

Optimum control strategies for maximum thrust production in underwater undulatory swimming

Li Fu,¹ Sardor Israilov,² Jesús Sánchez-Rodríguez ,³ Christophe Brouzet ,² Guillaume Allibert,⁴ Christophe Raufaste ,^{2,5} and Médéric Argentina ²

¹*Laboratoire de Tribologie et Dynamique des Systemes, École Centrale de Lyon, CNRS, Lyon, France*

²*Université Côte d'Azur, CNRS, INPHYNI, 17 rue Julien Lauprêtre, 06200 Nice, France*

³*Departamento de Física Fundamental, Universidad Nacional de Educación a Distancia, 28040 Madrid, Spain*

⁴*Université Côte d'Azur, CNRS, I3S, Sophia Antipolis, France*

⁵*Institut Universitaire de France (IUF), Paris, France*



(Received 26 March 2024; accepted 12 February 2025; published 11 April 2025)

Fishes, cetaceans, and many other aquatic vertebrates undulate their bodies to propel themselves through water. Swimming requires an intricate interplay between sensing the environment, making decisions, controlling internal dynamics, and moving the body in interaction with the external medium. Within this sequence of actions initiating locomotion, biological and physical laws manifest complex and nonlinear effects, which do not prevent natural swimmers from demonstrating efficient movement. This raises two complementary questions: how to model this intricacy and how to abstract it for practical swimming. In the context of robotics, the second question is of paramount importance to building efficient artificial swimmers driven by digital signals and mechanics. In this study, we tackle these two questions by leveraging a biomimetic robotic swimmer as a platform for investigating optimal control strategies for thrust generation. Through a combination of machine learning techniques and intuitive models, we identify a control signal that maximizes thrust production. Optimum tail-beat frequency and amplitude result from the subtle interplay between the swimmer's internal dynamics and its interaction with the surrounding fluid. We then propose a practical implementation for autonomous robotic swimmers that requires no prior knowledge of systems or equations. Direct fluid-structure simulations confirm the effectiveness and reliability of the proposed approach. Hence, our findings bridge fluid dynamics, robotics, and biology, providing valuable insights into the physics of aquatic locomotion.

DOI: [10.1103/PhysRevFluids.10.043101](https://doi.org/10.1103/PhysRevFluids.10.043101)

I. INTRODUCTION

The diversity of shapes and physiologies among multicellular organisms is tremendous, and it can be rationalized by the principles of Darwinian evolution and the various functions required by animals [1]. Locomotion is a vital activity for metazoans, as it enables them to fulfill essential functions necessary for survival, such as accessing favorable environments, engaging in reproduction, hunting, and evading predators.

Published by the American Physical Society under the terms of the [Creative Commons Attribution 4.0 International](https://creativecommons.org/licenses/by/4.0/) license. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

From tadpoles of a few centimeters to whales of 20 meters in length, swimming consists of pushing the water by undulating the body [2] which produces a thrust exploiting the inertia of the displaced fluid [3]. The kinematics of underwater undulatory swimming appear to be particularly robust in vertebrates, highlighting general physical principles. The wavelength of the body deformation is of the order of the length of the swimmer [4,5], while there is on average a factor of 0.2 between tail beat amplitude and swimmer length [6–10]. The tail beat frequency f is not fixed for an individual but tunes its swimming speed: the higher the frequency, the higher the speed [3,6,9–11]. There is evidence that each swimmer can vary its frequency within a frequency band whose range is set by the interplay between the muscle properties and the interaction of the swimmer with its surrounding fluid [10]. Muscles have their own limits in terms of speed of contraction and tension, as represented by Hill’s muscle model [12]. Constraints can also be imposed by decision processes that are either spontaneous, through proprioceptive reflexes [13–15], or conscious, for instance, through the choice of the activity level [16,17]. These decisions drive the gait [5] and we can expect different control strategies for an individual swimming at burst speed [18] and the same swimmer exhibiting a swim-and-coast gait in a sustained level of activity [19].

These considerations are echoed in biomimetic robotic swimmers [15,20–23]. The biological nature of the internal dynamics is here replaced by robotic elements from electronics, mechanics, and/or computer science. Developing efficient and fully autonomous artificial swimmers requires finding the appropriate control strategies tailored to specific constraints to achieve optimal performance for a given action. Recent years have witnessed the development of various approaches to control soft robot fish, both in simulations and experiments. Classical control techniques such as PI [24], PID [25], and robust controllers [26] have been employed to improve trajectory tracking performance. The emergence of artificial intelligence algorithms has paved the way for novel control approaches designed specifically for soft swimmers. Models and simulations are now utilized to explore various machine learning algorithms, achieving high swimming speeds while maintaining trajectory accuracy [27–30]. Regardless of the methodology employed, the quest to achieve the highest swimming speed through robotic fish remains an enduring challenge in the domain of aquatic locomotion. While certain investigations have explored the correlation between swimming speed and tail beat frequency [21,30,31], none of these prior studies has offered a conclusive comprehension of the optimal control strategy required to attain this objective.

In this article, we present our findings on the most effective scheme for generating maximum thrust in swimming using a single-parameter control system. Initially, we showcase the experimental outcomes achieved by employing deep reinforcement learning (RL) techniques [32] to drive a swimming robot. Through these experiments, we identify the optimal control strategy, which involves utilizing a square wave function that alternates between the two extreme values allowed by the controller. To gain a deeper understanding of the underlying physics, we develop a theoretical model that further supports and explains the observed results. Subsequently, we rigorously validate our optimized thrust strategy through a comprehensive 2D numerical simulation. By aligning experiments, theory, and numerical simulations, we successfully determine the most efficient approach to achieve maximum swimming speed. This integrated approach provides valuable insights into the best methods for enhancing propulsion in aquatic systems.

II. RESULTS

A. Experiments and machine learning

We exploited the robotic experimental platform described in [15,33] and detailed in Appendix A. The robotic fish, shown in Fig. 1, consisted of a deformable skeleton with a fin attached to its end, fabricated through 3D printing using a flexible polymer. To achieve controlled deformation, a servomotor was employed, connected to the skeleton’s end via two cables. By rotating the wheel of the servomotor by an angle $\phi(t)$, we induced a deformation that controlled the fin angle $\alpha(t)$.

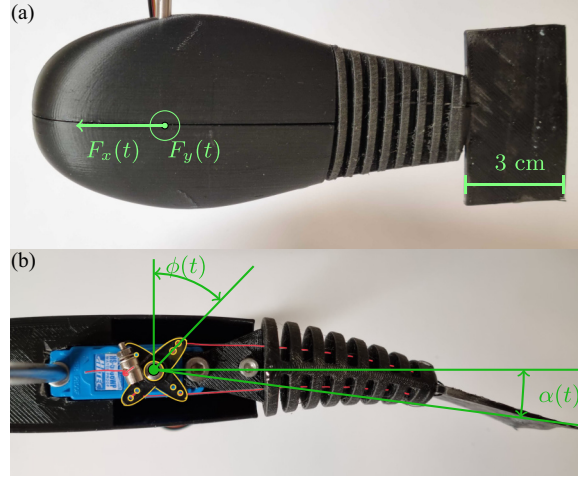


FIG. 1. (a) Side view of the robotic fish. (b) Top view without the top fairing. The head is attached to a dual-force sensor. The tail and the fin are printed with a flexible polymer. The body deformation is triggered by the actuation of a waterproof servomotor (blue): The rotation of its wheel pulls cables (drawn in red) to deform the elastic tail [33]. The contours of the servomotor wheel are drawn in yellow.

This design closely emulates the functioning of antagonistic muscles found in natural swimmers, responsible for initiating body deformations. The robot was immersed inside a water tank.

To enhance the quality of the learning process, its head was fixed to a force sensor that measured the longitudinal force $F_x(t)$. This sensor served as a measure of thrust and was, on average, positive when the fish was in the propulsion phase, since the thrust slightly depends on locomotion velocity [9,33]—primarily because the maximum angle of the fin remains small.

We employed an efficient RL algorithm to identify the optimal control signal that maximized the average thrust. In accordance with the specified details in Appendix B, we limited the instruction angle $\phi_c(t)$ of the servomotor to vary within the interval $[-\Phi, \Phi]$. The RL algorithm was designed to maximize a reward proportional to the thrust $F_x(t)$. The algorithm required inputs to characterize the state of the system at every time step. Two independent sets of inputs were considered consecutively to probe the robustness of the learning process and ensure convergence toward the optimal control signal.

First, the algorithm was provided with $\phi_c(t)$, $F_y(t)$, and its time derivative $\dot{F}_y(t)$ to account for the oscillatory nature of the system. After a few hours, the learning process converged toward a periodic motion. As shown in Fig. 2(a), the optimal control signal $\phi_c^*(t)$ consists of a square wave function that switches abruptly between the two allowed extreme values $\pm\Phi$.

Second, we provided images recorded by a camera, as shown in Fig. 1(b), as inputs instead of relying on lateral force measurements. At each time step, four images were used (i.e., the current image and the previous three) to calculate convergence and speed, details are given in Appendix C. Despite this change in inputs, the resulting optimal control signal remained consistent with the previous findings. That is, the algorithm converged to a square wave function with the same amplitude and frequency, reaffirming the robustness of the previously identified optimal control strategy.

The two approaches were then compared in terms of relevant physical quantities while varying the maximum instruction angle Φ . In Fig. 3, the maximized average thrust $\bar{F}_x^*$, the optimal frequency f^* , and the corresponding maximum fin angle $\alpha_{\max}^*$ are plotted as functions of Φ . Whatever Φ , the agreement is excellent between the two approaches for all quantities. Specifically, $\alpha_{\max}^*$ is proportional to Φ and the optimal frequency is a slowly decreasing function of Φ varying between 2.5 and 1 Hz in the parameter range considered.

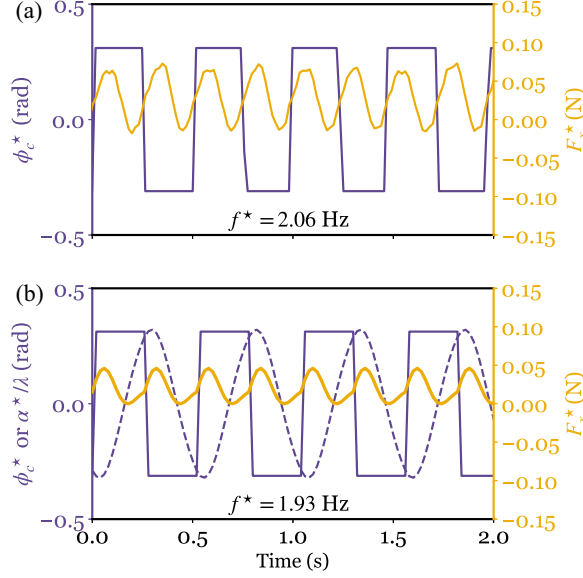


FIG. 2. (a) Experimental result: instruction angle $\phi_c^*(t)$ and $F_x^*(t)$ of the best strategy determined by the algorithm, for $\Phi = 0.31$ rad. (b) Results from the model: temporal evolution of the instruction angle $\phi_c^*(t)$ (purple solid line), the rescaled fin angle α^*/λ (purple dashed line) and the resulting thrust $F_x^*(t)$ (orange solid line), obtained for optimal control with the model and the parameters $\Omega = 5.8 \text{ rad s}^{-1}$, $\Delta = 0.29$ rad, $\omega_0 = 12.5 \text{ rad s}^{-1}$, $\xi = 1.2$, and $\Phi = 0.31$ rad.

To confirm that square wave forcing produces the greatest thrust, we applied three types of classical periodic forcing (sinusoidal, triangular, and square) for a given Φ and measured the averaged thrust as a function of forcing frequency, f . Results are shown in Fig. 4 for $\Phi = 0.42$ rad. Regardless of the type of actuation, the variation of the forcing frequency results in the presence of a peak in thrust, but the square wave forcing consistently generates the highest average thrust. This maximum thrust is achieved at a frequency of 1.8 Hz, a value that closely aligns with the frequency obtained using the RL approach.

The analysis thus reveals a robust feature of optimal control: a square wave function oscillating between two extreme values. This suggests the existence of a strong mechanism driving maximum thrust.

B. Model

We present a comprehensive model that combines the physical aspects of underwater undulatory swimming with the internal dynamics of the servomotor to gain insights into the experimentally discovered optimal solutions.

The dynamics of the fin angle $\alpha(t)$ can be described by a damped harmonic oscillator, influenced by the angle $\phi(t)$ of the servomotor wheel [15]:

$$\ddot{\alpha}(t) + \xi\omega_0\dot{\alpha}(t) + \omega_0^2(\alpha(t) - \alpha_c(t)) = 0, \quad \alpha_c(t) = \lambda\phi(t). \quad (1)$$

This equation captures the interaction between the deformable fin and the surrounding water, driven by an instruction angle, $\alpha_c(t)$, that is proportional to the servomotor wheel angle, $\phi(t)$. The numerical values of the parameters were determined following the procedure described in [15]. The proportionality factor $\lambda = 0.46$ depends on the length of the cables and the elasticity of the polymeric skeleton. Additionally, the servomotor has its own internal dynamics to adjust the

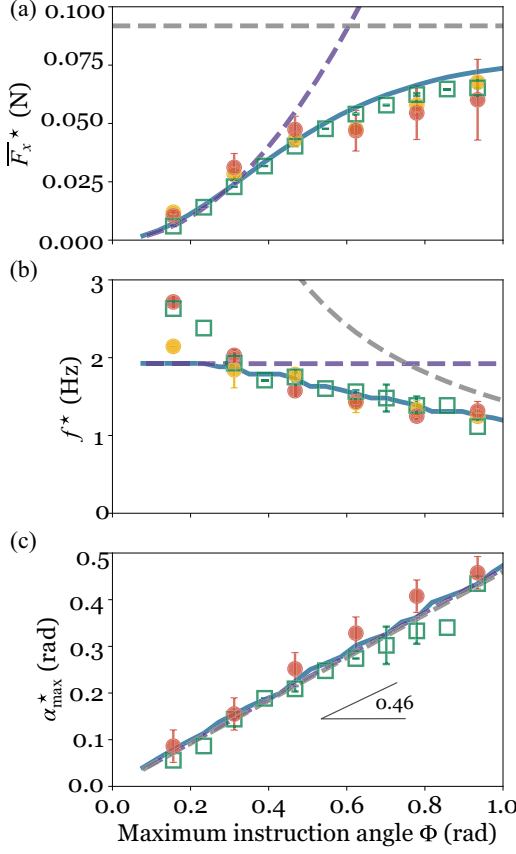


FIG. 3. (a) Maximum average thrust $\overline{F}_x^*$. (b) Optimal undulation frequency f^* . (c) Maximal fin angle $\alpha_{\max}^*$. For each figure, we compare the experimental results and those obtained with the model [Eqs. (1)–(3)]. Experimental results: The red symbols are obtained via RL with the state $(F_y, \dot{F}_y, \phi_c)$; the orange symbols are obtained via image learning, where the state is a set of four successive pictures of the robot. Simulation results: The open square symbols denote RL results with the simulation of the model. The dashed violet and gray curves represent theoretically predicted values in the limits of very small and very large Φ , respectively. The solid green curve corresponds to the numerical determination of the frequency f^* that yields the highest thrust, F_x^* , assuming a square waveform for ϕ_c .

servomotor wheel angle $\phi(t)$ to the instruction angle $\phi_c(t)$ [15]:

$$\dot{\phi}(t) = \Omega \tanh\left(\frac{1}{\Delta}(\phi_c(t) - \phi(t))\right). \quad (2)$$

The model includes the parameters $\Omega = 5.8 \text{ rad s}^{-1}$ and $\Delta = 0.29 \text{ rad}$, which represent the maximum angular speed of the wheel and the required angle difference for the servomotor to operate at its maximum angular speed, respectively. The nonlinearity introduced by the tanh function results in the saturation of the wheel speed $\dot{\phi}(t)$ at $\pm\Omega$ if the servomotor is unable to keep up with the provided instruction. This saturation occurs when the servomotor is too slow to reach the desired angle $\phi_c(t)$.

Equations (1) and (2) correspond to two essential aspects of the dynamics: One pertains to the interaction between the undulating swimmer and its aquatic environment, while the other accounts for the limitations imposed by the swimmer's internal dynamics. Our experimental system thus

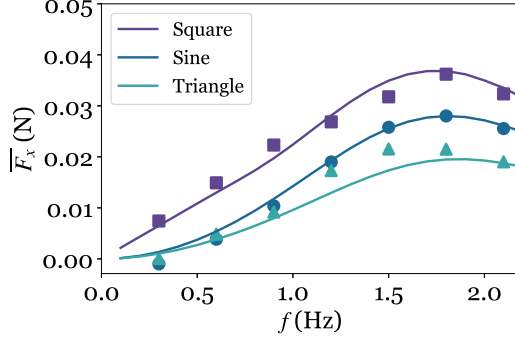


FIG. 4. Influence of the frequency on the thrust produced by some periodic functions: square (purple squares), sinusoidal (blue circles), and triangular (green triangles) waves. Filled symbols and solid lines correspond to the experimental points and the model predictions, respectively. $\Phi = 0.42$ rad.

serves as a suitable candidate for mimicking the dual nature of dynamics observed in natural fish. In natural fish, the internal dynamics are limited by biological constraints at the muscular level [10].

The thrust is proportional to the mass of water accelerated in the longitudinal direction and is written $F_x = -K\alpha(t)\ddot{\alpha}(t)$ [15,34], where the parameter $K = 12.9 \cdot 10^{-3} \text{ N}\cdot\text{rad}^{-2}\cdot\text{s}^2$ characterizes the thrust efficiency of the robot in water and is measured following the procedure described in [15]. The average thrust over an undulation period $T = 1/f$ can be written as

$$\overline{F}_x = -\frac{K}{T} \int_0^T \alpha(t)\ddot{\alpha}(t)dt = \frac{K}{T} \int_0^T \dot{\alpha}(t)^2 dt. \quad (3)$$

We employ RL techniques on the model to verify the optimal control strategy that yields the maximum average thrust. In Fig. 2(b), we show the time evolution of the optimal instruction angle $\phi_c^*(t)$ for $\Phi = 0.31$ rad, alongside the corresponding rescaled fin angle $\alpha^*(t)/\lambda$ and thrust $F_x^*(t)$. Consistent with the experimental results, the optimal instruction angle adopts a square wave function, alternating between Φ and $-\Phi$.

In fact, the dynamics presented in Figs. 2(a) and 2(b), measured in experiments and simulated with the model, are very similar. This is confirmed by the excellent agreements shown in Figs. 3(b) and 3(c) for the relevant quantities such as undulation frequency and maximum fin angle of the optimal control. Furthermore, this simple model is able to reproduce the average thrust force, as depicted in Fig. 3(a).

Regardless of the specific value of Φ , both the experiments and the model demonstrate that the system waits for the fin angle to approach its instruction before changing the control. In other words, the control signal switches direction when the fin angle is close to its maximal value $\alpha_{\max}^* = \lambda\Phi$. This finding suggests an intuitive mechanism for selecting the frequency simultaneously with the control switch, emphasizing the efficiency and adaptability of the system in achieving maximum thrust.

The model is also validated with the data obtained in Fig. 4 in which sinusoidal, triangular, and square wave forms were enforced. The agreement between the experiments and the model is again excellent. The maximum thrust is generated around a frequency of 1.8 Hz, indicating that the system operates optimally near its resonance in $\dot{\alpha}$, which occurs at a frequency $\omega_0/(2\pi) \sim 2.0$ Hz, regardless of the damping factor. For comparison, the resonance in α suggested by various other studies [35–37] occurs at a lower value, $\omega_0\sqrt{1-\xi^2}/(2\pi) \simeq 1.1$ Hz in our system.

C. Bang-bang control

The periodic and abrupt changes in the instruction angle resemble the behavior of a bang-bang controller [38]. In the context of the the model, we apply Pontryagin’s maximum principle [39]

to explain why square wave control at maximum allowed values $\pm\Phi$ in the instruction angle ϕ_c produces the maximum thrust $\overline{F}_x$, defined in Eq. (3), under the constraints of Eqs. (1) and (2).

The determination of the maximum thrust thus requires the resolution of a variational problem subject to three Lagrange multipliers, associated with the dynamic equations for $\alpha(t)$, $\dot{\alpha}(t)$, and $\phi(t)$. The bang-bang controller provides the optimal control if the response is driven by linear equations [40]. This condition is not completely satisfied in this system due to the nonlinear behavior of the servomotor's internal dynamics, Eq. (2). However, a bang-bang controller remains the optimal choice in this system because of the specific nature of the nonlinear function driving the relaxation dynamics of ϕ (details in Appendix D).

This appears particularly simple to explain for a fast servomotor, i.e., if $\phi(t)$ almost immediately follows the command $\phi_c(t)$. In our case, we can assimilate $\phi(t)$ to $\phi_c(t)$ for $\Phi \ll \Phi_s$, with

$$\Phi_s = \frac{\Omega}{\omega_0}, \quad (4)$$

as shown in Appendix E. In both limits, the forcing command acts linearly through Eq. (1), which completely justifies the choice of a bang-bang controller as an optimal strategy. In addition, this asymptotic analysis permits the computation of the average thrust resulting from a square wave driving at frequency f :

$$\overline{F}_x = K\lambda^2\Phi^2\omega_0^2 \frac{\frac{4f}{\xi\omega_0} \left(\sinh\left(\frac{\xi\omega_0}{4f}\right) - \frac{\xi \sinh\left(\sqrt{\xi^2-4}\frac{\omega_0}{4f}\right)}{\sqrt{\xi^2-4}} \right)}{\cosh\left(\sqrt{\xi^2-4}\frac{\omega_0}{4f}\right) + \cosh\left(\frac{\xi\omega_0}{4f}\right)}, \quad (5)$$

where $K\lambda^2\Phi^2\omega_0^2$ is the dimensional value that gives the scaling of the average thrust in this limit (see Appendix F for details). Regardless of the value of ξ between 0 and 2, this thrust is maximum when α is resonant and the oscillator is driven close to its undamped frequency: $f^* \simeq \omega_0/(2\pi)$.

For the case of a slow servomotor with $\Phi \gg \Phi_s$, the system of equations is nonlinear, and the maximum thrust can be expressed as

$$\overline{F}_x = K\lambda^2\Omega^2, \quad (6)$$

which is reached as long as the servomotor wheel is moving at maximum angular speed Ω [i.e., $\dot{\alpha}(t) = \pm\lambda\Omega$ in Eq. (3)]. The square wave forcing at maximum allowed values appears as the optimal solution to maximize the difference between $\phi_c(t)$ and $\phi(t)$ in Eq. (2) and ensures that the servomotor wheel is moving at maximum angular speed. In this limit, the optimal undulation period is determined by the shortest time required to sweep the servomotor angle from $-\Phi$ to Φ and then back to $-\Phi$, all at maximum angular speed. This leads to the expression $f^* = \Omega/(4\Phi)$.

The behavior resulting from both limits is depicted in Fig. 3, and agrees with the outcomes obtained in experiments and with the model, as well as indicating a transition for Φ around Φ_s .

D. Swinging control: a model-free strategy

In the above model, to maximize thrust, the optimal frequency must be found in order to select the best square wave function for control. It is therefore necessary to perform preliminary calibrations to measure the relevant quantities, in this case ω_0 , ξ , Ω , and Δ , to bring a particular system closer to optimality. Here, we explore another strategy that does not require any prior knowledge about the system.

Starting from the thrust expression, Eq. (3), the optimization is intrinsically linked to the time evolution of the fin angle velocity $\dot{\alpha}$. An intuitive approach would be to favor the phases of maximum speed by reversing the sign of the instruction angle when the fin slows down too much. We call this swinging control in reference to the way children are able to swing without knowing the physical laws involved.

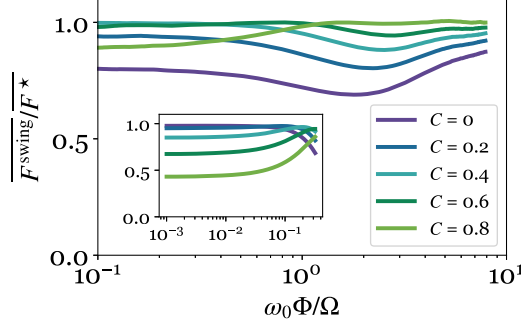


FIG. 5. Performance of the swinging strategies compared to the optimal solutions as a function of $\Phi/\Phi_s = \omega_0\Phi/\Omega$, with different instruction-changing criteria. We fixed $\omega_0 = 12.5 \text{ rad s}^{-1}$, $\xi = 1.2$, $\Omega = 5.8 \text{ rad s}^{-1}$, and $\Delta = 0.29 \text{ rad}$. Inset: Performance of the swinging strategies with the same instruction-changing criteria for a fast servomotor $\omega_0\Phi/\Omega < 1$ and $\xi = 0.05$.

To this end, we introduced an instruction-changing criterion C ($0 \leq C \leq 1$), such that ϕ_c changes sign if the speed slows to $C\dot{\alpha}_M$, with $\dot{\alpha}_M$ the last observed maximal angle speed. This condition ensures that the highest possible speed is maintained and that the instruction is changed when the speed becomes too slow. We found numerically that this strategy is always independent of the system history and the initial conditions, as the algorithm converges toward a unique solution.

We evaluated the average thrust resulting from the swinging strategy by varying the parameters $\omega_0\Phi/\Omega$, which measures the servomotor's ability to follow the instruction, and C . Figure 5 shows that a criterion value of C ranging from 0 to 0.8 yields satisfactory performance for the swinging strategy whatever the value of $\omega_0\Phi/\Omega$. The swinging control consistently delivers excellent outcomes without imposing stringent constraints on the choice of C : the thrust efficiency, defined as $\overline{F^{\text{swing}}}/\overline{F^*}$, being higher than 70% in the parameter range. For $C = 0.6$, this ratio exceeded 95% regardless of the value of $\omega_0\Phi/\Omega$, but fine-tuning remains possible. This strategy can be proposed even in specific cases: in the example of a fast servomotor ($\Phi \ll \Phi_s$) and an undamped oscillator ($\xi \rightarrow 0$), the optimum is attained when the instruction angle changes sign at $\dot{\alpha} = 0$ (i.e., $C = 0$) with a thrust efficiency very close to 100%, as shown in the inset of Fig. 5, and detailed in Appendix G and in [41].

Swinging control is a robust strategy for achieving the highest thrusts without prior knowledge of the system or complex control algorithms. Its straightforward implementation using basic sensors makes it practical for various applications.

E. Direct numerical simulations

To extend our results to free-swimming conditions, we conducted simulations that accounted for the complete fluid-structure interaction. For better convergence, we used 2D configurations, which realistically capture the leading-order dynamics of natural swimmer locomotion [3]. In this numerical setup, the swimmer moves within a tank filled with liquid having the same density and viscosity as water. We consider the fish body to be viscoelastic, and we adjust its parameters to match the values of ω_0 and ξ obtained from the robotic fish experiments. The entire body measures $L = 10 \text{ cm}$ and $H = 4 \text{ cm}$, which corresponds to an averaged thickness $= 1.3 \text{ cm}$. To model muscular activity, we impose a spatiotemporal variation of the elastic body length at equilibrium. In particular, we set that the equilibrium of the strain component ϵ_{xx} varies parabolically as a function of the distance to the head X , and linearly from the midline Y : $\epsilon_{xx} \propto (X/L)^2(Y/H)a(t)$, where $a(t)$ drives the swimmer deformation and varies temporally. With this functional form, the half body ($Y > 0$) extends its length while the other part ($Y < 0$) retracts, resulting in the swimmer bending to compensate for the inhomogeneous change in length across the body thickness. We have simulated

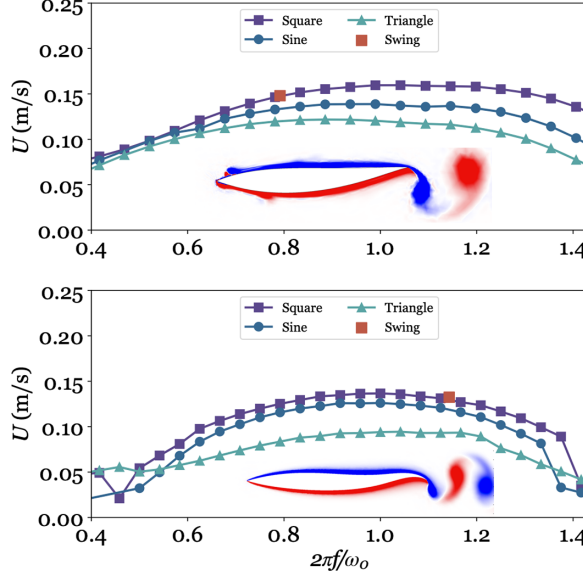


FIG. 6. Computation of the swimming speed using a full 2D fluid-structure interaction simulation. The solid lines and symbols represent the forcing with different wave shapes and the red square corresponds to the swinging controller. (a) Results obtained for a thunniform gait. (b) Eel-like swimming. For both swimmers, we show the close vorticity field.

the motion of this 2D active elastic beam embedded in water, driven by different functional forms of $a(t) \in [-1, 1]$. To check the generality of our results, we also simulated an eel-like elongated body, whose geometry is detailed in the Appendix H.

In Fig. 6, we present the cruising speed achieved with square, sine, and triangular wave functions at various control frequencies (f) for both tuna and eel-like bodies. The square wave forcing consistently leads to the highest speed, regardless of the frequency, and the morphology, confirming the predictions from the RL algorithm and the model.

In addition, we have implemented the swinging strategy with $C = 0$. The swinging controller automatically selects a frequency that propels the swimmer to a speed close to the maximum, as depicted in Fig. 6. Therefore, our numerical simulations validate our interpretations regarding the mechanisms to achieve the highest swimming speed.

III. DISCUSSION

Our study examines how different actuation patterns influence the kinematics of undulatory swimming, which is a key factor in determining propulsion efficiency and thrust production. Prior research, such as Borazjani and Sotiropoulos [42], has extensively analyzed the role and the importance of kinematics in optimizing swimming performance. We observe that different actuation strategies lead to distinct kinematic profiles and efficiencies. This aligns with previous studies that have demonstrated how wave amplitude plays a crucial role in determining swimming performance [43–46].

In this article, our primary objective was to determine the actuation that yields the highest thrust, and thus the highest swimming speed, for underwater swimmers. Considering the complex fluid-structure interaction at relatively high Reynolds numbers, we employed various approaches and methods to shed light on this topic. Notably, reinforcement learning techniques played a crucial role in revealing that the most effective method to achieve high thrust in experiments involves employing a bang-bang controller. This type of actuation oscillates abruptly and periodically between the two

extreme command values. To rationalize and validate these experimental results, we studied a simple model using reinforcement learning. The results from this model closely aligned with those obtained through the experimental approach, further reinforcing the effectiveness of the bang-bang controller. Additionally, full 2D numerical simulations of autonomous swimmers confirmed the validity of our findings.

We have successfully illuminated the strategy to attain the highest thrust for underwater swimmers, comprising three key messages. First, the command should resemble a periodic square function, exploiting the natural oscillation of the tail-fin system. Second, the frequency of actuation needs to be close to that which maximizes tail speed. If damping vanishes, this frequency will approach the body's natural undulation frequency. Finally, an efficient approach to selecting a nearly optimal swimming gait is to switch the actuation sign as the fin speed becomes too small.

In the experiments, we constrained the robotic fish to be attached. This approach reduces the number of degrees of freedom and enhances the quality of the learning process. In contrast, a free-swimming robot would face challenges related to drift and buoyancy, which could significantly disrupt the search for the optimal solution during the two-hour training process. While our tethered setup focuses on revealing the physical laws governing the system, achieving optimality for a free-swimming robot will be the next step for new applications in the context of autonomous underwater vehicles. Nevertheless, maximizing thrust is effectively equivalent to maximizing speed at leading order, as all aquatic swimmers adhere to the same scaling laws, arising from a balance between thrust and drag, which can be evaluated separately [3,33]. Despite the diversity of gaits observed in fish, with body and fin movements varying from anguilliform to ostraciiform swimmers, these scaling laws consistently apply across different species, from tadpoles to whales. The same laws hold for the robotic fish and the principles used to maximize thrust in tethered conditions can confidently be applied to a free-swimming robot.

The rhythmic control strategies we employ—such as the square wave form—mirror the function of central pattern generators (CPGs) in animals, which are known to coordinate locomotion across a wide variety of species and morphologies [47]. By demonstrating that our control strategy is effective across different robotic morphologies, including thunniform and eel-like body shapes, we show how such a rhythm-based control system can adapt to different forms of locomotion, much like CPG-based control does in animals. Moreover, similar to how proprioceptive and mechanosensory feedback refines motor actions in animals, our system integrates feedback from the robot's sensors to optimize performance in real time. This reflects a key aspect of neuromechanics, where the interaction between sensory input and control outputs is critical for robust and adaptive locomotion. Finally, our approach enlances the concept of 'embodiment' found in neuromechanical systems, wherein the physical structure and passive biomechanics of the robotic swimmer directly influence the effectiveness of locomotion [48]. The interplay between control signals and the robot's physical environment allows our system to achieve optimal performance, echoing the dynamics observed in biological swimmers.

While our study focuses on burst locomotion, aimed at maximizing thrust and swimming speed, it would be interesting for future work to consider frugal locomotion. In that case, optimality in terms of power consumption would need to be explored, for example, by minimizing quantities such as the cost of transport [49]. This approach could also help explain intermittent regimes observed in nature, such as burst-and-coast behavior [19].

Given the simplicity and practicality of our results, we envision powerful applications in terms of underwater biomimetic swimmers. Furthermore, these findings can be utilized to develop strategies that enhance the performance of human athletes, particularly those involved in aquatic disciplines. The potential for real-world implementation is promising, opening up exciting avenues for both biomimetic technology and athletic performance improvement.

ACKNOWLEDGMENTS

We thank F. Boyer for very interesting comments. This work has been supported by UCAJEDI, UCA NIDS (Ref. No. ANR-15-IDEX-01), UCA DS4H, (Ref. No. ANR-17-EURE-0004), Ministerio de Universidades, and European Union - NextGenerationEU.

APPENDIX A: EXPERIMENTAL SETUP

To measure force, a force sensor is linked to the robot fish via an aluminum rod, enabling bidirectional force measurements (longitudinal force, F_x , and normal force, F_y) with a precision of approximately 10^{-3} N. An analog force signal is converted into a digital format using an ADC converter (Adafruit 1115) and then collected by the Raspberry Pi via the I2C interface.

The robotic swimmer is positioned in a water tunnel (Rolling Hills Research Corporation, Model 0710) using a beam clamp that holds the aluminum rod. In this experimental study, our focus is on determining the highest thrust generated by the oscillation of the swimmer's tail; hence, no water flow was generated by the tunnel pump.

For the learning process based solely on force sensor measurements, we conducted it directly on the Raspberry Pi. However, for the image-based learning process, we utilized an external computer due to computational constraints. Data exchange between the Raspberry Pi and the computer occurred via the TCP/IP protocol using an RJ45 Ethernet cable.

APPENDIX B: MACHINE LEARNING WITH FORCE SENSOR

In our search for the best functional form for servomotor driving, we employed deep reinforcement learning (RL) techniques. Within this framework, we described the control of the swimmer using Markov decision processes (MDP).

The state of the swimmer, denoted by $s = (F_y(t), \dot{F}_y(t), \phi_c(t))$, is characterized by the normal force, $F_y(t)$, and its derivative, $\dot{F}_y(t)$, as well as the command angle of the servomotor, $\phi_c(t)$. The action, represented by $\phi_c(t) \in [-\Phi, \Phi]$, refers to the command angle of the servomotor, and the reward is directly proportional to the thrust force, $F_x(t)$. The objective of the RL algorithm is to maximize the total cumulative reward, which, in this context, translates to maximizing the thrust, F_x , generated by the swimmer.

To achieve this, the RL algorithm explores the available action and state spaces and gradually converges to the best control sequence through a trial and error process. We exploit the PPO (proximal policy optimization) algorithm [50] for this study because it is capable of handling both discretized and continuous action spaces. PPO is a policy gradient method in RL that stabilizes the learning process to avoid large excursions.

In our experiments, RL applies the control policy and sends a command to the robotic fish every 50 ms. The control policy is incrementally updated (trained) after each 768 control steps. We chose this number to ensure sufficient exploration of the action and state spaces before each update. The entire training process spans 10^5 control steps, which amounts to approximately 2 hours. Throughout this process, we save the control policy and the state value function every 5000 steps. The hyperparameters values are given in [41].

After each training of the neural network, we conduct an inference to evaluate the quality of the actual best control policy: only the best action is chosen at each step during this process. This evaluation provides valuable insights into the performance of the swimmer with the optimized control sequence.

APPENDIX C: MACHINE LEARNING WITH IMAGES

In our study, we employed a web camera (ODROID USB-CAM 720P) placed above the robotic fish's undulating fin to record its motion. Instead of the previous state representation ($F_y(t), \dot{F}_y(t), \phi_c(t)$), we used a stack of four consecutive images separated by the sampling time

interval [51], to represent the state, s , of the robotic fish. The recorded RGB images were preprocessed by converting them to grayscale and rescaling them from their original size to 84×84 pixels.

This resizing was performed to reduce the computation time, while maintaining sufficient information. We used a convolutional neural network (CNN), as introduced by [52,53], as an image compression tool for the raw data of the images ($84 \times 84 \times 4$ pixels). The output of the CNN is a latent vector with 512 dimensions that is then passed to the fully connected neural networks to perform the actor-critic algorithm. Beyond this change in state representation, the remaining experimental methods for learning remained the same as those used for learning from the force sensor.

APPENDIX D: WHY A BANG-BANG CONTROLLER IS OPTIMAL

Following the Pontryagin principle [39], we aim to determine the best command to optimize the dimensionless force, averaged for a duration T :

$$I = \frac{\overline{F_x}}{K\Lambda^2\omega_0^2} = \frac{1}{\Lambda^2\omega_0^2} \frac{1}{T} \int_0^T \dot{\alpha}^2 dt, \quad \Lambda = \lambda\Phi,$$

under the dynamics given by Eqs. (1) and (2) written as first-order ODEs

$$\begin{aligned} \dot{\alpha} &= \omega \\ \dot{\omega} &= -\xi\omega_0\omega - \omega_0^2(\alpha - \alpha_c), \quad |\alpha_c| \leq \Lambda, \quad \alpha_c = \lambda\phi \\ \dot{\phi} &= \Omega \tanh \frac{\phi_c - \phi}{\Delta}. \end{aligned}$$

Within the variational methods framework, we write the Lagrangian

$$\begin{aligned} L &= \omega^2 + p_1(\dot{\alpha} - \omega) + p_2(\dot{\omega} + \xi\omega_0\omega + \omega_0^2(\alpha - \lambda\phi)) \\ &\quad + p_3\left(\dot{\phi} - \Omega \tanh \frac{\phi_c - \phi}{\Delta}\right), \end{aligned}$$

where we have introduced three Lagrangian multipliers, $p_{i=1..3}$ to impose the dynamics of α , ω , and ϕ . We reformulate this problem using the natural Hamiltonian

$$\begin{aligned} H &= -L + p_1\dot{\alpha} + p_2\dot{\omega} + p_3\dot{\phi} \\ H &= -\omega^2 + p_1\omega + p_2(-\xi\omega_0\omega - \omega_0^2(\alpha - \lambda\phi)) \\ &\quad + p_3\Omega \tanh \frac{\phi_c - \phi}{\Delta}. \end{aligned} \tag{D1}$$

The costate equations for this system are

$$\begin{aligned} \dot{p}_1 &= -\partial_\alpha H = \omega_0^2 p_2 \\ \dot{p}_2 &= -\partial_\omega H = 2\omega - p_1 + \xi\omega_0 p_2 \\ \dot{p}_3 &= -\partial_\phi H = -\omega_0^2 \lambda p_2 + p_3 \frac{\Omega}{\Delta} \cosh^{-2} \frac{\phi_c - \phi}{\Delta}. \end{aligned}$$

Following the Pontryagin seminal idea, the value of the control parameter ϕ_c is chosen to minimize the Hamiltonian value, to maximize the integral I . The bang-bang controller is the optimal command if the fourth term in the Hamiltonian (D1) which contains ϕ_c is linear, which is not the case here. Nevertheless, a square forcing for $\phi_c = \pm\Phi$ will minimize the Hamiltonian (D1): If $p_3 > 0$, we choose $\phi_c = -\Phi$, and $\phi_c - \phi$ will be negative. On the contrary, if $p_3 < 0$, we choose $\phi_c = \Phi$, and $\phi_c - \phi$ will be positive. Minimizing the Hamiltonian is therefore acquired by imposing

$$\phi_c = -\Phi \operatorname{sign}(p_3), \tag{D2}$$

and we recover a bang-bang controller. Hence, the optimal command for achieving the highest possible thrust is a periodic step function.

APPENDIX E: CONDITIONS FOR ENSURING THAT A SERVOMOTOR WILL PERFECTLY FOLLOW THE COMMAND

In this section, we perform a dimension analysis to evaluate the rapidity of the servomotor. We make the following change of variables:

$$\begin{aligned} s &= \omega_0 t \\ \alpha(t) &= \lambda \Phi \beta(s) \\ \phi(t) &= \Phi \psi(s), \end{aligned}$$

and we get the system to solve

$$\begin{aligned} \ddot{\beta}(s) + \xi \dot{\beta}(s) + \beta(s) - \psi(s) &= 0 \\ \dot{\psi}(s) &= \frac{\Omega}{\Phi \omega_0} \tanh \left(\frac{\Phi}{\Delta} (\psi_c(s) - \psi(s)) \right), \end{aligned}$$

which uncovers the three dimensionless parameters that govern the dynamics.

- (I) The parameter ξ determines the Q factor of the oscillator in $\alpha(t)$ or $\beta(s)$.
- (II) The parameter $\frac{\Omega}{\Phi \omega_0}$ controls the capacity of the servomotor to perfectly follow the command $\psi_c(s)$. If $\frac{\Omega}{\Phi \omega_0} \gg 1$, we can eliminate adiabatically the dynamics of $\psi(s)$: In this limit, we assume that $\phi(t) = \phi_c(t)$. This corresponds to the definition of a fast servomotor.
- (III) The parameter Φ/Δ measures the nonlinear response of the servomotor.
 - (1) if $\Phi/\Delta \gg 1$, the function $\tanh$, can be replaced by the function sign and because of the relaxational dynamics, the servomotor equation is reduced to

$$\dot{\phi}(t) = \Omega \text{sign} \phi_c(t).$$

- (2) If $\Phi/\Delta \ll 1$, the equation in ψ can be linearized, and we can write the equation:

$$\dot{\psi}(s) = \frac{\Omega}{\omega_0 \Delta} (\psi_c(s) - \psi(s)),$$

or in dimension variables

$$\dot{\phi}(t) = \frac{\Omega}{\Delta} (\phi_c(t) - \phi(t)).$$

Hence, the servomotor will perfectly follow the command if $\frac{\Omega}{\omega_0 \Delta} \gg 1$.

In conclusion, the servomotor can be considered as fast in two regimes: first if $\frac{\Omega}{\Phi \omega_0} \gg 1$, second if $\frac{\Omega}{\Delta \omega_0} \gg 1$, and $\Phi/\Delta \ll 1$.

APPENDIX F: COMPUTATION FOR A FAST ACTUATOR

In the limit of a fast servo motor, ϕ and therefore α_c follows adiabatically ϕ_c . Consequently, we set that the variable $\alpha_c \in [-\Lambda, \Lambda]$ is the control variable, where $\Lambda = \lambda \Phi$. In this limit, the dynamics

for the fin angle α is described with

$$\begin{aligned}\dot{\alpha} &= \omega \\ \dot{\omega} &= -\xi\omega_0\omega - \omega_0^2(\alpha - \alpha_c), \quad |\alpha_c| \leq \Lambda.\end{aligned}$$

While the Hamiltonian H from Eq. (D3) reduces to

$$H = -\omega^2 + p_1\omega + p_2(-\xi\omega_0\omega - \omega_0^2(\alpha - \alpha_c)). \quad (\text{F1})$$

We remark that $H = \alpha_c(\omega_0^2 p_2) + \text{others terms}$. Hence, by setting $\alpha_c = -\Lambda$, if $p_2 > 0$ and $\alpha_c = \Lambda$, if $p_2 < 0$, we impose that the Hamiltonian is always minimum. Consequently, if p_2 oscillates then the control α_c also oscillates between the two constant values' extreme values $\pm\Lambda$. The optimal command is therefore

$$\alpha_c = -\Lambda \text{sign}(p_2), \quad (\text{F2})$$

which corresponds to the classical bang-bang controller.

In the next part of this Appendix, we determine the dynamics of α . We impose that $\alpha_c = \Lambda$ for $t \in [0, T/2]$ and $\alpha_c = -\Lambda$ for $t \in [T/2, T]$. Solving the equation for the angle α in the first half period gives

$$\alpha = \Lambda + e^{-\frac{\xi\omega_0}{2}t}(a \cos vt + b \sin vt), \quad (\text{F3})$$

$$v = \frac{1}{2}\omega_0\sqrt{4 - \xi^2}, \quad (\text{F4})$$

where a and b are fixed by two boundary conditions. Applying the condition of continuity and differentiability,

$$\alpha(0) = -\alpha(T/2)$$

$$\dot{\alpha}(0) = -\dot{\alpha}(T/2),$$

the unknowns a and b are determined

$$\begin{aligned}a &= \frac{\Lambda\xi\omega_0 \sin\left(\frac{Tv}{2}\right) - 2\Lambda v \cos\left(\frac{Tv}{2}\right) - 2\Lambda v e^{T\xi\omega_0/4}}{2v(\cosh\left(\frac{1}{4}\xi T\omega_0\right) + \cos\left(\frac{Tv}{2}\right))} \\ b &= -\frac{\Lambda\xi\omega_0 \cos\left(\frac{Tv}{2}\right) + 2\Lambda v \sin\left(\frac{Tv}{2}\right) + 2\Lambda v e^{T\xi\omega_0/4}}{2v(\cosh\left(\frac{1}{4}\xi T\omega_0\right) + \cos\left(\frac{Tv}{2}\right))}.\end{aligned}$$

The averaged dimensionless thrust writes

$$I = -\frac{1}{\Lambda^2\omega_0^2} \frac{2}{T/2} \int_0^{T/2} \alpha \ddot{\alpha} dt, \quad (\text{F5})$$

$$= \frac{4}{\xi\omega_0 T} \frac{2 \sinh\left(\frac{1}{4}\xi T\omega_0\right) - \xi\omega_0/v \sin\left(\frac{Tv}{2}\right)}{\cos\left(\frac{Tv}{2}\right) + \cosh\left(\frac{1}{4}\xi T\omega_0\right)}. \quad (\text{F6})$$

The function I presents a maximum near $T = \frac{2\pi}{\omega_0}$.

This dimensionless thrust can be further optimized by computing the value T^* that renders it maximal. We remark, at this point, that the parameter T always appears multiplied by ω_0 in Eq. (F6), since v is proportional to ω_0 [as defined in Eq. (F4)]: I only depends on ξ and $\omega_0 T$.

APPENDIX G: ANALYSIS OF THE SWINGING STRATEGY

Here, we would like to measure the efficiency of the swinging strategy at least for $C = 0$, where C is defined in the main text. It consists of changing the sign of α_c as $\dot{\alpha}(t)$ zeroes. In the cruising

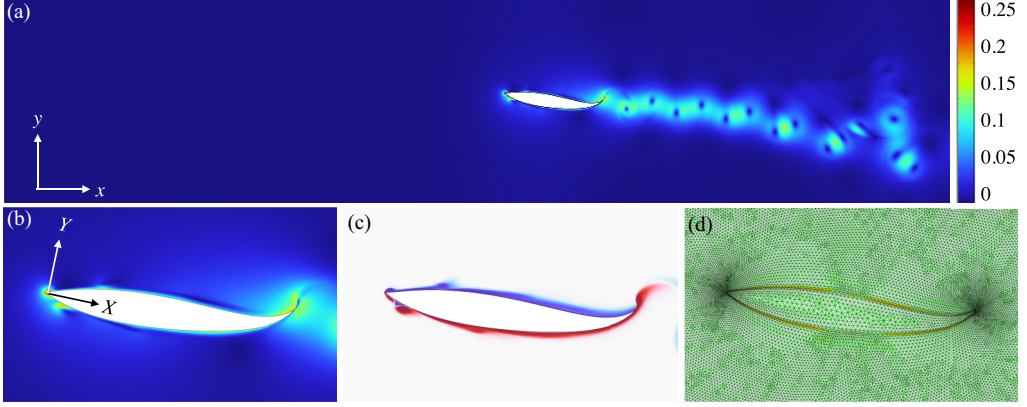


FIG. 7. (a) Setup of the computational domain. The white region represents the swimmer body, and the color codes the speed in m/s. (b) Zoom around the swimmer body. (c) Typical vorticity field around the swimmer. (d) Zoom around the swimmer to show the typical mesh precision used for the simulations.

regime, we have computed so far that this strategy predicts that $\dot{\alpha}(0) = \dot{\alpha}(T/2)$ should be zero. We therefore compute $\dot{\alpha}(0)$ to test the strategy:

$$\dot{\alpha}(0) = -\frac{2\Lambda\omega_0 \sin\left(\frac{1}{4}\sqrt{4-\xi^2}T\omega_0\right)}{\sqrt{4-\xi^2}\left(\cos\left(\frac{1}{4}\sqrt{4-\xi^2}T\omega_0\right) + \cosh\left(\frac{1}{4}\xi T\omega_0\right)\right)}.$$

This expression predicts that $\dot{\alpha}(0) = 0$ for $T = T_s$ and the thrust I_s :

$$T_s = \frac{4\pi}{\sqrt{4-\xi^2}\omega_0}, \quad (G1)$$

$$I_s = \frac{2\sqrt{4-\xi^2}}{\pi\xi} \coth\left(\frac{\pi\xi}{2\sqrt{4-\xi^2}}\right). \quad (G2)$$

APPENDIX H: FLUID-STRUCTURE INTERACTION SIMULATION

To simulate the fish swimming driven by an optimal command, we used the software COMSOL 6.1 following the approach outlined in [54]. The 2D computational domain covers a rectangle with dimensions $100 \times 20 \text{ cm}^2$, representing water.

In Fig. 7(a), we show the geometry of the simulation. The two horizontal borders represent slip boundary conditions for the velocity, while the left and right borders are associated with entrance and exit boundary conditions. The swimmer is approximated by a viscoelastic beam of Young's modulus 10^4 Pa , a Poisson ratio 0.3, and a viscosity 10^{-4} Pa.s . It navigates toward the left. The thickness $t(X)$ of the viscoelastic beam is given by the relation

$$t(X) = H \frac{X}{L} \left(1 - \frac{X}{L}\right) e^{-\frac{X}{L}},$$

where X is the curvilinear distance along the midline, measured from the head. Here, $H = 4 \text{ cm}$ and $L = 10 \text{ cm}$, such that the thickest part of the beam measures 1.3 cm , see Fig. 7(b). To model the antagonistic muscle action, we impose on the swimmer that the equilibrium component ϵ_{xx} of the strain varies spatiotemporally:

$$\epsilon_{xx}(X, Y, t) = 0.01(X/L)^2(Y/H)a(t),$$

where $a(t)$ drives the motion dynamics and ϵ is the strain tensor [55]. This forcing modifies the equilibrium length of each part of the body in an opposite manner: When the superior part of the

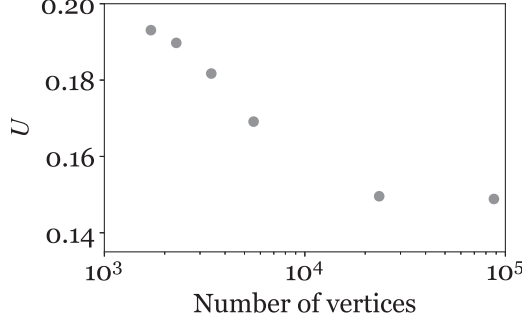


FIG. 8. Mesh refinement study. We plot the cruising velocity U as a function of the number of vertices of the mesh used for the 2D numerical simulations.

swimmer elongates its length, the other part contracts it, following the dynamics of $a(t)$. $a(t)$ can be a wave function or defined as $a(t) = \text{sign}(v(t))$, where $v(t)$ is the normal velocity of the swimmer at the tail. The numerical value is chosen such that the typical amplitude of the tail oscillation is close to 0.2.

In addition to the previous swimmer, that moves in a thunniform gait, we have also simulated a thinner active beam to mimic eel-like locomotion. For such a model, the thickness of the swimmer has been reduced to $H = 2$ cm, and the forcing term included was

$$\epsilon_{XX}(X, Y, t) = 0.01(X/L)^2(Y/H) \sin(qX)a(t),$$

where $q = 0.9 \, 2\pi/L$, as measured in experiments [5].

The fluid-structure problem is solved using a fully coupled approach and the PARDISO linear solver; the nonlinear problem is tackled with a Newton algorithm. Because the swimmer deforms its shape, the mesh is adapted using a Yeoh method. To avoid excessively large deformations in the mesh due to the swimmer's movement, the entire computational domain is remeshed if the discretization is too distorted. Approximately 7000 vertices are needed for almost 40 000 degrees of freedom to coarsely solve the complete fluid-structure interaction on the whole domain, and predict a correct swimming velocity. A finer element distribution is ensured at the head and tail [Fig. 7(d)]. However, to accurately capture the wake, 300 000 elements are required, as shown in Fig. 7(d). The typical time step used is 10^{-3} s. The center of mass of the swimmer is computed at each time step during the simulation. This comprehensive setup enables us to study and analyze the swimming behavior of the robotic fish driven by the optimal command obtained through the reinforcement learning process.

We have performed a mesh refinement study to gain confidence in the accuracy of the numerical prediction. In Fig. 8, we represent the cruising speed of a swimmer as a function of the number of vertices used for discretizing the fields: It appears that the use of more than 20 000 vertices is enough to accurately predict the swimming speed.

-
- [1] K. Healy, T. H. G. Ezard, O. R. Jones, R. Salguero-Gómez, and Y. M. Buckley, Animal life history is shaped by the pace of life and the distribution of age-specific mortality and reproduction, *Nat. Ecol. Evol.* **3**, 1217 (2019).
 - [2] S. Childress, *Mechanics of Swimming and Flying*, 2 (Cambridge University Press, Cambridge, 1981).
 - [3] M. Gazzola, M. Argentina, and L. Mahadevan, Scaling Macroscopic Aquatic Locomotion, *Nat. Phys.* **10**, 758 (2014).
 - [4] J. J. Videler, *Fish Swimming* (Chapman & Hall, London, 1993).

- [5] V. Di Santo, E. Goerig, D. K. Wainwright, O. Akanyeti, J. C. Liao, T. Castro-Santos, and G. V. Lauder, Convergence of undulatory swimming kinematics across a diversity of fishes, [Proc. Natl. Acad. Sci. USA](#) **118**, e2113206118 (2021).
- [6] R. Bainbridge, The speed of swimming of fish as related to size and to the frequency and amplitude of the tail beat, [J. Exp. Biol.](#) **35**, 109 (1958).
- [7] J. J. Rohr and F. E. Fish, Strouhal numbers and optimization of swimming by odontocete cetaceans, [J. Exp. Biol.](#) **207**, 1633 (2004).
- [8] J. R. Hunter, Swimming speed, tail beat frequency, tail beat amplitude and size in jack mackerel, *trachurus symmetricus*, and other fishes. [Fish. Bull.](#) **69**, 253 (1971).
- [9] M. Saadat, F. E. Fish, A. G. Domel, V. Di Santo, G. V. Lauder, and H. Haj-Hariri, On the rules for aquatic locomotion, [Phys. Rev. Fluids](#) **2**, 083102 (2017).
- [10] J. Sanchez-Rodriguez, C. Raufaste, and M. Argentina, Scaling the tail beat frequency and swimming speed in underwater undulatory swimming, [Nat. Commun.](#) **14**, 5569 (2023).
- [11] G. S. Triantafyllou, M. S. Triantafyllou, and M. A. Grosenbaugh, Optimal thrust development in oscillating foils with application to fish propulsion, [J. Fluids Struct.](#) **7**, 205 (1993).
- [12] A. V. Hill, The heat of shortening and the dynamic constants of muscle, [Proc. R. Soc. Lond. B](#) **126**, 136 (1938).
- [13] K. G. Pearson, Proprioceptive regulation of locomotion, [Curr. Opin. Neurobiol.](#) **5**, 786 (1995).
- [14] R. Williams IV, N. Neubarth, and M. E. Hale, The function of fin rays as proprioceptive sensors in fish, [Nat. Commun.](#) **4**, 1729 (2013).
- [15] J. Sánchez-Rodríguez, F. Celestini, C. Raufaste, and M. Argentina, Proprioceptive mechanism for bioinspired fish swimming, [Phys. Rev. Lett.](#) **126**, 234501 (2021).
- [16] J. R. Brett, The respiratory metabolism and swimming performance of young sockeye salmon, [J. Fish. Res. Bd. Can.](#) **21**, 1183 (1964).
- [17] J. R. Brett, The metabolic demand for oxygen in fish, particularly salmonids, and a comparison with other vertebrates, [Respir. Physiol.](#) **14**, 151 (1972).
- [18] M. R. Hirt, W. Jetz, B. C. Rall, and U. Brose, A general scaling law reveals why the largest animals are not the fastest, [Nat. Ecol. Evol.](#) **1**, 1116 (2017).
- [19] G. Li, I. Ashraf, B. François, D. Kolomenskiy, F. Lechenault, R. Godoy-Diana, and B. Thiria, Burst-and-coast swimmers optimize gait by adapting unique intrinsic cycle, [Commun. Biol.](#) **4**, 40 (2021).
- [20] V. Lebastard, F. Boyer, and S. Lanneau, Reactive underwater object inspection based on artificial electric sense, [Bioinspir. Biomim.](#) **11**, 045003 (2016).
- [21] J. Zhu, C. White, D. K. Wainwright, V. Di Santo, G. V. Lauder, and H. Bart-Smith, Tuna robotics: A high-frequency experimental platform exploring the performance space of swimming fishes, [Sci. Robot.](#) **4**, eaax4615 (2019).
- [22] R. Thandiackal, K. Melo, L. Paez, J. Herault, T. Kano, K. Akiyama, F. Boyer, D. Ryczko, A. Ishiguro, and A. J. Ijspeert, Emergence of robust self-organized undulatory swimming based on local hydrodynamic force sensing, [Sci. Robot.](#) **6**, eabf6354 (2021).
- [23] K. Y. Lee, S.-J. Park, D. G. Matthews, S. L. Kim, C. A. Marquez, J. F. Zimmerman, H. A. M. Ardoña, A. G. Kleber, G. V. Lauder, and K. K. Parker, An autonomously swimming biohybrid fish designed with human cardiac biophysics, [Science](#) **375**, 639 (2022).
- [24] F. Zhang, F. D. Lagor, D. Yeo, P. Washington, and D. A. Paley, Distributed flow sensing for closed-loop speed control of a flexible fish robot, [Bioinspir. Biomim.](#) **10**, 065001 (2015).
- [25] J. Yu, M. Tan, S. Wang, and E. Chen, Development of a biomimetic robotic fish and its control algorithm, [IEEE](#) **34**, 1798 (2004).
- [26] F. Zhang, O. Ennasr, E. Litchman, and X. Tan, Autonomous sampling of water columns using gliding robotic fish: Algorithms and harmful-algae-sampling experiments, [IEEE Syst. J.](#) **10**, 1271 (2016).
- [27] G. Novati, L. Mahadevan, and P. Koumoutsakos, Controlled gliding and perching through deep-reinforcement-learning, [Phys. Rev. Fluids](#) **4**, 093902 (2019).
- [28] Y. Jiao, F. Ling, S. Heydari, N. Heess, J. Merel, and E. Kansa, Learning to swim in potential flow, [Phys. Rev. Fluids](#) **6**, 050505 (2021).

- [29] S. K. Rajendran and F. Zhang, Design, modeling, and visual learning-based control of soft robotic fish driven by super-coiled polymers, *Front. Robot. AI* **8**, 809427 (2022).
- [30] S. M. Youssef, MennaAllah Soliman, M. A. Saleh, A. H. Elsayed, and A. G. Radwan, Design and control of soft biomimetic pangasius fish robot using fin ray effect and reinforcement learning, *Sci. Rep.* **12**, 21861 (2022).
- [31] B. P. Epps, P. Valdivia y Alvarado, K. Youcef-Toumi, and A. H. Teche, Swimming performance of a biomimetic compliant fish-like robot, *Exp. Fluids* **47**, 927 (2009).
- [32] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (MIT Press, Cambridge, MA, 2018).
- [33] F. Gibouin, C. Raufaste, Y. Bouret, and M. Argentina, Study of the thrust–drag balance with a swimming robotic fish, *Phys. Fluids* **30**, 091901 (2018).
- [34] M. Gazzola, M. Argentina, and Lakshminarayanan Mahadevan, Gait and speed selection in slender inertial swimmers, *Proc. Natl. Acad. Sci. USA* **112**, 3874 (2015).
- [35] S. Michelin and S. G. L. Smith, Resonance and propulsion performance of a heaving flexible wing, *Phys. Fluids* **21**, 071902 (2009).
- [36] F. Paraz, L. Schouveiler, and C. Eloy, Thrust generation by a heaving flexible foil: Resonance, nonlinearities, and optimality, *Phys. Fluids* **28**, 011903 (2016).
- [37] A. P. Hoover, R. Cortez, E. D. Tytell, and L. J. Fauci, Swimming performance, resonance and shape evolution in heaving flexible panels, *J. Fluid Mech.* **847**, 386 (2018).
- [38] L. M. Sonneborn and F. S. Van Vleck, The bang-bang principle for linear control systems, *J. Soc. Ind. Appl. Math. A: Control* **2**, 151 (1964).
- [39] L. S. Pontryagin, *Mathematical Theory of Optimal Processes* (CRC Press, 1987).
- [40] D. E. Kirk, *Optimal Control Theory: An Introduction* (Courier Corporation, 2004).
- [41] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevFluids.10.043101> for details related to RL, and some asymptotic results.
- [42] I. Borazjani and F. Sotiropoulos, On the role of form and kinematics on the hydrodynamics of self-propelled body/caudal fin swimming, *J. Exp. Biol.* **213**, 89 (2010).
- [43] E. D. Tytell, C. Y. Hsu, T. L. Williams, A. H. Cohen, and L. J. Fauci, Interactions between internal forces, body stiffness, and fluid environment in a neuromechanical model of lamprey swimming, *Proc. Natl. Acad. Sci. USA* **107**, 19832 (2010).
- [44] R. E. Shadwick and S. Gemballa, *Fish Physiology*, Vol. 23 (Academic Press, 2006), pp. 241–280.
- [45] C. Eloy, On the best design for undulatory swimming, *J. Fluid Mech.* **717**, 48 (2013).
- [46] G. V. Lauder and E. D. Tytell, Hydrodynamics of undulatory propulsion, *Fish Physiol.* **23**, 425 (2005).
- [47] P. Ramdya and A. J. Ijspeert, The neuromechanics of animal locomotion: From biology to robotics and back, *Sci. Robot.* **8**, eadg0279 (2023).
- [48] L. Meteyard, S. R. Cuadrado, B. Bahrami, and G. Vigliocco, Coming of age: A review of embodiment and the neuroscience of semantics, *Cortex* **48**, 788 (2012).
- [49] R. Bale, M. Hao, A. P. S. Bhalla, and N. A. Patankar, Energy efficiency and allometry of movement of swimming and flying animals, *Proc. Natl. Acad. Sci. USA* **111**, 7517 (2014).
- [50] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, Proximal policy optimization algorithms, [arXiv:1707.06347](https://arxiv.org/abs/1707.06347).
- [51] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, Playing Atari with deep reinforcement learning, [arXiv:1312.5602](https://arxiv.org/abs/1312.5602).
- [52] Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, and L. Jackel, Handwritten digit recognition with a back-propagation network, in *Advances in Neural Information Processing Systems* Vol. 2 (Morgan-Kaufmann, 1989).
- [53] Y. LeCun, Y. Bengio, and G. Hinton, Deep learning, *Nature (London)* **521**, 436 (2015).
- [54] M. Curatolo and L. Teresi, Modeling and simulation of fish swimming with active muscles, *J. Theor. Biol.* **409**, 18 (2016).
- [55] L. D. Landau, E. M. Lifšic, E. M. Lifshitz, A. M. Kosevich, and L. P. Pitaevskii, *Theory of Elasticity: Volume 7* (Elsevier, 1986).