mathbf{r}_j|$ denotes the unit vector between spheres i and j. The torque acting on the sphere i is calculated by $\mathbf{T}_i = \mathbf{r}_i \times \mathbf{F}_i$. The rate of actuation of the arm lengths $\dot{L}_1$, $\dot{L}_2$ and the intermediate angle $\dot{\theta}_{31}$ can be expressed in terms of the velocities of the spheres $\mathbf{V}_i$. The kinematics of the swimmer is fully determined upon applying the force free ($\sum_i \mathbf{F}_i = 0$) and torque-free ($\sum_i \mathbf{T}_i = 0$) conditions. The Oseen tensor hydrodynamic description is valid when the spheres are not in close proximity ($R \ll L$). We therefore constrain the arm and angle contractions such that $0.6L \le L_1, L_2 \le L$ and $2\pi/3 \le \theta_{31} \le 4\pi/3$.

The actuation rate of the arm lengths $\dot{L}_1, \dot{L}_2$ can be expressed in terms of the relative velocities of the spheres parallel to the arm orientations:

$$(\mathbf{V}_2 - \mathbf{V}_1) \cdot \hat{\mathbf{r}}_{21} = \dot{L}_1, \tag{3}$$

$$(\mathbf{V}_3 - \mathbf{V}_2) \cdot \hat{\mathbf{r}}_{32} = \dot{L}_2, \tag{4}$$

The actuation rate of the intermediate angle $\dot{\theta}_{31}$ can be expressed in terms of the relative velocities of the spheres perpendicular to the arm orientations:

$$(\mathbf{V}_2 - \mathbf{V}_1) \cdot \frac{d\hat{\mathbf{r}}_{21}}{d\theta_1} = L_1\dot{\theta}_1, \tag{5}$$

$$(\mathbf{V}_3 - \mathbf{V}_2) \cdot \frac{d\hat{\mathbf{r}}_{32}}{d\theta_2} = L_2\dot{\theta}_2, \tag{6}$$

$$\dot{\theta}_1 - \dot{\theta}_2 = \dot{\theta}_{31}, \tag{7}$$

where $\dot{\theta}_1$ and $\dot{\theta}_2$ are the arm rotation speeds. Together with the Oseen tensor description of the hydrodynamic interaction between the spheres, Eqs. (1) and (2) in the main text, and the overall force-free and torque-free conditions, the kinematics of the swimmer is fully determined.

In presenting our results, we scale lengths by the fully extended arm length L, velocities by a characteristic actuation rate of the arm V_c, and hence time by L/V_c and forces by $\mu L V_c$ (see Non-dimensionalization under Supplementary methods).

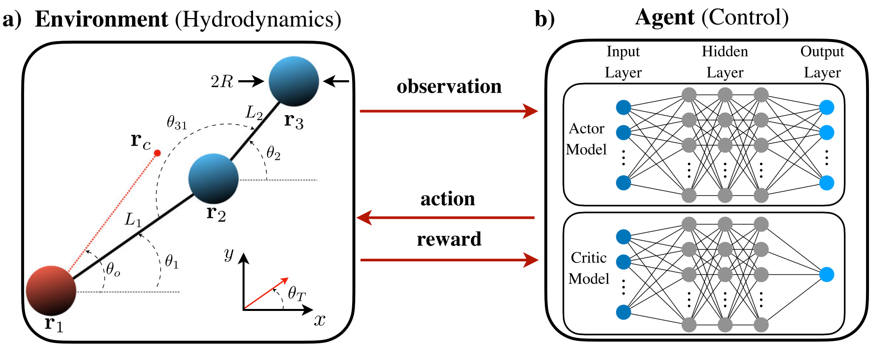

Fig. 1 Schematics of the model microswimmer and the deep neural network with Actor-Critic structure. a Schematic of the model microswimmer consisting of three spheres with raidus R and centers $\mathbf{r}_i$ $(i = 1, 2, 3)$. We mark the leftmost sphere $\mathbf{r}_1$ as red and the other two spheres $\mathbf{r}_2$, $\mathbf{r}_3$ as blue to indicate the current orientation of the swimmer. The spheres are connected by two arms with variable lengths L_1, L_2 and orientations θ_1, θ_2, where θ_{31} is the intermediate angle between two arms. The swimmer's orientation θ_o is defined based on the relative position between the swimmer's centroid $\mathbf{r}_c = \sum_i \mathbf{r}_i/3$ and $\mathbf{r}_1$ as $\theta_o = \arg(\mathbf{r}_c - \mathbf{r}_1)$. The swimmer is trained to swim along a target direction θ_T. **b** Schematic of Actor-Critic neural networks. Both networks consist of three sets of layers (input layer, hidden layer, and output layer). Each layer is composed of neurons (marked as nodes). The weights of the neural network are illustrated as links in between the nodes. The input layer has the same dimension as the observation. The three linear hidden layers have the dimension of 64,32,32, respectively. The output layer dimension of the actor network is the same as the action space dimension, whereas the output layer of the actor network has only 1 neuron. We discuss the general idea as follows: based on the current observation, a reinforcement learning agent decides the next action using the Actor neural network. The next action is then evaluated by the Critic neural network to guide the training process. The swimmer performs the action advised by the agent and interacts with the hydrodynamic environment, leading to movements that constitute the next observation and reward. Both the Actor and Critic neural network are updated periodically to improve the overall performance. See more details in the "Methods" section.

Targeted navigation. We first use the deep RL framework to train the model system in swimming along a target direction θ_T, given any arbitrary initial swimmer's orientation θ_o. The swimmer's orientation is defined based on the relative position between the swimmer's centroid $\mathbf{r}_c = \sum_i \mathbf{r}_i/3$ and $\mathbf{r}_1$ as $\theta_o = \arg(\mathbf{r}_c - \mathbf{r}_1)$ (Fig. 1).

In the RL algorithm, the state $s \in (\mathbf{r}_1, L_1, L_2, \theta_1, \theta_2)$ of the system is specified by the sphere center $\mathbf{r}_1$, arm lengths L_1, L_2, and arm orientations θ_1, θ_2. The observation $o \in (L_1, L_2, \theta_{31}, \cos\theta_d, \sin\theta_d)$ is extracted from the state, where θ_{31} is the intermediate angle and $\theta_d = \theta_T - \theta_o$ is the difference between the target direction θ_T and the swimmer's orientation θ_o; note that the angle difference is expressed in terms of $(\cos\theta_d, \sin\theta_d)$ to avoid discontinuity in the orientation space. The AI decides the swimmer's next action based on the observation using the Actor neural network: for each action step Δt, the swimmer performs an action $a \in (\dot{L}_1, \dot{L}_2, \dot{\theta}_{31})$ by actuating its two arms, leading to swimmer displacement. To quantify the success of a given action, the reward is measured by the displacement of the swimmer's centroid along the target direction, $r_t = (\mathbf{r}_{c_{t+1}} - \mathbf{r}_{c_t}) \cdot (\cos\theta_T, \sin\theta_T)$.

We divide the training process into a total of N_e episodes, with each episode consisting of $N_t = 150$ learning steps. To ensure a full exploration of the observation space o, both the initial swimmer state s and the target direction θ_T are randomized in each episode. Based on the training results after every 20 episodes, the critic neural network updates the AI to maximize the expected long-term rewards $\mathbf{E}[R_{t=0}|\pi_\theta]$, where π_θ is the stochastic control policy, $R_t = \sum_{t'}^{\infty} \gamma^{t'-t} r_{t'}$ is the infinite-horizon discounted future returns, and γ is the discount factor measuring the greediness of the algorithm[45,50]. A large discount factor $\gamma = 0.99$ is set here to ensure farsightedness of the algorithm. As the episodes proceed, the Actor-Critic structure progressively trains the AI and thereby enhances the performance of the swimmer.

In Fig. 2 (Supplementary Movie 1) we visualize the navigation of a trained swimmer along a target direction θ_T, given a substantially different initial orientation, θ_o. The swimmer's targeted navigation is accomplished in three stages: (1) in the initial phase (blue curve and regime), the swimmer employs "steering" gaits primarily for re-orientation, followed by (2) "transition" phase (red curve and regime) in which the swimmer continues to adjust its direction while self-propelling, before reaching (3) the "translation" phase (green curve and regime), in which the re-orientation is complete and the swimmer simply self-propels along the target direction. This example illustrates

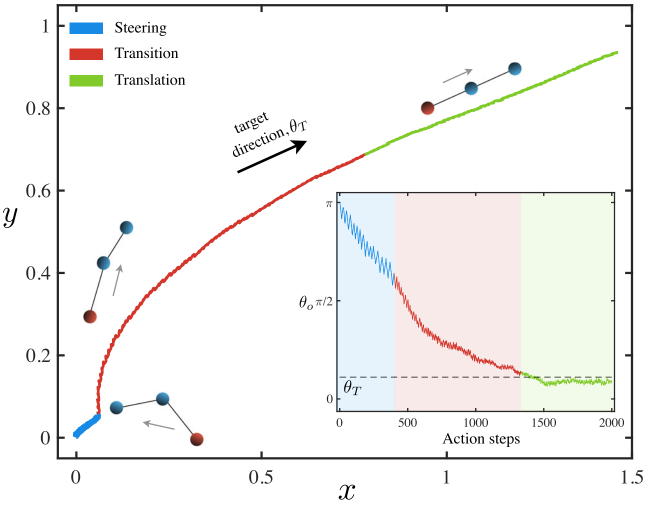

Fig. 2 Example of target navigation utilizing three distinct locomotory gaits. The Artificial Intelligence powered swimmer switches between distinct locomotory gaits (steering, transition, translation) advised by the reinforcement learning algorithm to steer itself towards a specified target direction θ_T (black arrow) and swim along the target direction afterwards. Different parts of the swimmer's trajectory are colored to represent the locomotion due to different locomotory gaits, where the steering, transition, and translation gaits are marked as blue, red, green, respectively. Schematics of the swimmer configurations (not-to-scale) are shown for illustrative purpose, where the leftmost sphere is marked as red and other two spheres marked as blue to indicate the swimmer's current orientation (gray arrows). The inset shows the change in swimmer's orientation θ_o over action steps. An animation of this simulation is shown in Supplemental Movie 1.

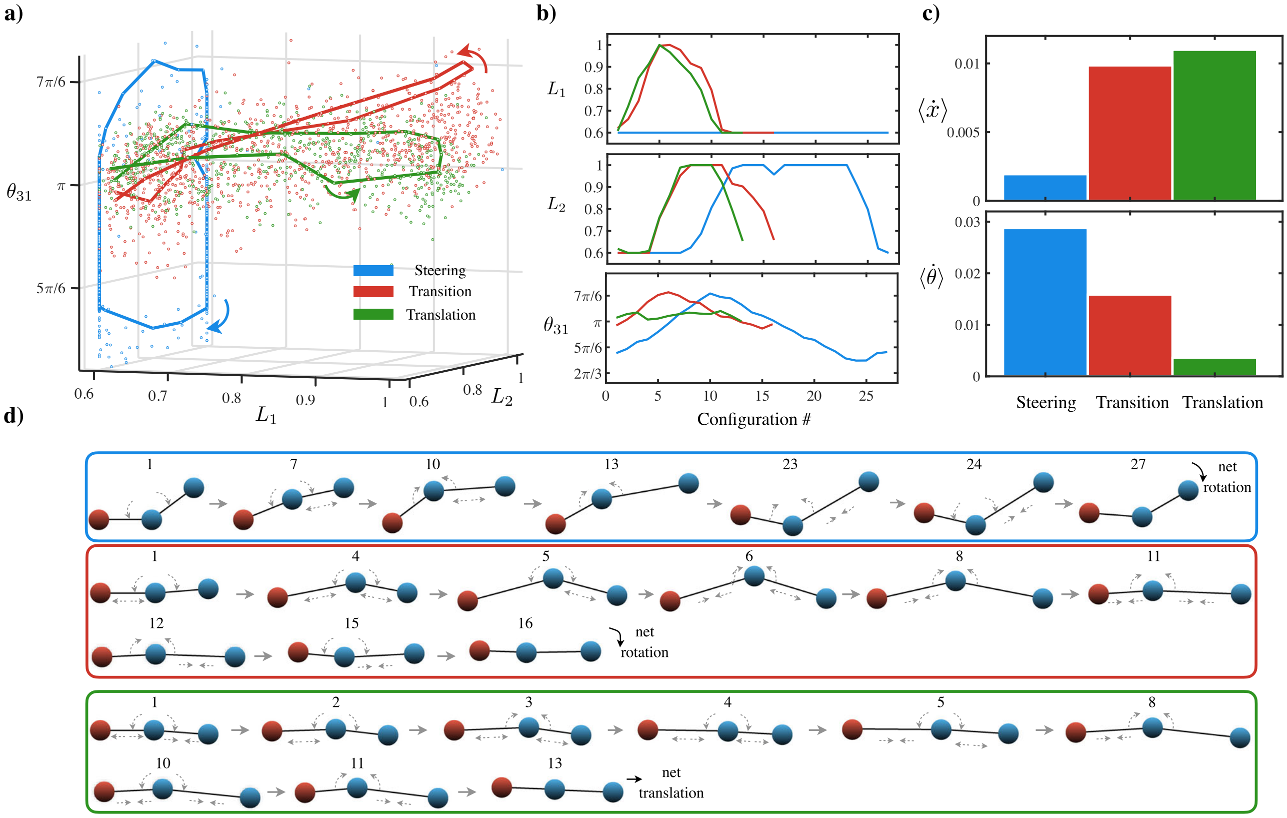

Fig. 3 Analysis of configurational changes revealing three distinct modes of locomotory gaits. The steering, transition, and translation gaits are marked as blue, red, green, respectively. **a** A 3D configuration plot for a typical simulation which the swimmer aligns with the target direction via a counterclockwise rotation, where L_1, L_2 are the arm lengths and θ_{31} is the intermediate angle. Each dot represents one specific configuration of a locomotory gait. The solid lines mark an example cycle of each locomotory gait. **b** The changes in the arm lengths L_1 and L_2 and the intermediate angle θ_{31} with respect to the configuration number for each locomotory gait. **c** The average translational velocity $\langle\dot{x}\rangle$ and rotational velocity $\langle\dot{\theta}\rangle$ are calculated by averaging the centroid translation along the target direction θ_T and the change of swimmer's orientation θ_o over the total number of action steps for each locomotory gaits. **d** Representative configurations labeled with the configuration number are displayed to illustrate the configurational changes for each selected sequence of locomotory gaits for the steering (blue box), transition (red box), and translation (green box) modes. The leftmost sphere of the swimmer is marked as red and other two spheres are marked as blue to indicate the swimmer's current orientation. The gray arrows indicate the contraction/extension of the arms and the intermediate angle. For illustration, the reference frame of the configurations are rotated consistently such that the left arm of the first configuration is aligned horizontally in each sequence. The animation of counterclockwise and clockwise simulations are shown in the Supplementary Movies 2, 3 and 7, 8.

how an AI-powered reconfigurable system evolves a multimodal navigation strategy without explicitly programmed or relying on any prior knowledge of low-Reynolds-number locomotion. We next analyze the locomotory gaits in each mode in the evolved strategy.

Multimodal locomotory gaits. Here we examine the details of the locomotory gaits acquired by the swimmer for targeted navigation in the steering, transition, and translation modes. We distinguish these gaits by visualizing their configurational changes in the three-dimensional (3D) configuration space of the swimmer (L_1, L_2, θ_{31}) in Fig. 3. Here we utilize an example of a swimmer navigating towards a target direction with $|\theta_d| > \pi/2$ to illustrate the switching between different locomotory gaits (Fig. 3a), Supplementary Movies 2 and 3). The swimmer needs to re-orient itself in the counter-clockwise direction in this example; an example for the case of clockwise rotation is included in the Supplementary Note 1 (Supplementary Fig. 1, Movies 7 and 8). The dots in Fig. 3a represent configurations at different action steps. The configurations for the steering (blue dots), transition (red dots), and translation (green dots) gaits are clustered in different regions in the configuration space. A representative sequence of configurational changes for each mode of gaits are shown as solid lines to aid visualization (Fig. 3a).

We further examine the evolution of L_1, L_2, and θ_{31} using the representative sequences of configurational changes identified in Fig. 3a for each mode of gaits. For the steering gaits (Fig. 3b, blue lines and Fig. 3d, blue box), the swimmer repeatedly extends and contracts L_2 and θ_{31}, but keeps L_1 constant (the left arm rests in the fully contracted state). The steering gaits thus reside in the $L_2{-}\theta_{31}$ plane in Fig. 3a (blue line). The large variation in θ_{31} generates net rotation, substantially re-orientating the swimmer orientation with a relatively small net translation (Fig. 3c). For the transition gaits (Fig. 3b, red lines and Fig. 3d, red box), the swimmer repeatedly extends and contracts all L_1, L_2 and θ_{31}, leading to significant amounts of both net rotation and translation (Fig. 3c). In the configuration space (Fig. 3a), the transition gaits tilt into the $L_1{-}L_2$ plane with an average θ_{31} less than π (red line). Compared with the steering gaits, the variation of θ_{31} becomes more restricted (Fig. 3b), resulting in smaller net rotation for fine tuning of the swimmer's orientation in the transition phase. Finally, for the translation gaits (Fig. 3b, green lines and Fig. 3d, green box), the swimmer's orientation is aligned with the target direction ($\theta_d \approx 0$); the swimmer repeatedly extends and contracts L_1 and L_2, while keeping θ_{31} close to π (i.e., all three

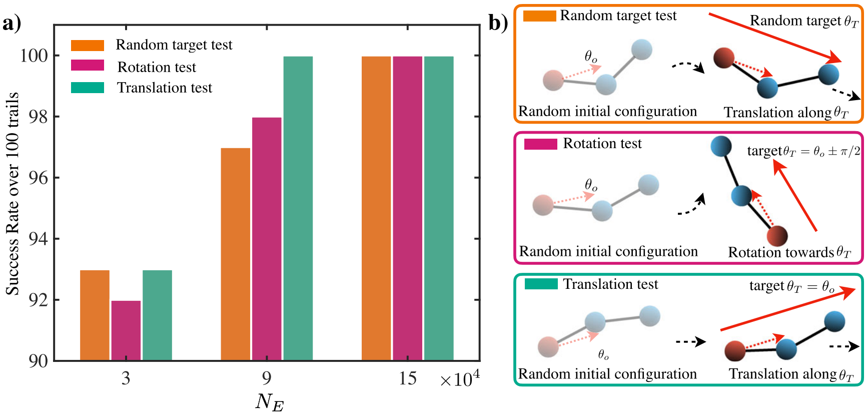

Fig. 4 Analysis of the swimmer's performance with increasing number of episodes. Number of episodes N_e indicates the total training time of the swimmer. Each episode during training contains a fixed amount of action steps $N_l = 150$. **a** We used three tests (random target test, rotation test and translation test) to measure the swimmer's performance in a fixed number of training steps $N_l = 150$. For all tests, the swimmer starts with a random initial configuration to ensure a full exploration of the observation space. A total of 100 trials are considered for each test with swimmers trained at different N_e. A swimmer with insufficient training (3×10^4 episodes) may occasionally fails in the three tests (success rate ≈ 90%). At $N_e = 9 \times 10^4$, the swimmer masters translation and improves its rotation ability. When N_e increases to 1.5×10^5, the swimmer obtains a 100% success rate in all tests. **b** Schematics of the random target test, rotation test, and translation test. The leftmost sphere is marked as red and other spheres are marked as blue to indicate the swimmer's orientation θ_o (red dashed arrows). Given a random initial configuration, we test the swimmer's ability to translate along or rotate towards a target direction θ_T (solid red arrows). The black dashed arrows indicate the swimmer's intended moving direction.

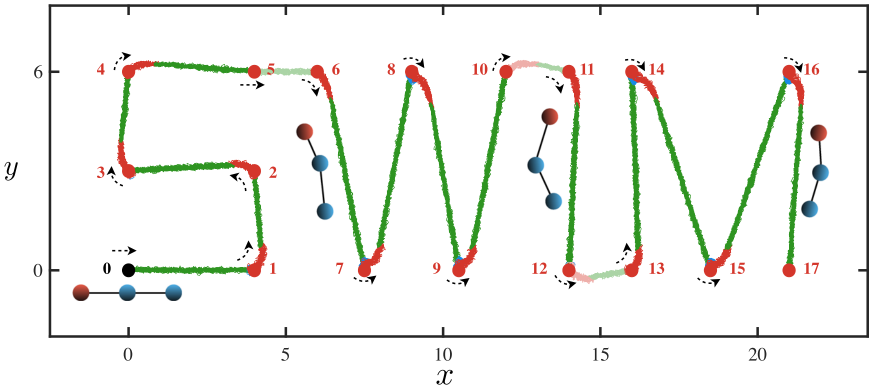

Fig. 5 Demonstration of complex navigation capability of Artificial Intelligence powered swimmer. The Artificial Intelligence powered swimmer switches between various locomotory gaits autonomously in tracing a complex trajectory "SWIM". The trajectory of the central sphere of the swimmer is colored based on the mode locomotory gaits: steering (blue), transition (red), and translation (green). The swimmer is given a lists of target points (1–17) with one target point at a time. The black arrows at each point indicate the intended direction of the swimmer. From the current target point, the swimmer determines the target direction for the next action step $t+1$, $\theta_{T_{t+1}}$ and adapts the locomotory gaits based on its AI in navigating towards that direction. Schematics of the swimmer configurations (not-to-scale) are shown for illustrative purposes, where the leftmost sphere is marked as red and other two spheres are marked as blue to indicate the swimmer's current orientation. An animation of this simulation is shown in Supplemental Movie 4.

spheres of the swimmer are aligned), resembling the swimming gaits of Najafi–Golestanian swimmers[39,51]. In the configuration space (Fig. 3a), the translation gaits reside largely in the L_1–L_2 plane with an approximately zero average θ_{31}, generating the maximum net translation with minimal net rotation (Fig. 3c). The details of gaits categorization are summarized under Supplementary methods.

It is noteworthy that the multimodal navigation strategy emerges solely from the AI without relying on prior knowledge of locomotion. The switching between rotation, transition, and translation gaits is analogous to the switching between turning and running modes observed in bacterial locomotion[2,5]. These results demonstrate how an AI-powered swimmer, without being explicitly programmed, self-learns complex locomotory gaits from rich action and configuration spaces and undergoes autonomous gait switching in accomplishing targeted navigation.

Performance evaluation. Here we investigate the improvement of swimmer's performance with increased number of training episodes N_e. At initial stage of training with a small N_e, the swimmer may fail to identify the right sets of locomotory gaits to achieve targeted navigation due to insufficient training. Continuous training with increased number of episodes would enable the swimmer to identify better locomotory gaits to complete navigation tasks. Here we measure the improvement of swimmer's performance with increased N_e by three locomotion tests: (1) Random target test: the swimmer is assigned a target direction selected randomly from a uniform distribution in $[0, 2\pi]$; (2) Rotation test: the swimmer is assigned a targeted direction with a large angle of difference with swimmer's orientation (i.e., $\theta_d = \pm \pi/2$); (3) Translation test: the swimmer is assigned a target direction equal to the swimmer's orientation (i.e., $\theta_d = 0$). A test is considered to be successful if the swimmer travels along the

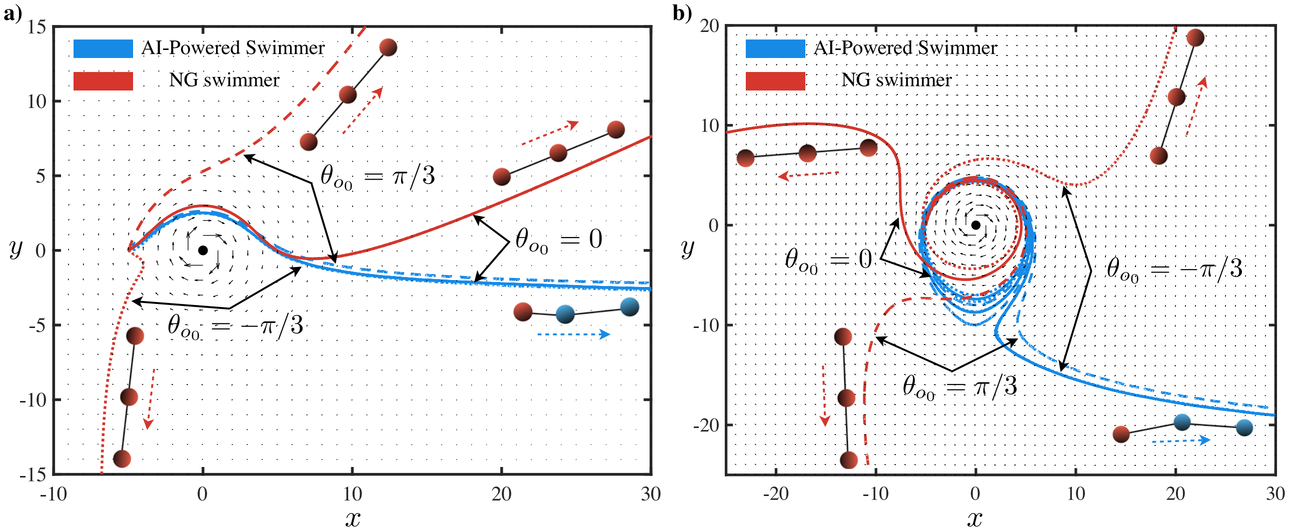

Fig. 6 Analysis of the performance of targeted navigation under the influence of flows. a The Artificial Intelligence powered swimmer and the Najafi–Golestanian (NG) swimmer escape from a relatively weak rotlet flow, $\mathbf{u}_\infty = -\boldsymbol{\gamma} \times \mathbf{r}/r^3$, where $\boldsymbol{\gamma} = \gamma \mathbf{e}_z$ prescribes the strength of the rotlet in the z-direction, $r = |\mathbf{r}|$ is the magnitude of the position vector $\mathbf{r}$ from the origin ($\gamma = 0.15$). The leftmost sphere of the AI-powered swimmer is marked as red and other spheres are marked as blue to indicate the swimmer's current orientation (blue dashed arrow). The NG swimmer is colored red with its orientation marked as red dashed arrows. Three sets of trajectories (dashed, dotted, and solid lines) are shown with different initial swimmer orientation θ_{o_0}. The AI-powered swimmer travels to the right regardless of its initial orientation whereas the trajectory for the NG swimmer is highly affected by the rotlet flow. **b** We compare the trajectories of the AI-powered swimmer and the NG swimmer in a strong rotlet flow ($\gamma = 1.5$). The NG swimmer completely loses control in the flow, while the AI-powered swimmer maintains its orientation towards the positive x-direction, with similar trajectories for different initial orientations. The animation of the two simulations are shown in Supplemental Movies 5 and 6.

target direction for a distance of 5 *unit* in 10,000 action steps. These tests ensure that the trained swimmer acquires a set of effective locomotory gaits to swim along any specified direction with robust rotation and translation.

We consider the success rates of the three tests over 100 trials (Fig. 4). For $N_e = 3 \times 10^4$, success rates of around 90% are obtained for the three tests. When N_e is increased to 9×10^4, the swimmer masters translation with a 100% success rate but still needs more training for rotation. When N_e is increased further to 15×10^4, the swimmer obtains 100% success rates for all tests. This result demonstrates the continuous improvement in the robustness of targeted navigation with increased N_e up to 15×10^4. As we further increase N_e, we found the relationship between N_e and performance to be non-monotonic. For a total training episodes much greater than $N_e = 15 \times 10^4$, the overall success rate will begin to drop and eventually fluctuate around 95%. We selected the trained result at $N_e = 15 \times 10^4$ for the best overall performance.

To better understand the swimmer's training process, we also varied the number of steps in each episodes, N_l. For a range from 100 to 300 and a fixed total episodes N_e, we found $N_l = 150$ provides the most efficient way to balance translation and rotation and require least amount of action steps to complete both the rotation and translation tests. We remark that, when $N_l = 100$, the swimmer was only able to translate but not to rotate, indicating the significant role N_l plays in learning.

Lastly, we remark that the swimmer appears to require more training, both in N_e and N_l, to learn rotation compared to translation. This may be attributed to the inherit complexity of rotation gaits, where the swimmer needs to actuate its intermediate angle in addition to the actuation of the two arms required in translation gaits.

Path tracing–"SWIM". Next we showcase the swimmer's capability in tracing complex paths in an autonomous manner. To illustrate, the swimmer is tasked to trace out the English word “SWIM" (Fig. 5, Supplementary Movie 4). We note that the hydrodynamic calculations required to design the locomotory gaits to trace such complex paths become quickly intractable as the complexity increases. Here, instead of explicitly programming the gaits of the swimmer, we only select target points (p_i, $i = 1, 2, \ldots, 17$, red spots in Fig. 5) as landmarks and require the swimmer to navigate towards these landmarks with its own AI, with the target directions at action step $t+1$ given by $\theta_{T_{t+1}} = \arg(\mathbf{p}_i - \mathbf{r}_{c_t})$. The swimmer is assigned with the next target point p_{i+1} when its centroid is within a certain threshold (0.1 of the fully extended arm length) from p_i. The completion of these multiple navigation tasks sequentially enables the swimmer to successfully trace out the word “SWIM" with a high accuracy (Fig. 5, Supplementary Movie 4). In accomplishing this task, the swimmer switches between the three modes of locomotory gaits autonomously to swim towards individual target points and turn around the corners of the path based on the AI-powered navigation strategy. It is noteworthy that the swimmer is able to navigate around some corners (e.g., at target points 4 and 6) without activating the steering gaits, which are employed for corners with more acute angles (e.g., at target points 8, 14, and 16). While past approaches based on detailed hydrodynamic calculations, manual interventions, or other control methods may also complete such tasks, here we present reinforcement learning as an alternative approach in accomplishing these complex maneuvers in a more autonomous manner.

Robustness against flows. Last, we examine the performance of targeted navigation under the influence of flows (Fig. 6a, b, Supplementary Movies 5, 6). In particular, to determine to what extent the AI-powered swimmer is capable of maintaining its target direction against flow perturbations, we use the same AI-powered swimmer trained without any background flow, and impose a rotational flow generated by a rotlet at the origin[47,48],

$\mathbf{u}_\infty = -\boldsymbol{\gamma} \times \mathbf{r}/r^3$, where $\boldsymbol{\gamma} = \gamma \mathbf{e}_z$ prescribes the strength of the rotlet in the z-direction, $r = |\mathbf{r}|$ is the magnitude of the position vector $\mathbf{r}$ from the origin (see the section "Simulations of background flow" under Supplementary methods). Here the AI-powered swimmer is tasked to navigate towards the positive x-direction under flow perturbations due to the rotlet. We examine how the swimmer adapts to the background flow when performing this task. For comparison, we contrast the resulting motion of the AI-powered swimmer with that of an untrained swimmer (i.e., a Najafi–Golestanian (NG) swimmer that performs only fixed locomotory gaits without any adaptivity[39]). Without the background flow, both swimmers self-propel with the same speed. Both swimmers are initially placed close to the rotlet with $\mathbf{r}_c = -5\mathbf{e}_x$ and we sample their performance with three different initial orientations: $\theta_{o_0} = -\pi/3$, 0, and $\pi/3$, under different flow strengths. Under a relatively weak flow ($\gamma = 0.15$, Fig. 6a), Supplementary Movie 5), the AI-powered swimmer is capable of navigating towards the positive x-direction regardless of its initial orientations against flow perturbations. In contrast, the trajectories of the NG swimmer are largely influenced by the rotlet flow passively depending on the initial orientation of the swimmer. For an increased flow strength ($\gamma = 1.5$, Fig. 6b, Supplementary Movie 6), the NG swimmer completely loses control of its direction and is scattered by the rotlet into different directions again due to the absence of any adaptivity. Under such a strong flow, the AI-powered swimmer initially circulates around the rotlet but eventually manages to escape from it, navigating to the positive x-direction successfully with similar trajectories for all initial orientations. We note that the vorticity experienced by the swimmer in this case is comparable with typical re-orientation rates of the AI-powered swimmer. We also remark that when navigating under flow perturbations, the AI-powered swimmer adopts the transition gaits to constantly re-orient itself towards the positive x-direction and self-propels along that direction eventually. These results showcase the AI-powered swimmer's capability in adapting its locomotory gaits to navigate robustly against flows.

Conclusions

In this work, we present a deep RL approach to enable navigation of an artificial microswimmer via gait switching advised by the AI. In contrast to previous works that considered active particles with prescribed self-propelling velocities as minimal models[32,34,35] or simple one-dimensional swimmers[37,38,46], here we demonstrate how a reconfigurable system can learn complex locomotory gaits from rich and continuous action spaces to perform sophisticated maneuvers. Through RL, the swimmer develops distinct locomotory gaits for a multimodal (i.e., steering, transition, and translation) navigation strategy. The AI-powered swimmer can adapt its locomotory gaits in an autonomous manner to navigate towards any arbitrary directions. Furthermore, we show that the swimmer can navigate robustly under the influence of flows and trace convoluted paths. Instead of explicitly programming a swimmer to perform these tasks in the traditional approach, the swimmer is advised by the AI to perform complex locomotory gaits and autonomous gait switching in accomplishing these navigation tasks. The multimodal strategy employed by the AI-powered swimmer is reminiscent of the run-and-tumble in bacteria[2,5]. Taken together, our results showcase the vast potential of this deep RL approach in realizing adaptivity similar to that of biological organisms for robust locomotive capabilities. Such adaptive behaviors are crucial for future biomedical applications of artificial microswimmers in complex media with uncontrolled and/or unpredictable environmental factors.

We finally discuss several possibilities for subsequent investigations based on this deep RL approach. While we demonstrate only planar motion in this work, the approach can be readily extended to three-dimensional navigation by allowing out-of-plane rotation the swimmer's arms with expanded observation and action spaces for the additional degrees of freedom. Moreover, the deep RL framework is not tied to any specific swimmers; a simple multi-sphere system is used in this work for illustration, and the same framework applies to other reconfigurable systems. We also remark that the AI-powered swimmer is able to overcome some influences of flows even though such flows were absent in the training. Subsequent investigations including the flow perturbation in the training may lead to even more powerful AI that could exploit the flows to further enhance the navigation strategies. Another practical aspect to consider is the effect of Brownian noise[52–54]. Specifically, the characterization of the effect of thermal fluctuations in both the training process of the swimmer and its resulting navigation performance is currently underway. In addition to flow and thermal fluctuations, other environmental factors, including the presence of physical boundaries and obstacles, may be addressed in similar manners in future studies. The deep RL approach here opens an alternative path towards designing adaptive microswimmers with robust locomotive and navigation capabilities in more complex, realistic environments.

Methods

Here we briefly explain the Proximal Policy Optimization (PPO) alogrithm we used to train our AI-powered swimmer.

In the PPO algorithm, the agent's motion control is managed with a neural network with an Actor-Critic structure. The Actor network can be considered as a stochastic control policy $\pi_\theta(a_t|o_t)$, where it generates an action a_t given an observation o_t following a Gaussian distribution. Here θ represents all the parameters of the actor neural network. The Critic network is used to compute the value function V_ϕ by assuming the agent starts at an observation o and acts according to a particular policy π_θ. The parameters in the critic network is represented as ϕ.

To effectively train the swimmer, we divide the total training process into episodes. Each episode can be considered as one round, which terminates after a fixed amount of training steps ($N_\mathrm{l} = 150$). To ensure fully exploration of the observation space, we randomly initialize the swimmer's geometric configurations ($L_1, L_2, \theta_1, \theta_2$) and the target direction (θ_T) at the beginning of each episode.

At time t, the agent receives its current observation o_t and samples action a_t based on the policy π_θ. Given a_t, the swimmer interacts with its surrounding and calculates the next state s_{t+1} and reward r_t. The next observation o_{t+1} extracted from s_{t+1} is sent to the agent for the next iteration. All the observations, actions, rewards and sampling probabilities are stored for the agent's update. The update process begins after running fix amount of episodes $N_\mathrm{E} = 20$ (Total training steps of an update is therefore: $N = N_\mathrm{E}{}^*N_\mathrm{l} = 3000$). The goal for the update is to optimize θ so that the expected long term rewards $J(\pi_\theta) = \mathbf{E}[R_{t=0}|\pi_\theta]$ is maximized.

The expectation is taken with respect to each running episode, τ. Here, we use the infinite-horizon discounted returns $r_t = \sum_{t'}^{\infty} \gamma^{t'-t} r_{t'}$, where γ is the discount factor measuring the greediness of the algorithm. We set $\gamma = 0.99$ ensuring its farsightedness. To solve this optimization problem, we use the typical policy gradient approach estimation: $\nabla_\theta J(\pi_\theta)$. More specifically, we implemented the clipped advantage PPO algorithm to avoid large changes in each gradient update. We estimated the surrogate objective $J(\pi_\theta)$ by clipping the probability ratio $r(\theta)$ times the advantage function $\hat{A}_t$. The probability ratio measures the probability of selecting an action for the current policy over the old policy ($r(\theta) = \frac{\pi_\theta(a|o)_{N\times 1}}{\pi_{\theta_\mathrm{old}}(a|o)_{N\times 1}}$). The advantage function $\hat{A}_t$ describes the relative advantage of taking an action a based on an observation o over a randomly selected action and is calculated by subtracting the value function $V_{N\times 1}$ from the discounted return $R_{N\times 1}$ ($\hat{A}_t = R_{N\times 1} - V_{N\times 1}$).

We then update the parameters θ, ϕ via a typical gradient descent algorithm: Adam optimizer. The full detail for our implementation is included in the Algorithm 1 and 2 below. Here, K is the total epoch number. N_l is the number of steps in one episode, and N is the total number of steps for each update. The PPO algorithm uses fixed-length trajectory segments τ. During each iteration, each of N_A parallel actors collect T time steps of data, then we construct the surrogate loss on these $N_\mathrm{A} T$ time steps of data, and optimize it with Adam for K epochs.

In the following we present the algorithm tables for the PPO algorithm employed in this work. We refer the readers to classical monographs for more details[45].

Algorithm 1. Environment

```
for  time step $t = 0, 1, \ldots$ do
    if $\mathrm{mod}(t, N_l) = 0$ then
        Reset state $s_t$
        Compute observation $o_t$
    end if
    Sample action $a_t$ from policy $\pi_\theta$
    Evaluate the next state $s_{t+1}$ and reward $r_t$ following the swimmer's hydrodynamics
    Compute the next observation $o_{t+1}$ from state $s_{t+1}$
    if  $t = 0$ or $\mathrm{mod}(t, N) \neq 0$ then
        append observation $o_{t+1}$, action $a_t$, reward $r_t$ and action sampling probability $\pi_\theta(a_t|o_t)$ to observation list $o_{N\times5}$, action list $a_{N\times3}$, reward list $R_{N\times1}$ and action sampling probability list $\pi_{\theta_{\mathrm{old}}}(a|o)_{N\times1}$
    else
        Update the Agent using Algorithm 2
    end if
end for
```

Algorithm 2. Proximal Policy Optimization, Actor-Critic, Update the Agent

```
Input: Initial policy parameter $\theta$, initial value function parameter $\phi$
for  $k = 0, 1, 2, \ldots K$  do
    Compute infinite-horizon discounted returns $R_{N\times1}$
    Evaluate expected returns $V_{N\times1}$ using observations $o_{N\times5}$ and value function $V_\phi$
    Compute the advantage function: $\hat{A}_t = R_{N\times1} - V_{N\times1}$
    Evaluate the probability for policy $\pi_\theta$ using observations $o_{N\times5}$ and actions $a_{N\times3}$, store the probability to $\pi_\theta(a|o)_{N\times1}$
    Compute the probability ratio: $r(\theta) = \frac{\pi_\theta(a|o)_{N\times1}}{\pi_{\theta_{\mathrm{old}}}(a|o)_{N\times1}}$
    Compute the clipped surrogate loss function:
    $L^{\mathrm{CLIP}}(\theta) = \mathbb{E}[\min(r(\theta)\hat{A}_t,\ \mathrm{clip}\,(r(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)]$, with constant $\epsilon$
    Compute the value-function loss: $L^{\mathrm{VF}}(\phi) = \frac{1}{2}\mathbf{E}[(R_{N\times1} - V_{N\times1})^2]$
    Compute the entropy loss: $L^{\mathrm{S}} = \alpha S[\pi_\theta]$, with constant $\alpha$
    Compute the total loss: $L(\theta, \phi) = -L^{\mathrm{CLIP}}(\theta) + L^{\mathrm{VF}}(\phi) - L^{\mathrm{S}}$
    Optimize surrogate $L$ with respect to $(\theta, \phi)$, with $K$ epochs and minibatch size $M \leq N_A T$, with $N_{\mathrm{A}}$ is the number of parallel actors and $T$ is the time step.
    $\theta_{\mathrm{old}} \leftarrow \theta$, $\phi_{\mathrm{old}} \leftarrow \phi$
end for
```

Data availability

The data and Supplementary movies 1–8 that support the findings of this study are available from the corresponding author upon reasonable request.

Code availability

The codes that support the findings of this study are available from the corresponding author upon reasonable request.

Received: 7 February 2022; Accepted: 31 May 2022;
Published online: 21 June 2022

References

1. Lauga, E. & Powers, T. R. The hydrodynamics of swimming microorganisms. *Rep. Prog. Phys.* **72**, 096601 (2009).
2. Berg, H. C. & Brown, D. A. Chemotaxis in *Escherichia coli* analysed by three-dimensional tracking. *Nature* **239**, 500–504 (1972).
3. Stocker, R., Seymour, J. R., Samadani, A., Hunt, D. E. & Polz, M. F. Rapid chemotactic response enables marine bacteria to exploit ephemeral microscale nutrient patches. *Proc. Natl Acad. Sci. USA* **105**, 4209–4214 (2008).
4. Xie, L., Altindal, T., Chattopadhyay, S. & lun Wu, X. Bacterial flagellum as a propeller and as a rudder for efficient chemotaxis. *Proc. Natl Acad. Sci. USA* **108**, 2246 – 2251 (2011).
5. Ipiña, E. P., Otte, S., Pontier-Bres, R., Czerucka, D. & Peruani, F. Bacteria display optimal transport near surfaces. *Nat. Phys.* **15**, 610–615 (2019).
6. Wan, K. Y. & Goldstein, R. E. Time irreversibility and criticality in the motility of a flagellate microorganism. *Phys. Rev. Lett.* **121**, 058103 (2018).
7. Tsang, A. C. H., Lam, A. T. & Riedel-Kruse, I. H. Polygonal motion and adaptable phototaxis via flagellar beat switching in the microswimmer euglena gracilis. *Nat. Phys.* **14**, 1216–1222 (2018).
8. Gao, W. et al. Cargo-towing fuel-free magnetic nanoswimmers for targeted drug delivery. *Small* **8**, 460–467 (2012).
9. Zhang, L. et al. Characterizing the swimming properties of artificial bacterial flagella. *Nano Lett.* **9**, 3663–3667 (2009).
10. Ghosh, A. & Fischer, P. Controlled propulsion of artificial magnetic nanostructured propellers. *Nano Lett.* **9**, 2243–2245 (2009).
11. Ceylan, H. et al. 3d-printed biodegradable microswimmer for theranostic cargo delivery and release. *ACS Nano* **13**, 3353–3362 (2019).
12. Huang, T.-Y. et al. 3d printed microtransporters: compound micromachines for spatiotemporally controlled delivery of therapeutic agents. *Adv. Mater.***27**, 6644–6650 (2015).
13. Nassif, X., Bourdoulous, S., Eugène, E. & Couraud, P.-O. How do extracellular pathogens cross the blood–brain barrier? *Trends Microbiol.* **10**, 227–232 (2002).
14. Celli, J. P. et al. Helicobacter pylori moves through mucus by reducing mucin viscoelasticity. *Proc. Natl Acad. Sci. USA* **106**, 14321–14326 (2009).
15. Mirbagheri, S. A. & Fu, H. C. *Helicobacter pylori* couples motility and diffusion to actively create a heterogeneous complex medium in gastric mucus. *Phys. Rev. Lett.* **116**, 198101 (2016).
16. Purcell, E. M. Life at low Reynolds number. *Am. J. Phys.* **45**, 3–11 (1977).
17. Hu, W., Lum, G. Z., Mastrangeli, M. & Sitti, M. Small-scale soft-bodied robot with multimodal locomotion. *Nature* **5554**, 81–85 (2016).
18. Ohm, C., Brehmer, M. & Zentel, R. Liquid crystalline elastomers as actuators and sensors. *Adv. Mater.* **22**, 3366–3387 (2010).
19. Dai, B. et al. Programmable artificial phototactic microswimmer. *Nat. Nanotechnol.* **11**, 1087–1092 (2016).
20. Palagi, S. et al. Structured light enables biomimetic swimming and versatile locomotion of photoresponsive soft microrobots. *Nat. Mater.* **15**, 647 (2016).
21. von Rohr, A., Trimpe, S., Marco, A., Fischer, P. & Palagi, S. Gait learning for soft microrobots controlled by light fields. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* 6199–6206 (IEEE, 2018).
22. Huang, H.-W., Sakar, M. S., Petruska, A. J., Pané, S. & Nelson, B. J. Soft micromachines with programmable motility and morphology. *Nat. Commun.* **7**, 12263 (2016).
23. Huang, H.-W. et al. Adaptive locomotion of artificial microswimmers. *Sci. Adv.* **5**, eaau1532 (2019).
24. Reddy, G., Celani, A., Sejnowski, T. & Vergassola, M. Learning to soar in turbulent environments. *Proc. Natl Acad. Sci. USA* **113**, E4877 – E4884 (2016).
25. Reddy, G., Wong-Ng, J., Celani, A., Sejnowski, T. J. & Vergassola, M. Glider soaring via reinforcement learning in the field. *Nature* **562**, 236–239 (2018).
26. Gazzola, M., Tchieu, A. A., Alexeev, D., de Brauer, A. & Koumoutsakos, P. Learning to school in the presence of hydrodynamic interactions. *J. Fluid Mech.* **789**, 726–749 (2016).
27. Biferale, L., Bonaccorso, F., Buzzicotti, M., Clark Di Leoni, P. & Gustavsson, K. Zermelo's problem: optimal point-to-point navigation in 2d turbulent flows using reinforcement learning. *Chaos* **29**, 103138 (2019).
28. Verma, S., Novati, G. & Koumoutsakos, P. Efficient collective swimming by harnessing vortices through deep reinforcement learning. *Proc. Natl Acad. Sci. USA* **115**, 5849–5854 (2018).
29. Jiao, Y. et al. Learning to swim in potential flow. *Phys. Rev. Fluids* **6**, 050505 (2021).
30. Cichos, F., Gustavsson, K., Mehlig, B. & Volpe, G. Machine learning for active matter. *Nat. Mach. Intell.* **2**, 94–103 (2020).
31. Tsang, A. C. H., Demir, E., Ding, Y. & Pak, O. S. Roads to smart artificial microswimmers. *Adv. Intell. Syst.* **2**, 1900137 (2020).
32. Colabrese, S., Gustavsson, K., Celani, A. & Biferale, L. Flow navigation by smart microswimmers via reinforcement learning. *Phys. Rev. Lett.* **118**, 158004 (2017).
33. Alageshan, J. K., Verma, A. K., Bec, J. & Pandit, R. Machine learning strategies for path-planning microswimmers in turbulent flows. *Phys. Rev. E* **101**, 043110 (2020).
34. Schneider, E. & Stark, H. Optimal steering of a smart active particle. *Europhys. Lett.* **127**, 64003 (2019).
35. Muiños-Landin, S., Fischer, A., Holubec, V. & Cichos, F. Reinforcement learning with artificial microswimmers. *Sci. Robot.* **6**, eabd9285 (2021).
36. Yang, Y., Bevan, M. A. & Li, B. Micro/nano motor navigation and localization via deep reinforcement learning. *Adv. Theory Simul.* **3**, 2000034 (2020).
37. Tsang, A. C. H., Tong, P. W., Nallan, S. & Pak, O. S. Self-learning how to swim at low Reynolds number. *Phys. Rev. Fluids* **5**, 074101 (2020).
38. Hartl, B., Hübl, M., Kahl, G. & Zöttl, A. Microswimmers learning chemotaxis with genetic algorithms. *Proc. Natl Acad. Sci. USA* **118**, e2019683118 (2021).
39. Najafi, A. & Golestanian, R. Simple swimmer at low Reynolds number: three linked spheres. *Phys. Rev. E* **69**, 062901 (2004).

40. Ledesma-Aguilar, R., Löwen, H. & Yeomans, J. A circle swimmer at low Reynolds number. *Eur. Phys. J. E* **35**, 1–9 (2012).
41. Avron, J. E., Kenneth, O. & Oaknin, D. H. Pushmepullyou: an efficient micro-swimmer. *New J. Phys.* **7**, 234 (2005).
42. Golestanian, R. & Ajdari, A. Stochastic low Reynolds number swimmers. *J. Phys. Condens. Matter* **21**, 204104 (2009).
43. Alouges, F., DeSimone, A., Giraldi, L. & Zoppello, M. Self-propulsion of slender micro-swimmers by curvature control: N-link swimmers. *Int. J. Nonlinear Mech.* **56**, 132–141 (2013).
44. Wang, Q. Optimal strokes of low reynolds number linked-sphere swimmers. *Appl. Sci.* **9**, 4023 (2019).
45. Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. Proximal policy optimization algorithms. Preprint at arxiv: 1707.06347 (2017).
46. Liu, Y., Zou, Z., Tsang, A. C. H., Pak, O. S. & Young, Y.-N. Mechanical rotation at low Reynolds number via reinforcement learning. *Phys. Fluids* **33**, 062007 (2021).
47. Happel, J. & Brenner, H. *Low Reynolds Number Hydrodynamics: with Special Applications to Particulate Media* (Noordhoff International Publishing, 1973).
48. Kim, S. & Karrila, S. J. *Microhydrodynamics: Principles and Selected Applications* (Dover, New York, 2005).
49. Dhont, J. *An Introduction to Dynamics of Colloids* (Elsevier, 1996).
50. Sutton, R. S. & Barto, A. G. *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, 1998).
51. Golestanian, R. & Ajdari, A. Analytic results for the three-sphere swimmer at low Reynolds number. *Phys. Rev. E* **77**, 036308 (2008).
52. Howse, J. R. et al. Self-motile colloidal particles: From directed propulsion to random walk. *Phys. Rev. Lett.* **99**, 048102 (2007).
53. Lobaskin, V., Lobaskin, D. & Kulić I. M. Brownian dynamics of a microswimmer. *Eur. Phys. J.: Spec. Top.* **157**, 149–156 (2008).
54. Dunkel, J. & Zaid, I. M. Noisy swimming at low Reynolds numbers. *Phys. Rev. E* **80**, 021903 (2009).

Acknowledgements

Funding support by the National Science Foundation (Grant Nos. 1830958 and 1931292 to O.S.P. and Grant Nos. 1614863 and 1951600 to Y.-N.Y.) is gratefully acknowledged. Y.-N.Y. acknowledges support from Flatiron Institute, part of Simons Foundation. A.C.H.T. acknowledges funding support from the Croucher Foundation. Z.Z. and O.S.P. acknowledge the use of computational resources at the WAVE computing facility (enabled by the E.L. Wiegand Foundation) at Santa Clara University. We also thank Yi Fang for useful discussion.

Author contributions

Z.Z., Y.L., Y.-N.Y., O.S.P., and A.C.H.T. designed research; Z.Z., Y.L., Y.-N.Y., O.S.P., and A.C.H.T. performed research; Z.Z., Y.L., Y.-N.Y., O.S.P., and A.C.H.T. analyzed data; and Z.Z., Y.L., Y.-N.Y., O.S.P., and A.C.H.T. wrote the paper.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s42005-022-00935-x.

Correspondence and requests for materials should be addressed to Y.-N. Young, On Shun Pak or Alan C. H. Tsang.

Peer review information *Communications Physics* thanks Giovanni Volpe and the other, anonymous, reviewers for their contribution to the peer review of this work.

Reprints and permission information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.