

# Neuroevolution of Decentralized Decision-Making in N-Bead Swimmers Leads to Scalable and Robust Collective Locomotion

Benedikt Hartl,<sup>1,2</sup> Michael Levin,<sup>2,3</sup> and Andreas Zöttl<sup>4</sup>

<sup>1)</sup>*Institute for Theoretical Physics, TU Wien, Austria*

<sup>2)</sup>*Allen Discovery Center at Tufts University, Medford, MA, 02155, USA*

<sup>3)</sup>*Wyss Institute for Biologically Inspired Engineering at Harvard University, Boston, MA, 02115, USA*

<sup>4)</sup>*Faculty of Physics, University of Vienna, Austria*

(\*Electronic mail: hartl.bene.research@gmail.com)

(Dated: 15 July 2024)

Many microorganisms are capable of swimming through viscous fluids such as water in order to search for nutrients, swim towards oxygen or light, or to escape from predators. To navigate their environment they often perform large nonreciprocal periodic deformations of their shape, by waving appendages such as cilia or flagella, or by deforming their entire body. Even unicellular organisms are fundamentally made of parts, which need to be cooperatively utilized to allow these creatures to navigate their environment, without using a centralized control mechanism. Here, we investigate the physical implications of decentralized decision-making of the actuators of a generalized N-bead Najafi Golestanian microswimmer, self-propelling via coordinated non-reciprocal swimming strokes. We treat each bead as an artificial neural network-based agent that perceives information about its neighbors and whose actions induce strokes of its adjacent arms. With neuroevolution techniques, we evolve optimal policies for the single-bead decision centers such that the N-bead collective efficiently self-propels as an individual, allowing us to investigate optimal locomotion policies for increasingly large microswimmer bodies. We demonstrate that such decentralized policies are robust and tolerant concerning morphological changes or defects and facilitate cargo transport or drug delivery applications “out of the box“, without further optimization. Our approach allows us to train large swimmers ( $N = 100$  and more), and we show that long-wavelength solutions lead to surprisingly efficient swimming gaits. Our work is of relevance to understand robust locomotion of biological microswimmers, to develop robust artificial microswimmer navigation strategies, and, in a broader conceptual context, for *Artificial Life* and in general emergent levels of individuality.

## I. INTRODUCTION

Many microorganisms live in viscous environments and their motion is governed by the physics of low Reynolds numbers where viscous forces dominate over inertia<sup>1-4</sup>. As a consequence, a common strategy is to periodically deform their body shape in a non-reciprocal fashion to swim. To thrive in their environment, they have developed different tailored strategies to exploit their swimming capabilities, such as actively navigating toward a nutrient-rich source, hunting down prey, escaping from a predator, or reproducing<sup>5,6</sup>. Besides of direct biological relevance, understanding the corresponding navigation strategies of microorganisms bears potential for biomedical or technical applications, potentially utilized by synthetic microswimmers used as targeted drug delivery systems<sup>7</sup>.

Nature evolved a large variety of strategies and control mechanisms to deform their shapes in order to swim fast, efficient, and adaptive to environmental conditions. In particular many eukaryotic cells are capable of complex and large shape deformations consisting of associated deformation amplitudes and wave lengths on the order of the entire organism. For example swimming algae cells or sperm cells move with the help of waving cilia or flagella, respectively, or amoebae such as *Dictyostelium*<sup>8</sup> and unicellular protists such as *Euglenia*<sup>9</sup> by deforming their entire cell body. These collective large deformations are typically achieved with the help of orchestrated and cooperative action of molecular motors and many other involved proteins, for example the local deformation of the cilia-forming axoneme<sup>5,10</sup>. Yet, such organisms - without an apparent centralized controller - cooperatively utilize their body components to navigate their environments in order to swim.

Moreover, such decentralized navigation policies tend to be robust and failure tolerant, e.g., if parts of the locomotive components are disabled or missing, thus showing strong signs of generalizability, as known for example for swarms, slime molds<sup>11</sup>, robots<sup>12</sup>. Such plastic and functional robustness and the innate drive for adaptability seem key features of biological organisms and evolution in general and, as recently suggested<sup>13</sup>, the underlying collective decision-making can be seen as a unifying concept for integrating biology across scales and substrates.

Here we combine low Reynolds number physics and ML assisted computation to better understand functional behavior of microswimmer locomotion. We thus investigate decentralized yet collective decision-making strategies of the locomotive parts of a model microswimmer, which represents e.g. an unicellular organism, or a controllable

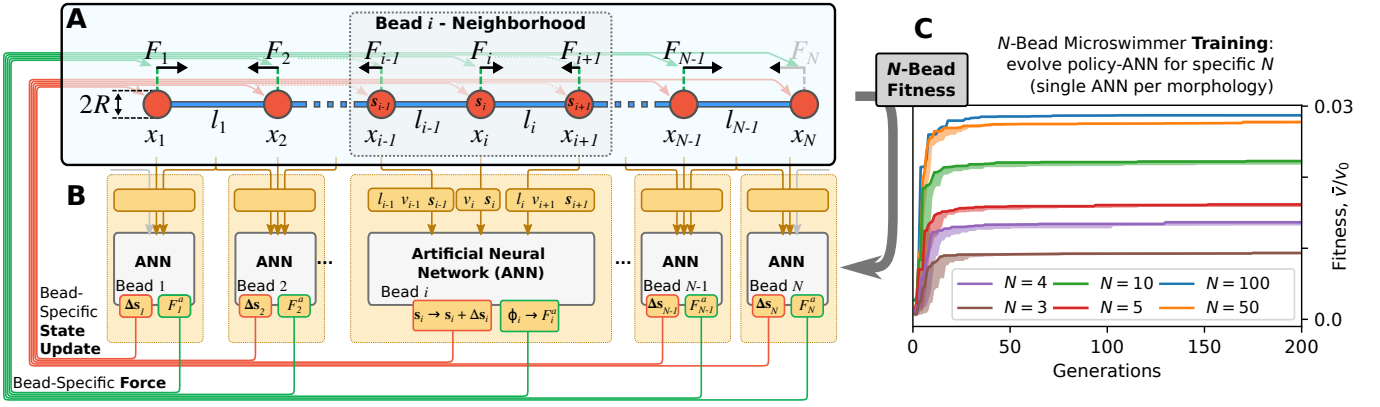


FIG. 1. (A) Schematics of an  $N$ -bead microswimmer environment, with (B) parametrically identical yet operationally independent Artificial Neural Networks (ANNs) acting as decentralized decision-making centers (or controllers) to update the respective internal states of the beads,  $\mathbf{s}_i \rightarrow \mathbf{s}_i + \Delta \mathbf{s}_i$  (red arrows), and to apply bead-specific forces,  $F_i \in [-2F_0, 2F_0]$  (green arrows); ensuring  $\sum_i F_i = 0$ , such that the entire microswimmer self-propels purely based on local perception-action cycles of the constituting bead controllers. (C) The training progress of optimizing various  $N$ -bead microswimmer navigation policies of type A (see text and fig. 2), respectively identifying for predefined values of ( $N = 3 - 100$ ) the parameters of the morphology-specific ANN controllers via evolutionary algorithms (EAs). The fitness-score for different  $N$ , quantifying a specific  $N$ -bead center of mass velocity  $\bar{v}$  (see sections II and V), is presented over 200 subsequent generations. Opaque-colored areas below the fitness trajectories indicate the corresponding STD of 10 independent EA searches per morphology and serve as a measure for convergence for the optimization process.

swimming microrobot. In our simplified approach the microswimmer can only deform in one dimension by longitudinal extractions and contractions of its body parts, modeled by  $N$  aligned beads linked by muscles. For  $N = 3$  our swimmer reduces to the famous Najafi Golestanian (NG) model microswimmer. We use Reinforcement Learning (RL) to optimize policies for finding efficient shape-deformation based locomotion strategies. In recent years RL has been employed to microswimmer models and various simple active agents.

The NG swimmer has been used recently as a standard model to employ *centralized* decisions on shape deformations based on sensory input of the swimmer's internal state and its environment, to perform tasks such as learning to swim or to perform chemotaxis. Since conventional centralized decision-making relies on the (sensory) input and states of all individual entities (*i.e.*,  $N$  beads) to control their actions globally, the dimension of the input vector to the decision-making machinery, such as artificial neural networks (ANNs), scale with  $N$ , which complicates the estimation of optimum policies by the controlling unit. Thus, while generalized NG swimmers with  $N > 3$  had been employed to optimize their swimming strokes<sup>14–17</sup>, they are limited to  $N \leq 10$ . In particular in recent ML studies, increasing the number of beads  $N$  requires exponentially more complex controllers which quickly becomes difficult, if not infeasible to train with current ML methods.

In our approach we are able to overcome these limitations by employing biologically motivated decentralized decision-making that allows us to extend our swimmers to much larger  $N$  than previously feasible, allowing us to identify locomotion strategies in the limit  $N \rightarrow \infty$ . To this end, we interpret each bead of this *collective swimmer* as an agent that can only perceive information about its adjacent beads and whose actions induce contractions or extensions of its adjacent muscles. We substitute the internal decision-making machinery of such single-bead agents by ANNs, and employ genetic algorithms and neuroevolution to machine learn optimal policies for such single-bead decision-making centers such that the entire  $N$ -bead swimmer can efficiently self-propel collectively, *i.e.*, in a decentralized way.

We show that such policies are robust and failure-tolerant concerning morphological changes of the collective microswimmer and that such decentralized policies - trained for microswimmers with a specific number of beads - generalize well to (vastly) different morphologies.

## II. SYSTEM

### A. The $N$ -Bead Swimmer Model

Here, we investigate swimming strategies optimized by RL and the corresponding physical implications of  $N$ -bead generalized NG<sup>18</sup> swimmer models moving in a fluid of viscosity  $\mu$  with local and bounded energy input. The swimmers consist of  $N$  co-centrally aligned spheres of radius  $R$  located at positions  $x_i(t_k)$ ,  $i = 1 \dots, N$ , at time  $t_k = k\Delta t$ ,  $k =$

$1, \dots, N_i$ , that are connected pairwise by massless arms of length  $l_i(t_k) = |x_{i+1}(t_k) - x_i(t_k)|$ , as illustrated in fig. 1 (A). The swimmer deforms and moves by applying time-dependent forces  $F_i(t_k) = (F_i^a(t_k) + F_i^r(t_k)) \in [-2F_0, 2F_0]$  on the beads, where  $F_0$  defines the force scale in our system. The active forces  $F_i^a(t_k)$  are proposed by RL agents (see below), and passive restoring forces<sup>16</sup>  $F_i^r(t_k)$  are applied when arm lengths  $l_i(t_k)$  becomes smaller than  $0.7L_0$  or larger than  $1.3L_0$ , where we choose  $L_0 = 10R$  as the reference arm length. The swimmer is force-free,  $\sum_i F_i(t_k) = 0$ , ensured by our procedure (see below), and the bead velocities  $v_i(t_k)$  are obtained in the Oseen approximation<sup>19</sup>,  $v_i = \dot{x}_i = F_i/(6\pi\mu R) + \sum_{j \neq i} F_j/(4\pi\mu|x_i - x_j|)$  (for details see Sec. section V A).

## B. Modeling system-level decision-making with decentralized controllers

To identify the active forces  $F_i^a(t_k)$  on the beads, we assign an ensemble of independent yet identical controllers to every bead which respectively can only perceive local information about adjacent beads (such as distances and velocities of their neighbors) and propose actions to update their respective states (such as proposing bead-specific forces to update their own positions), but with the shared objective of collectively self-propelling the entire  $N$ -bead swimmer's body. More specifically - as illustrated in fig. 1 and detailed in section V B - for each time  $t_k$  the controller associated with the bead  $i$  perceives its left- and right-neighbor distances,  $\mathcal{L}_i(t_k) = \{l_i(t_k), l_{i+1}(t_k)\}$ , and its own- and the neighboring beads' velocities,  $\mathcal{V}_i(t_k) = \{v_{i-1}(t_k), v_i(t_k), v_{i+1}(t_k)\}$ . Moreover, each bead maintains an internal vector-valued state,  $\mathbf{s}_i(t_k)$ . This state can be utilized by every controller to store, update, and actively share recurrent information with other beads that is not necessarily bound to the physical state of the swimmer but an emergent property of the collective RL system: Every controller thus perceives its neighboring states,  $\mathcal{S}_i(t_k) = \{\mathbf{s}_{i-1}(t_k), \mathbf{s}_i(t_k), \mathbf{s}_{i+1}(t_k)\}$ , additionally guiding the former's decision-making. In total, the perception of a single bead agent is given by  $\mathbf{p}_i(t_k) = \{\mathcal{L}_i(t_k), \mathcal{V}_i(t_k), \mathcal{S}_i(t_k)\}$ .

After integrating information about its local environment  $\mathbf{p}_i(t_k)$ , the controller of each bead  $i$  computes, and then outputs an action,  $\mathbf{a}_i(t_k) = \{\phi_i(t_k), \Delta\mathbf{s}_i(t_k)\}$ , comprising a *proposed* active force,  $\phi_i(t_k)$ , and an internal state update,  $\mathbf{s}_i(t_{k+1}) = \mathbf{s}_i(t_k) + \Delta\mathbf{s}_i(t_k)$  (see Fig. fig. 1). Notable, the proposed forces are limited to  $\phi_i(t_k) \in [-F_0, F_0]$  and hence set the force scale and maximum power consumption of the swimmer. To model a force-free swimmer, we propose two different methods how the mapping between the *proposed* forces  $\phi_i(t_k)$ , and the *actual* active forces  $F_i^a(t_k)$ , is achieved:

First, we interpret the proposed forces as pairwise arm forces  $\phi_i(t_k)$  and  $-\phi_i(t_k)$  applied between two consecutive beads  $i$  and  $i + 1$ , respectively (see fig. 2). This leads to the actual active forces  $F_i^a(t_k) = (\phi_i(t_k) - \phi_{i-1}(t_k)) \in [-2F_0, 2F_0]$  for beads  $i = 2, \dots, (N - 1)$ , and  $F_1^a(t_k) = \phi_1(t_k) \in [-F_0, F_0]$  and  $F_N^a(t_k) = -\phi_{N-1}(t_k) \in [-2F_0, 2F_0]$ . This automatically ensures  $\sum_{i=1}^N F_i^a(t_k) = 0$ . In this sense, the proposed actions can be understood as local decisions to expand/contract muscles between the beads.

Second, we assume that the proposed force  $\phi_i(t_k)$  of every controller directly targets the actual force applied to its associated bead, but, in order to fulfill the force-free condition, we subtract the mean  $\bar{\phi}(t_k) = \frac{1}{N} \sum_{j=1}^N \phi_j(t_k)$  from every proposed force and arrive at  $F_i^a(t_k) = (\phi_i(t_k) - \bar{\phi}(t_k))$ , again with  $F_i^a(t_k) \in [-2F_0, 2F_0]$ .

Hence the first approach ensures the global force-free condition via a series of locally annihilating pair-forces motivated by biological force dipole generation at small scales that cause the arms between the corresponding beads  $i+1$  and  $i$  to contract or extend. In contrast, the second approach using the regularization by collective feedback (via  $\bar{\phi}(t_k)$ ) can be interpreted as a mean-field approach which may be used by external controllers for artificial microswimmers<sup>20</sup>. Henceforth, we refer to the first scenario as *type A*, and to the second scenario as *type B* microswimmers, and alike for the corresponding self-navigation strategies or policies. In fig. 2, we schematically visualize how type A and B microswimmers execute their respective force-free conditions.

Following RL terminology, we refer to the mapping between perceptions and actions of an agent (or here synonymously, a controller) as the latter's *policy*,  $\pi_i : \mathbf{p}_i(t_k) \rightarrow \mathbf{a}_i(t_k)$ . In general, such a policy is a complicated and complex function of the input, and *Artificial Neural Networks* (ANNs) as universal function approximators<sup>21</sup> are well-suited tools to parameterize these objects for arbitrary agents and environments (see section V B). Thus, we approximate the RL agent's policy  $\pi_i$  by an ANN, formally expressed as a function  $f_\theta(\cdot)$  with parameters  $\theta$ , such that  $\mathbf{a}_i(t_k) = f_\theta(\mathbf{p}_i(t_k))$ . More specifically, we treat a single  $N$ -bead swimmer as a multi-agent system, each bead being equipped with an independent but identical ANN-based controller,  $f_\theta(\cdot)$ , reminiscent of a *Neural Cellular Automaton*<sup>22,23</sup> (NCA), where the union of the decentralized actions of all individual controllers give rise to a collective policy  $\Pi = \{\pi_1, \dots, \pi_N\} \approx \{f_\theta(\mathbf{p}_1(t_k)), \dots, f_\theta(\mathbf{p}_N(t_k))\}$ , of the entire virtual organism (see also<sup>24</sup>). Notably, only a single set of parameters  $\theta$  is used for all  $N$  bead-specific agents, *i.e.*, the same ANN controller is deployed to every bead (the states of the latter only differ in their initial conditions and dynamical input-output-relations). For our purposes, this renders the optimization problem much more tractable compared to situations with a single centralized controller,  $\bar{\Pi} \approx \bar{f}_\theta(\mathbf{p}_1(t_k), \dots, \mathbf{p}_N(t_k))$ , especially for large swimmer morphologies.

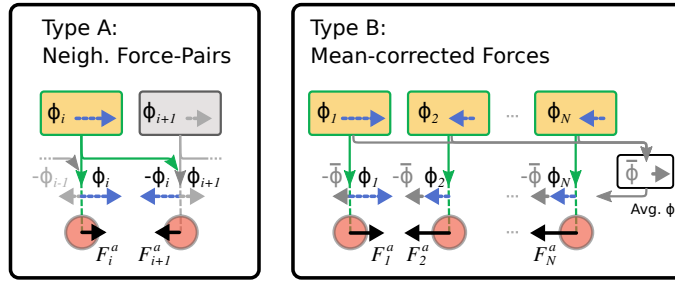


FIG. 2. Schematics of mapping the bead-specific proposed actions  $\phi_i(t_k)$  to the proposed active forces  $F_i^a(t_k)$  to ensure the global force-free condition  $\sum_{i=1}^N F_i^a(t_k) = 0$  (see text) by either interpreting the actions as force-pairs  $F_i^a(t_k) = (\phi_i(t_k) - \phi_{i-1}(t_k))$  between neighboring beads (left, type A microswimmers), or subtracting the global average  $\bar{\phi}(t_k) = \frac{1}{N} \sum_{i=1}^N \phi_i(t_k)$  from every proposed, bead-specific action  $F_i^a(t_k) = (\phi_i(t_k) - \bar{\phi}(t_k))$  (right, type B microswimmers).

Here, we aim at identifying optimal and robust swimming gates for arbitrarily large  $N$ -bead swimmers, which translates to finding suitable ANN parameters,  $\theta^{(\text{opt})}$ , such that the bead-specific perception-action cycles,  $\mathbf{a}_i(t_k) = f_{\theta^{(\text{opt})}}(\mathbf{p}_i(t_k))$ , collectively self-propel the multi-agent system efficiently in a certain direction. Thus, we utilize evolutionary algorithms (EAs)<sup>25–28</sup>, specifically a simple genetic algorithm<sup>29</sup> discussed in section V C, to adapt the ANN parameters (but not the ANN topology<sup>16</sup>) such that the entire  $N$ -bead swimmer's mean center of mass (COM) velocity,  $\bar{v} = \frac{1}{NT} \left| \sum_{i=1}^N (x_i(T) - x_i(0)) \right|$ , is maximized for a predefined swimming duration  $T = N_T \Delta t$ , chosen sufficiently large to provide the respective  $N$ -bead swimmer enough time to approach a steady swimming state and to execute several swimming strokes, starting from randomized initial positions in  $N_e = 10$  independent swimming episodes, see section V C for details. Thus, we define the objective, or fitness score as the episode-averaged mean COM velocity  $r = \langle \bar{v} \rangle_{N_e}$ , and search for  $\theta^{(\text{opt})} = \max_{\delta\theta}(r)$  through variation  $\delta\theta$  of the parameters  $\theta$  via EAs.

### III. RESULTS

#### A. Individual, bead-specific decisions facilitate collective swimming of an $N$ -bead swimmer

We utilize EAs to optimize the ( $\approx 50$ ) parameters of the ANN-based controllers (which are deployed to every bead in a specific morphology) for different realizations of the above-defined  $N$ -bead (multi-agent) microswimmer model (c.f., fig. 1 (A,B)). More specifically, we deploy morphologies ranging from  $N = 3$  to  $N = 100$  beads of both type A and B microswimmers and train every swimmer length  $N$  for both types independently via EAs, such that the corresponding  $N$ -bead collectives self-propel at a respective maximal mean COM velocity  $\bar{v}$ , maximizing the fitness score  $r$ ; for details on the utilized ANNs and the applied EA we refer to sections II and V.

The training progress for type A microswimmers with different length  $N$  is presented in fig. 1 (C). It demonstrates that the proposed decentralized decision-making strategy allows for finding fast swimming gates for all the considered swimmer lengths up to  $N = 100$ . Thus, our method is able to remove the bottleneck for machine-learning navigation policies of large-scale microswimmers by employing computationally manageable local ANN-based perception-action loops of their bead-specific agents.

To the best of our knowledge, this is the first time successful training of microswimmers with such a large number of degrees of freedom has been achieved.

#### B. Different strategies of autonomy: Large-scale coordination enables fast swimming

As depicted in fig. 3 (A,B), the COM velocities  $\bar{v}$  of both type A and B microswimmers monotonously increase with the body size, i.e. number of beads  $N$ . We normalize all velocities here with  $v_0 = 2F_0/(6\pi\mu R)$ , i.e. the velocity a bead would have if it dragged by an external force of strength  $2F_0$ . Interestingly, type B swimmers are significantly faster compared to type A swimmers by almost one order of magnitude, especially for large  $N$ . As illustrated in fig. 3 (A), for type A microswimmers with locally ensured force-free conditions, the mean COM velocity  $\bar{v}$  saturates with increasing  $N$  at  $\bar{v}_{\text{max}}/v_0 \approx 0.03$  for  $N = 100$ . In contrast (c.f., fig. 3 (B)), the fastest type B microswimmer (again at  $N = 100$ ) achieves a maximum COM velocity of  $\bar{v}_{\text{max}}/v_0 \approx 0.15$ .

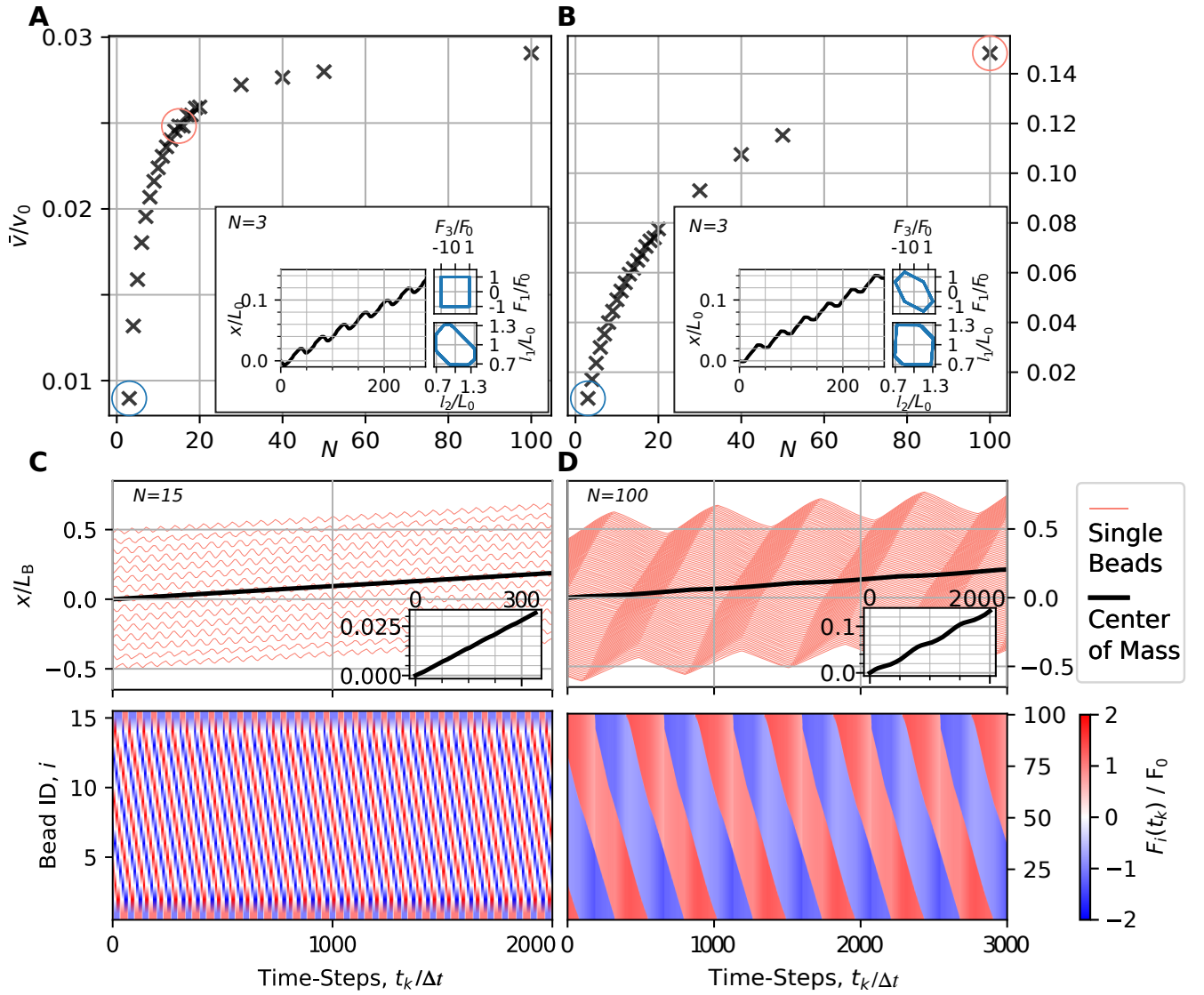


FIG. 3. (A) and (B): COM velocity,  $\bar{v}$ , of different type A and B microswimmers, respectively, corresponding to their final fitness score when optimized independently with EAs for  $N = 3$  to  $N = 100$  beads (see also fig. 1 (C)). Insets show COM trajectories, and  $(F_1^a, F_3^a)$ - and  $(l_1, l_2)$ -phase-space plots for  $N = 3$  (see also blue circles). (C) and (D): Typical bead-specific coordinate- (top panels) and force-trajectories (bottom panels) of an ( $N = 15$ )-bead type A and an ( $N = 100$ )-bead type B microswimmer, respectively (the examples, see red circles in (A, B), are chosen for illustrative purposes and are representative for all investigated  $N$  for both type A and B policies). Insets detail the corresponding COM trajectories. For type A microswimmers (A, C), periodic localized waves of arm strokes travel through the body. In contrast (B, D), large-scale collective body contractions allow large type B microswimmers to propagate much faster. Coordinate trajectories are normalized by the resting arm length  $L_0$  in the insets of (A,B), and by the body-length  $L_B = L_0 \times (N - 1)$  in (C,D).

These significant differences in the swimming velocities for type A and B swimmers are due to the different constraints on how the forces can be applied to the beads. The insets in panels (A, B) show results for the well-studied three-bead swimmer ( $N = 3$ ). First, they show the well-known periodic *1-step-back-2-step-forward* motion of the COM trajectory. Second, the corresponding steady state phase space dynamics of the active forces on beads 1 and 3,  $(F_1^a, F_3^a)$ , and of the arm lengths  $(l_1, l_2)$ , reiterating the periodic motion. Note that the force on bead 2 simply follows from  $F_2^a = -F_1^a, F_3^a$ .

In panels (C) and (D) we present trajectories of both the COM and bead-specific coordinates (top panels; respective insets emphasize the COM dynamics), and of the bead-specific proposed forces (bottom panels) for an ( $N = 15$ )-bead type A- and ( $N = 100$ )-bead type B microswimmer, respectively. These selected trajectories demonstrate the genuinely different swimming strategies for type A and B microswimmers (which have notably been optimized with

the same EA settings).

In type A microswimmers (fig. 3 (C)), the pairwise arm forces induce periodic waves of arm contractions of relatively high frequency but small wavelength, which travel through- and push forward the body of the  $N$ -bead microswimmer. For sufficiently large beads  $N$  this leads to a relatively smooth and linear COM motion (see inset in fig. 3 (C)).

In stark contrast, the fastest swimming strategies for type B microswimmers (fig. 3 (D)) assumes coordinated arm strokes across large fractions of their bodies, essentially contracting and extending almost the entire body simultaneously, which is reflected in the oscillatory COM motion even for very large  $N$  (see inset in fig. 3 (D)). This large-scale coordination exceeds the capabilities of the locally interlocked policies of type A microswimmers and strikes us as an emergent phenomenon<sup>30</sup> which - still based on purely local decision-making - is facilitated by the mean-field motivated feedback of the mean proposed force of all the agents in the system: type B microswimmers seemingly act as a single entity<sup>31</sup>, despite the fact that the precise number of constituents,  $N$ , is not important and can vary.

As shown in fig. 3 (C) and (D), typical policies of the respective type B swimmers are reminiscent of the large amplitude contraction-wave based locomotion of crawling animals such as caterpillars. Similarly, the crawling locomotion of crawling fly larvae had been optimized using RL recently<sup>32</sup>.

### C. Transferable evolved policies: decentralized decision-making generalizes to arbitrary morphologies

We recently showed<sup>33</sup> that biologically inspired NCA-based multi-agent policies - especially when evolved via evolutionary processes - can display behavior that is highly robust against structural and functional perturbations and thus exhibit increased generalizability, adaptability, and transferability. We thus investigate here, whether our decentralized locomotion policies - genuinely evolved for microswimmer bodies with exactly  $N_T$  beads - generalize to morphological changes. More specifically, we deploy - without any retraining or further adaptation - ANNs that have been optimized for a particular number of  $N_T$  beads into microswimmer bodies with a different number of  $N$  beads instead and evaluate the corresponding swimming velocities for such *cross-policy* environments, now for  $N = 3$  to 300.

Remarkably, as illustrated in fig. 4(A) and (B), we find that the vast majority of all policies that are most optimal for a particular value of  $N_T$  are also highly effective in self-propelling ( $N \neq N_T$ )-bead microswimmers for both type A and B, respectively. This even holds for situations where  $N_T \ll N$ , such as  $N_T = 3$  and  $N = 300$ . Notably, we did not explicitly optimize for this quality at all, but it is an emergent phenomenon of the inherent decentralized decision-making of the system. Thus, the collective nature of the here proposed swimming policies renders the evolved locomotive strategies highly adaptive to morphological changes, virtually irrespective of the specific number of beads  $N$  used during deployment.

Moreover, the set of policies evolved- and deployed for different bead numbers of  $N_T$  and  $N$ , respectively, allows us to estimate a transient trend for optimal swimming gates for large  $N$ : if unperturbed and starting from proper initial conditions, the arm-lengths  $l_i(t_k)$  of an arbitrary trained  $N_T$ -bead swimmer describe limit cycle dynamics (c.f, fig. 3). Assuming that all arms oscillate at the same angular velocity  $\bar{\omega}$ , and exhibit the same cross-correlation times  $\bar{\tau}$  to their neighboring arms  $l_i(t_k)$  and  $l_{i+1}(t_k + \bar{\tau})$ , we can express each of the  $i = 1, \dots, N$  arm lengths as a  $2\pi$ -periodic function  $\hat{l}_i(t) = f_\ell((t - i\bar{\tau})\bar{\omega} + \phi)$ , with the phase shift  $\phi$  defining the initial conditions (which we ignore henceforth); for more details about  $\bar{\omega}$  and  $\bar{\tau}$  we refer to section VD. Notably, we can also write  $\hat{l}_i(t)$  in the form of a wave-equation as  $\hat{l}_i(t) = f_\ell(t\bar{\omega} + 2\pi i/\bar{\lambda} + \phi)$ , where the bead index  $i$  controls the ‘‘spatial’’ oscillations at a (dimensionless) wavelength  $\bar{\lambda} = \frac{2\pi}{\bar{\omega}\bar{\tau}}$  irrespective of the corresponding physical bead positions  $x_i(t_k)$ .

In fig. 4 (C, E, G), we respectively present  $\bar{\omega}$ ,  $\bar{\tau}$ , and  $\bar{\lambda}$  for the fastest type A microswimmers as a function of  $N$  (blue circles), additionally to the training conditions where  $N = N_T$  (magenta ‘‘x’’ symbol); panels (D, F, H) show the analogous analysis for type B swimmers. We see that all three quantities behave very differently for type A and B: for type A (see fig. 4 (C, E, G)), a careful analysis reveals a minute transient logarithmic behavior of  $\bar{\omega} \approx a \ln N + b$  with  $a = -5.26 \times 10^{-3}$  rad/ $\Delta t$  and  $b = 1.13 \times 10^{-1}$  rad/ $\Delta t$ , an almost constant value of  $\bar{\tau} \approx 12.2 \Delta t$ , and a transient logarithmic behavior of  $\bar{\lambda} \approx a \ln N + b$  with  $a = 2.37 \times 10^{-1}$  and  $b = 4.64$ . In contrast, for type B (see fig. 4 (D, F, H)), the angular velocity follows a power law behavior  $\bar{\omega} \approx \alpha N^\beta$  with  $\alpha = 3.92 \times 10^{-1}$  rad/ $\Delta t$  and  $\beta = -8.19 \times 10^{-1}$ , and  $\bar{\lambda} \approx \alpha N^\beta$  follows again a power law behavior with  $\alpha = 1.33$  and  $\beta = 9.17 \times 10^{-1}$ . Notably, the evaluated  $\bar{\tau}$  values for type B microswimmers (fig. 4 (F)) exhibit a more complex pattern, but can be evaluated and thus functionally described by  $\bar{\omega}$  and  $\bar{\lambda}$ .

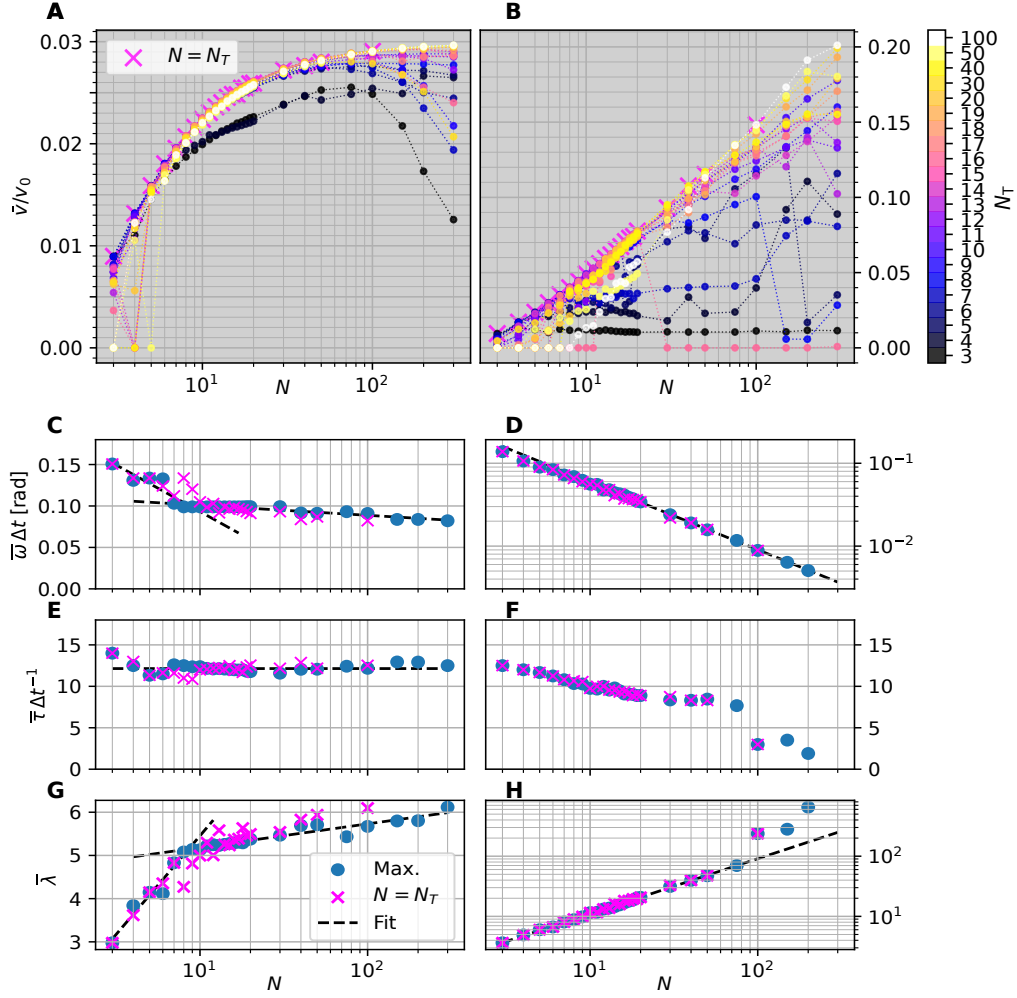


FIG. 4. (A) and (B): Cross-policy *transferability* evaluations, depicting the mean COM velocity of  $N$ -bead microswimmers deployed with policies optimized for  $N = N_T$  (color-coding) for type A and B microswimmers, respectively. The vast majority of the policies evolved for  $N_T$ -beads generalize well to vastly different  $N$ -bead morphologies *without further optimization*. (C) and (D): The mean angular velocity,  $\bar{\omega}$ , of the arm-length limit-cycle dynamics of the most optimal  $N$ -bead type A and B microswimmers, respectively (blue circles; fastest policies trained with  $N_T$ -beads and deployed to  $N$ -bead microswimmers); magenta  $\times$  symbols illustrate  $\bar{\omega}$  for the corresponding training conditions,  $N = N_T$ . (E,F) and (G,H): Similar to (C,D), but showing the mean cross-correlation time,  $\bar{\tau}$  ((E,F), see text), between neighboring arms,  $l_i(t)$  and  $l_{i+1}(t + \tau)$ , and the corresponding (dimensionless) wavelength,  $\bar{\lambda}$  ((G,H), see columns in lower panels of fig. 3 (C, D) at fixed time  $t_k$  and see text) of type A and B microswimmers, respectively, as a function of  $N$ . Dashed lines indicate functional fits (see text).

#### D. Large-scale coordination leads to efficient locomotion

In our approach we limit the maximum forces on each of the beads, and thus the maximum bead velocities. This procedure is somewhat similar as fixing the total power consumption of the swimmer. In previous optimization procedures on 3- and  $N$ -bead swimmers commonly the swimming efficiency is optimized<sup>14,34</sup>, where e.g. the power consumption of the entire swimmer is taken into account as a single quantity. In contrast, in our work we set local constraints on the swimmer by limiting the forces on every bead. Although we hence did not optimize in our RL procedure for the efficiency, we measure the efficiency  $\eta$  of our swimmers, determined by  $\eta = 6\pi\mu R_{\text{eff}}\bar{v}^2/P$  where  $\mathcal{P} = \frac{1}{T} \int \sum_i v_i(t)F_i(t)dt$  is the stroke-averaged power consumption. There is no unique way to define the effective radius of our swimmer (see also the discussion in Ref.<sup>34</sup>), and it is hence not straightforward to compare the effective radius of swimmers of different size  $N$ . we choose here  $R_{\text{eff}} = NR$  which reflects the combined drag on all of the spheres (neglecting hydrodynamic interactions). The power consumption is naturally limited to be  $\mathcal{P} < \mathcal{P}_{\text{max}}$  with  $\mathcal{P}_{\text{max}} = 2NF_0v_0 = 2NF_0^2/(6\pi\mu R)$ , for both cases type A and B. However type B swimmers can exploit their higher freedom to adjust the forces on the beads compared to the arm-force-limited type A swimmer to locomote at higher

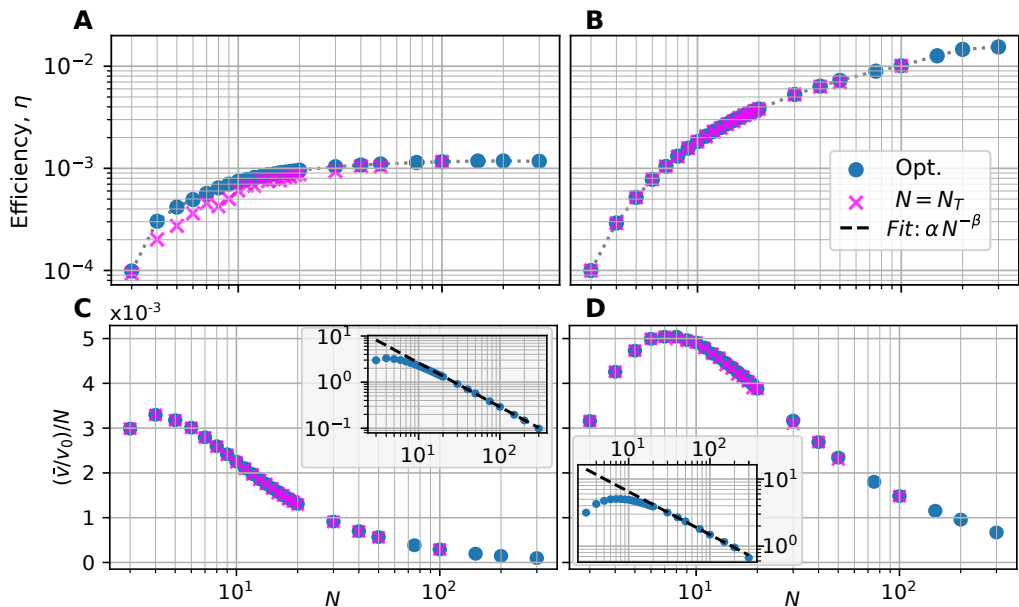


FIG. 5. (A) and (B): efficiency  $\eta$  of the most optimal  $N$ -bead type A and B microswimmers (blue circles), respectively; magenta  $\times$  symbol emphasize the efficiency of microswimmers trained with  $N = N_T$ . While for type A microswimmers, the efficiency plateaus at  $\eta_{\max}^{(a)} = 1.2 \times 10^{-3}$  for large  $N$ , especially the large type B microswimmers show biologically realistic efficiencies<sup>1</sup> of up to  $\eta_{\max}^{(b)} = 1.5 \times 10^{-2}$ . (C) and (D): mean COM velocity per bead  $\bar{v}/N$  for the respective type A and B microswimmers. The insets show that the transient decline of the COM velocity per bead follows a power law,  $(\bar{v}/v_0)/N = \alpha N^{-\beta}$ , with  $\alpha = 2.34 \times 10^{-2}$ ,  $\beta = 9.53 \times 10^{-1}$  (C) and  $\alpha = 2.83 \times 10^{-2}$ ,  $\beta = 6.41 \times 10^{-1}$  (D), c.f., black dashed fits.

speed, and hence at higher efficiency  $\eta$ . As seen in fig. 5 (A,B) for both type A and B swimmers the efficiency increases with swimmer length  $N$  and levels off at large  $N$ . The efficiency is quite low for all swimmer lengths  $N$  for type A swimmers, and levels off at  $\approx 0.1\%$ . In contrast, long type B swimmers can reach surprisingly high efficiencies of even  $> 1\%$ , which is quite efficient and comparable to the efficiency of real existing microswimmers<sup>1</sup>.

As we discussed, a larger microswimmer can swim faster due to the emergence of long wavelength longitudinal waves. Indeed it thus is not surprising that longer type B swimmers are faster than shorter ones. Since animals typically scale their speed with their size<sup>35</sup>, we determine the swimmer speed per size  $\bar{v}/N$  as shown in fig. 5 (C,D). Interestingly, we see that  $\bar{v}/N$  has a maximum for  $N = 4$  for type A and at  $N = 8$  for type B. Hence these swimmers need the smallest amount of time to swim their own body length  $NL_0$  in their respective swimmer category.

### E. Towards drug delivery: Robust- and failure tolerant locomotion allows cargo transport without re-training

Since our evolved microswimmer policies show strong resilience even against significant morphological perturbations (see section III C), we aim to investigate this operational plasticity even further: we systematically load the arms of the most optimal ( $N = 13$ )-bead type A and B microswimmers with extra passive ‘‘cargo’’ beads of variable radius  $R_c \in [0, 2R]$ , and evaluate the dynamics of the corresponding loaded ( $N = 13 + 1$ )-bead microswimmer - without retraining or further optimization; the choice of these particular swimmer instances is representative for other solutions and morphologies of this contribution.

These cargo beads can geometrically be located between two neighboring beads of the microswimmer, but remain functionally disjoint from the latter (cargo beads are not connected by arms to any body beads): when placed at an arm  $l_i$ , a cargo bead does not disrupt the exchange of information between the corresponding beads  $i$  and  $(i + 1)$  and thus does not affect the inputs of the respective bead-specific ANNs. Furthermore, cargo beads do not propose active forces independently,  $F_c^a(t_k) = 0$ , and are thus passive elements, pushed or dragged around by the hydrodynamic interaction with all other beads in the system and the restoring forces of nearby beads. Due to the one-dimensional geometry of our system and the harmonic restoring forces, once loaded between two beads of the microswimmer, the cargo-beads are topologically fixed in the microswimmer’s body.

Figure 6 demonstrates, that our approach of utilizing decentralized, bead-specific controllers for  $N$ -bead microswimmer locomotion not only gives rise to highly robust self-propulsion policies (see above) but can also be used - out



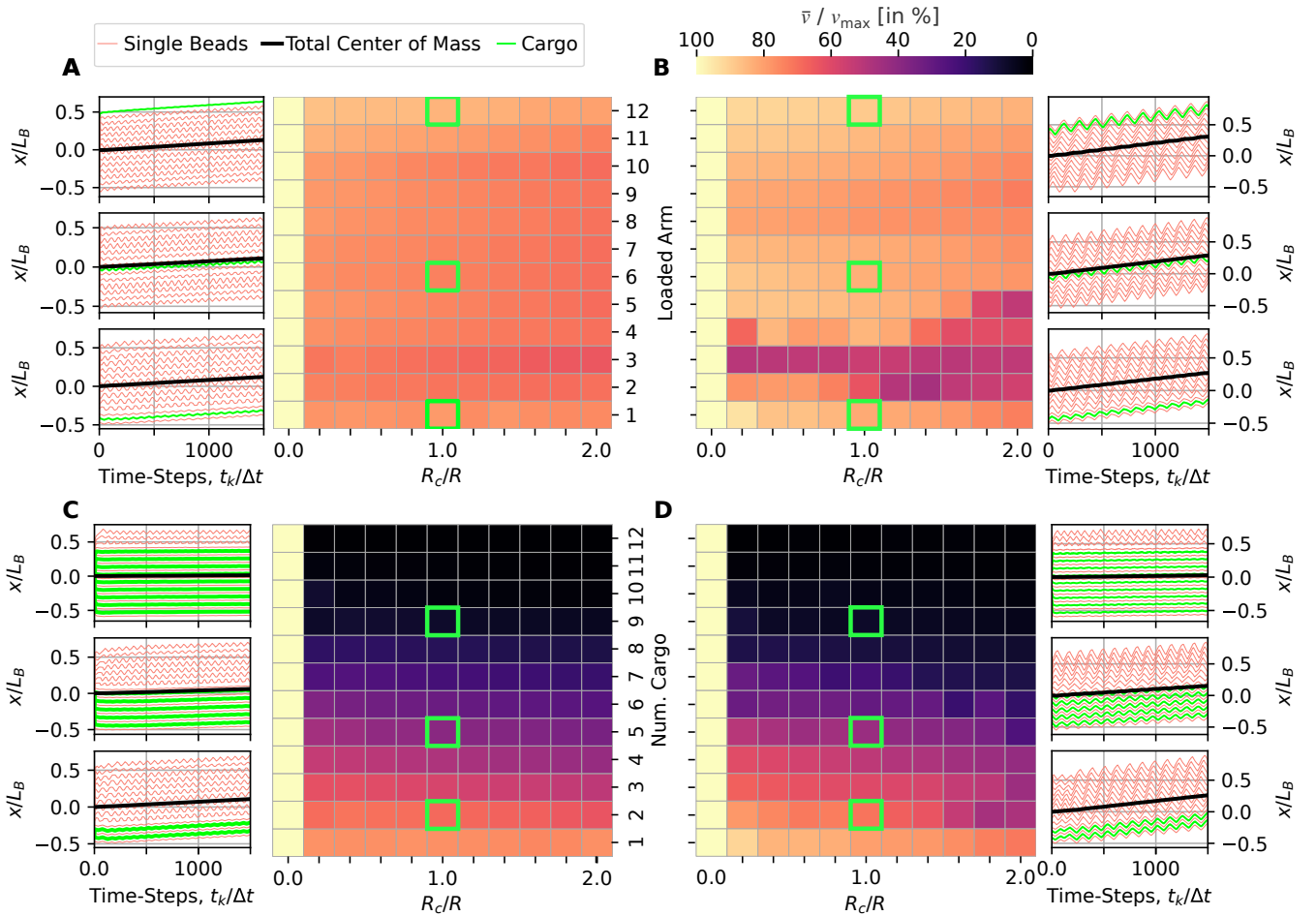


FIG. 6. (A) and (B): Mean COM velocity  $\bar{v}$  in % (color-coded) of the maximum velocity  $v_{\max}$  (c.f., respective left columns) of representatively chosen ( $N = 13$ )-bead type A and B microswimmer solutions, respectively, carrying a cargo bead of radius  $R_c \in [0, 2R]$  (horizontal axis) at different loading sights, *i.e.*, at the respective arms  $l_1$  through  $l_{N-1}$  (vertical axis). For each panel, three selected trajectory plots (indicated by green frames in respective heatmaps) highlight the corresponding cargo-swimmers' single-bead- (red), COM- (black), and cargo bead trajectories (green) scaled by the body-size  $L_B = (N-1)L_0$ . Both swimmers are capable - *without any further optimization* - of carrying different cargo loads efficiently. (C) and (D): Similar to (A) and (B) but loading the respective ( $N = 13$ )-bead microswimmers with a total number of  $N_c = 1, \dots, (N-1)$  cargo beads (vertical axis) of the same size,  $R_c$  (horizontal axis), where each cargo occupies a single arm  $l_i$  starting from  $i = 1, \dots, N_c$  (c.f., green cargo lines in designated example trajectories). Again, and depending on  $R_c$ , both swimmers are capable of carrying a large number of up to  $N_c = 7-8$  cargo beads (*i.e.*,  $\approx 50-60\%$  of their actual body-sizes with  $N = 13$ ), before the locomotion fails as the arms are (b)locked by the cargo (see top-most trajectories in (C) and (D)). For the reference case,  $R_c = 0$ , we do not include any cargo beads in the simulations, thus the corresponding velocity represents  $v_{\max}$  for type A and B, respectively.

of the box - for cargo transport applications<sup>36-38</sup>: Both type A and B microswimmers are capable of transporting large amounts of cargo beads efficiently. In general, the swimming speed of a loaded microswimmer decreases with increasing cargo size,  $R_c$ . But as illustrated in Figure 6 (A) and (B), although showing an effect, neither type A nor type B microswimmers are critically affected by the precise location of a single cargo bead (*i.e.*, at which arm the cargo bead is located), rendering the system highly robust against such geometrical defects.

Next, we successively fill all arms of a single microswimmer from the left,  $l_1$ , to the right,  $l_{N_c}$ , simultaneously with a total number of  $N_c = 1, \dots, (N-1)$  cargo beads of equal radius  $R_c$  (one cargo per arm) and measure the speed of the correspondingly loaded ( $N + N_c$ )-bead microswimmers as a function of the number of loaded arms and cargo size. Both type A and B microswimmers are capable of transporting multiple cargo loads at once efficiently: as illustrated in Figure 6 (C) and (D) both type A and B microswimmers can carry up to  $\approx 50-60\%$  of their comprising active beads  $N$  before the respective locomotion capabilities fail due to an increasing number of blocked arms.

Thus, our evolved navigation policies of the proposed microswimmer system not only show strong resilience against partly significant morphological changes of the swimmer's body (see also section III C), but even against functional

failures of single or multiple actuators (c.f., blocked arms in fig. 6). Notably, since we did not explicitly optimize for these qualities, the precise cargo transport behavior of different  $N$ -bead swimmer realizations may deviate. Anyhow, the inherent robustness and increased structural and functional transferability of such biologically-inspired decentralized decision-making systems<sup>13,33,39</sup> renders our approach as a promising candidate to further bio-medically relevant applications such as targeted drug-delivery systems<sup>40,41</sup> in complex biological environments<sup>42,43</sup>.

#### IV. CONCLUSION

Our study has shown that machine-learning of decentralized decision-making strategies of the actuators in an *in silico* microswimmers with neuroevolution techniques can lead to highly efficient system-level navigation policies of the entire (virtual) organism that are, moreover, highly robust with respect to morphological changes or defects.

By treating each of the  $N$  beads of a generalized Najafi Golestanian (NG) microswimmer model as an Artificial Neural Network (ANN)-based agent that perceives information about its adjacent beads and whose actions induce contractions or extensions of its adjacent muscles, we have been able to optimize policies for the single-bead decision-making centers which collectively facilitate highly efficient swimming gates on the system-level of the NG swimmer. This has allowed us to identify locomotion strategies for increasingly large microswimmer bodies (ranging from  $N = 3$  to  $N = 100$ ) with efficiencies of up to  $\eta \approx 1\%$ , close to that of “real” biological microswimmers; to the best of our knowledge, this is the first demonstration of successfully training an ( $N = 100$ )-bead microswimmer.

While having focused here on evolving the functional parameters of the decentralized controllers of the actuators with the common goal of collectively self-propelling their host organism, we report that the final navigation policies generalize well - *i.e.*, without any further optimization - towards morphological changes (training and deployment at vastly different numbers of beads  $N$ ). This renders our approach robust and even modular from both a physiological and information processing point of view<sup>24,33</sup>, going well beyond more traditional, in this sense much more brittle reinforcement learning (RL) applications, typically based on centralized (system-level-) controllers.

Intriguingly, and as we demonstrate, the inherent robustness and modularity of the decentralized decision-making in the evolved  $N$ -bead swimmer navigation policies makes our system directly suitable for cargo transport applications without further optimization or fine-tuning. Reminiscent to the remarkable structural and functional plasticity of “real” biological matter<sup>39,44</sup>, we emphasize the striking inherent ability of our microswimmer policies to adapt without any retraining to perturbations or morphological changes on the fly (in “zero-shots”), thus framing our work in the field of *Artificial Life*<sup>45</sup>. This resonates well with William James’ definition of intelligence of “achieving a fixed goal with variable means”<sup>13,46</sup>.

The limiting computational factor in our simulations is not the controller part, but the  $\mathcal{O}(N^2)$  complexity of the hydrodynamic model, which could be leveraged by further modelling, numerical optimization, or hardware accelerators. However, the scalability of our approach allows us to analytically generalize the optimized policies as a function of  $N$ , again overcoming limitations posed by traditional RL methods, and leveraging analytical investigations of the generalized NG model to the thermodynamic limit.

Thus, our approach - at the intersection of biology, biophysics, artificial intelligence, and robotics - constitutes a promising route toward developing highly robust and failure tolerant biologically inspired microswimmer policies (with respect to morphological defects) at manageable computational costs. Moreover, the here proposed ANN controllers and the learning paradigm are not limited to integrating information of three co-align spherical neighbors of an NG microswimmer, but can be extended to arbitrary swimmer geometries and models. Despite potential computational challenges, we thus aim to extend this approach to more realistic two- or three-dimensions microswimmer models in future work.

Due to the inherent structural and functional plasticity of the evolved collective policies, our approach represents a promising framework for developing biologically inspired autonomous cargo transport<sup>36,42,43</sup> or drug-delivery systems<sup>37,38,40,41</sup>, especially when combined with autonomous chemotactic capabilities (which can be utilized analogous to Ref. 16).

#### V. METHODS

##### A. Hydrodynamic interactions and numerical details for the $N$ -bead swimmer model

The microswimmer consists of  $N$  hydrodynamically interacting beads located at positions  $x_i(t_k)$ ,  $i = 1, \dots, N$ , at time  $t_k$ . The bead positions change over time by applying forces  $F_i(t_k)$ , consisting of active ( $F_i^a(t_k)$ ) and passive ( $F_i^r(t_k)$ ) contributions. The arm lengths are given by  $l_i(t_k) = x_{i+1}(t_k) - x_i(t_k)$ ,  $i = 1, \dots, N - 1$ . At time  $t_k$  the velocities of the beads  $v_i(t_k)$  depend linearly on the applied forces through the mobility tensor  $\mathcal{M}(t_k)$ :

$v_i(t_k) = \sum_j \mathcal{M}_{ij}(t_k) F_j(t_k)$ . Self-mobilities are given by Stokes formula  $\mathcal{M}_{ii} = 1/(6\pi\mu R)$ , while cross-mobilities describe hydrodynamic interactions, which we consider in the far-field limit in the Oseen approximation:  $\mathcal{M}_{ij}(t_k) = 1/(4\pi\mu|x_i(t_k) - x_j(t_k)|)$ . Active forces are applied as described in the main text. We apply passive spring forces as pairwise harmonic restoring forces with spring constant  $k = 10F_0/R$  to the beads if  $l_i < 0.7L_0$  or if  $l_i > 1.3L_0$  which limits the arm extensions, as can be seen for example in the inset of fig. 3 (A,B). The equations of motion of the microswimmer are then solved using a forth-order Runge Kutta scheme.

## B. Artificial Neural Network-based Decentralized Controllers

Mimicking the flexible operations of biological neural circuits, *Artificial Neural Networks* (ANNs) consisting of interconnected *Artificial Neurons* (ANs) have become invaluable numerical tools for statistical learning applications<sup>47–50</sup>. Each AN takes a set of inputs,  $\mathbf{x} \in \mathbb{R}^n$ , and maps them onto a single output value,  $y \in \mathbb{R}$ , through a weighted non-linear filter,  $y = \sigma(\mathbf{w} \cdot \mathbf{x} + b)$ , where the weights  $\mathbf{w} \in \mathbb{R}^n$  represent the strengths of every individual input connection, and  $b \in \mathbb{R}$  is the bias, representing the AN's firing threshold<sup>51</sup>. ANNs are commonly organized into layers of ANs. A *Feed Forward* (FF) ANN transforms an input,  $\mathbf{x}^{(1)} \in \mathbb{R}^{N_0}$ , through a series of hidden layers ( $i = 1, \dots, N_L$ ) to an output vector,  $\mathbf{y}^{(\text{out})} \in \mathbb{R}^{N_L}$ . Each layer's output is calculated as  $\mathbf{y}^{(i)} = \sigma(\mathcal{W}^{(i)} \cdot \mathbf{x}^{(i)} + \mathbf{b}^{(i)})$ , where  $\mathcal{W}^{(i)} = \{w_{jk}^{(i)}\} \in \mathbb{R}^{N_i \times N_{i-1}}$  is the weight matrix and  $\mathbf{b}^{(i)} \in \mathbb{R}^{N_i}$  is the bias vector. In a FF ANN, the output of layer  $i$  becomes the input to the next deeper layer ( $i + 1$ ) through successive dot-products, until an output is generated. Training an ANN thus involves optimizing a set of parameters,  $\theta = \{w_{jk}^{(i)}, b_k^{(i)}\}$ , *i.e.*, the entire network's weights and biases, such that the ANN's response to known inputs have minimal deviation to (typically predefined) desired outputs<sup>52–55</sup>.

We here utilize a single ANN to approximate a decision-making policy for microswimmer locomotion in contrast to our previous work<sup>16</sup>. The ANN is deployed to every bead of an  $N$ -bead microswimmer independently. Thus, each ANN-augmented bead represents an agent that is immersed into a chain of  $N$  single-bead agents comprising the body of an  $N$ -bead microswimmer (see fig. 1). As detailed in fig. 7, the bead specific agents of the microswimmer successively perceive the states of their respective neighboring beads and integrate this local information to initiate swimming strokes locally, following a decentralized policy that self-propels the entire microswimmer in the

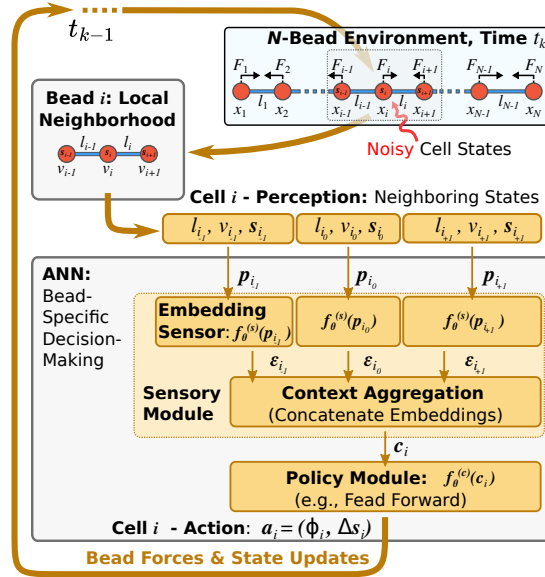


FIG. 7. Schematic information-flow chart and environmental updates (chronologically following thick brown arrows) of ANN-based bead-specific decentralized decision-making implementing a system level policy that controls the locomotion of an  $N$ -bead microswimmer. The detailing ANN architecture (inspired by<sup>33</sup>) emphasizes an ANN's perception,  $\mathbf{p}_{i,\nu}$ , of bead  $i$ 's local neighborhood,  $\nu = -1, 0, 1$  (see text), followed by an embedding,  $\mathbf{p}_{i,\nu} \rightarrow \epsilon_{i,\nu}$ , and concatenation layer in the sensory module that results in a bead-specific context matrix,  $C_i = (\epsilon_{i-1}, \epsilon_{i0}, \epsilon_{i+1})$ , based on which the policy module proposes an action,  $\mathbf{a}_i$ , comprising the proposed force,  $\phi_i$ , and the cell-state update,  $\Delta s_i$ . This step is performed by every bead independently at every successive time step,  $t_k$ , to induce an update of the microswimmer's state at times  $t_{k+1}$  by considering the regularized forces,  $\phi_i \rightarrow F_i$ , in the equations of motion of the  $N$ -bead hydrodynamic environment,  $x_i(t_k) \rightarrow x_i(t_{k+1})$ , and performing a noisy (c.f., red wiggly arrow) cell-state update.

hydrodynamic environment.

From the perspective of *Reinforcement Learning*<sup>55</sup>, our approach can thus be considered a trainable multi-agent system that needs to utilize local communication and decision-making to achieve a target system-level outcome<sup>33</sup>. The goal is to identify a set of ANN parameters  $\theta$  for the localized agents that facilitates such collective behavior, which we achieve here via evolutionary algorithms (EAs)<sup>33</sup>, see section V C.

Let us now specify the particular ANN architecture, and the perception (ANN input) and action (ANN output) conventions that we utilize in this contribution (as illustrated in fig. 7).

First, we define the neighborhood of a particular bead  $i = 1, \dots, N$ : In our example of a one-dimensional, linear  $N$ -bead swimmer, the direct neighbors of bead  $i$  are given by the beads  $i \pm 1$ . To address each bead in this “ $i$ -neighborhood”, we introduce the index notation  $i_\nu = i + \nu$  with  $\nu \in \{-1, 0, 1\}$ ;  $i_0$  thus addresses bead  $i$  itself.

Second, we define the perception, or ANN input of bead  $i$  as  $\mathcal{P}_i = \{\mathbf{p}_{i_{-1}}, \mathbf{p}_{i_0}, \mathbf{p}_{i_{+1}}\}$ , a composite matrix containing local, neighbor  $i_\nu$ -specific perceptions,  $\mathbf{p}_{i_\nu}$ , of bead  $i$ : We define the neighbor-specific perception as  $\mathbf{p}_{i_\nu} = (l_{i_\nu}, v_{i_\nu}, \mathbf{s}_i)$ , with bead  $i_\nu$ -specific arm length to the neighboring beads  $l_{i_\nu} = |x_i - x_{i_\nu}| \in \mathbb{R}$ , velocity  $v_{i_\nu} \in \mathbb{R}$ , and an internal, vector-valued state  $\mathbf{s}_i \in \mathbb{R}^{n_{ca}}$  at time  $t_k$  (see below); out-of-bound inputs for the head and tail beads are discarded by formally setting  $\mathbf{p}_0 = \mathbf{p}_{N+1} = \mathbf{0}$  as we count  $i = 1, \dots, N$ .

The internal state of a bead is inspired by the cell state of a *Cellular*<sup>56</sup>, or rather *Neural Cellular Automaton*<sup>57</sup> (NCA) that can be utilized by each bead to memorize or exchange information with neighbors. Analogous to previous work<sup>33</sup>, we define the update of a bead’s internal state between two successive time steps as  $\mathbf{s}_i(t_{k+1}) = (\mathbf{s}_i(t_k) + \Delta \mathbf{s}_i(t_k) + \xi)$ , where we introduced a zero-centered Gaussian noise term of STD  $\xi$  increasing the robustness of evolved solutions<sup>33</sup>. Additionally, we clamp the elements of  $\mathbf{s}_i(t_k)$  to the interval  $[-1, 1]$  after each update.

Third, we here utilize a fixed ANN architecture and deploy it to every single-bead agent, as illustrated in fig. 7 (see 24,33,58): we partition a bead’s ANN into a sensory module,  $f_\theta^{(s)}(\cdot)$ , and a policy module,  $f_\theta^{(c)}(\cdot)$ . The sensory module maps each neighbor-specific input separately into a respective sensor embedding,  $\varepsilon_{i_\nu}(t_k) = f_\theta^{(s)}(\mathbf{p}_{i_\nu}(t_k)) \in \mathbb{R}^{n_{embd}}$ , that are merged into a bead-specific context matrix  $\mathcal{C}_i(t_k) = (\varepsilon_{i_{-1}}(t_k), \varepsilon_{i_0}(t_k), \varepsilon_{i_{+1}}(t_k))$ . The subsequent policy module, or controller ANN eventually outputs the beads’s action,  $\mathbf{a}_i(t_k) = f_\theta^{(c)}(\mathcal{C}_i(t_k)) = (\phi_i, \Delta \mathbf{s}_i)$ , proposing a bead-specific force  $\phi_i \in [-F_0, F_0]$  (to-be regularized,  $\phi \rightarrow F_i$ , such that  $\sum_i F_i = 0$ , see section II) and an internal state update  $\Delta \mathbf{s}_i \in \mathbb{R}^{n_{ca}}$ .

Forth, we specifically utilize a single-layer FF sensory module,  $f_\theta^{(s)}(\cdot)$ , with  $(N_0^{(s)} = 2 + n_{ca})$  input and  $N_1^{(s)} = n_{embd}$  output neurons with a  $\tanh(\cdot)$  filter (the same network for all 3 neighbors). The  $(3 \times n_{embd})$  context matrix,  $\mathcal{C}_i$ , is then flattened into a  $(3 n_{embd})$ -dimensional vector, which is processed by the policy module,  $f_\theta^{(c)}(\cdot)$ : again, a single FF layer with  $N_0^{(c)} = 3 n_{embd}$  and  $(N_1^{(c)} = 1 + n_{ca})$ , followed by a clamping filter,  $\sigma^{(c)}(\cdot) = \max(\min(\cdot, -1), 1)$ .

Fifth, we use  $n_{ca} = 2$  and  $n_{embd} = 4$  in our simulations due to numerical reasons, resulting in  $N_0^{(s)} \cdot (N_1^{(s)} + 1) = 20$  sensory module parameters, and  $N_0^{(c)} \times (N_1^{(c)} + 1) = 39$  policy module parameters (accounting for the bias vectors), and thus in  $N_\theta = 59$  ANN parameters in total.

### C. Genetic algorithm and neuroevolution of single-agent policies with collective goals

*Genetic Algorithms* (GAs)<sup>25–27</sup> are heuristic optimization techniques inspired by the process of natural selection. In GAs, a set (or a population),  $\mathbf{X} = \{\theta_1, \dots, \theta_{N_p}\}$ , of sets of parameters (or individuals),  $\theta_i \in \mathbb{R}^{N_\theta}$ , is maintained and modified over successive iterations (or generations) to optimize an arbitrary objective function (or a fitness score),  $f(\theta_i) : \mathbb{R}^{N_\theta} \rightarrow \mathbb{R}$ <sup>33</sup>.

Many *Genetic-* or *Evolutionary Algorithm* (EA) implementations have been proposed<sup>28</sup>, which essentially follow the same biologically-inspired principles: Starting from an initial, often random population, high-quality individuals are selected (i) from the most recent generation for reproduction, depending on their associated fitness scores. Based on these selected high-fitness “parent” individuals, new “offspring” individuals are sampled, e.g., by genetic recombination (ii) of two mating parents,  $i, j$ , by randomly shuffling the elements (or genes) of their associated parameters, schematically expressed as  $\theta_o = \theta_i \oplus \theta_j$ . Such an offspring’s genome can be subjected to random mutations (iii), typically implemented by adding zero-centered Gaussian noise with a particular standard deviation,  $\xi$ , to the corresponding parameters,  $\theta_o \rightarrow \theta_o + \xi$ . The offsprings then either replace (iv) existing individuals in the population or are discarded depending on their corresponding fitness score,  $f(\theta_o)$ . In that way, the population is successively updated, and is thus guided towards high-fitness regions in the parameter space,  $\mathbb{R}^{N_\theta}$ , over many generations of successive reproduction cycles<sup>29,33</sup>.

Here, we utilize the “SimpleGA” implementation of D. Ha<sup>29</sup> (following steps (i)-(iv) above) to optimize the ANN parameters,  $\theta$ , of the single-bead agents of the here investigated  $N$ -bead microswimmers, see sections II and V B and figs. 1 and 7: After initializing the ANN parameters of a population of size  $N_p = 128$  by sampling from of a

zero-centered Gaussian of standard deviation  $\xi = 0.1$ , we successively (i) select at each generation the best 10% of individuals for the reproduction cycle (ii, iii) - according to the fitness score described in *section II* - and (iv) replace the remaining 90% of the population with sampled offsprings; we fix the mutation rate in step (iii) to  $\xi = 0.1$  and typically perform multiple independent GA runs for 200 – 300 generations (see fig. 1) each to ensure convergence of the evolved policies. For every parameter set  $\theta$  (per run, and per generation), we evaluate the fitness score as the average fitness of 10 independent episodes, each lasting for  $T = (400 - 800)$  environmental time-steps. For every episode, we randomize the respective  $N$ -bead swimmer's initial bead positions as  $x_i(0) \approx \mathcal{N}(\mu = i l_0, \sigma = r)$  drawn from a standard normal distribution centered around  $\mu = i l_0$  with a standard deviation of  $\sigma = R$ , and evaluate the episode fitness as mean center of mass velocity  $\bar{v}$  (see section II).

#### D. Swimming-gate Analysis

In section III C, we define a  $2\pi$ -period governing equation  $\hat{l}_i(t) = f_\ell((t - i\bar{\tau})\bar{\omega} + \phi)$  for the actual arm lengths  $l_i(t_k)$  for both type A and B  $N$ -bead microswimmers as a function of the mean angular velocity  $\bar{\omega} = \bar{\omega}(N)$  and the mean neighbor arm cross-correlation time  $\bar{\tau} = \bar{\tau}(N)$ . For all evolved type A and B microswimmer policies utilized in fig. 4 (A,B), we thus evaluate the corresponding mean angular velocity as  $\bar{\omega} = \frac{1}{N-1} \sum_{i=1}^{N-1} \omega_i$ , *i.e.*, by averaging the most dominant angular velocities  $\omega_i$  extracted for every arm length  $l_i(t)$  of a particular swimmer realization via Fourier transformation. We further define  $\bar{\tau} = \frac{1}{N-1} \sum_{i=1}^{N-1} \tau_i$ , with  $\tau_i$  being the optimal time delay between neighboring arm lengths  $l_i(t)$  and  $l_{i+1}(t + \tau_i)$  such that it maximizes their overlap,  $\frac{d}{dt} \int_0^T l_i(t) l_{i+1}(t + \tau) dt|_{\tau=\tau_i} = 0$ .

#### ACKNOWLEDGEMENTS

We thank Sebastian Risi for helpful discussions. BH acknowledges an APART-MINT fellowship of the Austrian Academy of Sciences. The computational results presented have been achieved (in part) using the Vienna Scientific Cluster 5.

#### AUTHOR DECLARATIONS SECTION

The authors have no conflicts to disclose.

#### DATA AVAILABILITY STATEMENT

Computational protocols and numerical data that support the findings of this study are shown in this article.

- <sup>1</sup>E. Lauga and T. R. Powers, "The hydrodynamics of swimming microorganisms," *Rep. Prog. Phys.* **72**, 096601 (2009).
- <sup>2</sup>J. Elgeti, R. G. Winkler, and G. Gompper, "Physics of microswimmers—single particle motion and collective behavior: a review," *Rep. Prog. Phys.* **78**, 56601 (2015).
- <sup>3</sup>C. Bechinger, R. Di Leonardo, H. Löwen, C. Reichhardt, G. Volpe, and G. Volpe, "Active Brownian Particles in Complex and Crowded Environments," *Rev. Mod. Phys.* **88**, 045006 (2016).
- <sup>4</sup>A. Zöttl and H. Stark, "Emergent behavior in active colloids," *J. Phys. Condens. Matter* **28**, 253001 (2016).
- <sup>5</sup>D. Bray, *Cell movements: from molecules to motility* (Garland Science, 2000).
- <sup>6</sup>K. Y. Wan, "Active oscillations in microscale navigation," *Animal Cognition* (2023), 10.1007/s10071-023-01819-5.
- <sup>7</sup>D. Jang, J. Jeong, H. Song, and S. K. Chung, "Targeted drug delivery technology using untethered microrobots: A review," *Journal of Micromechanics and Microengineering* **29** (2019), 10.1088/1361-6439/ab087d.
- <sup>8</sup>N. P. Barry and M. S. Bretscher, "Dictyostelium amoebae and neutrophils can swim," *Proc. Natl. Acad. Sci. U. S. A.* **107**, 11376–11380 (2010).
- <sup>9</sup>G. Noselli, A. Beran, M. Arroyo, and A. DeSimone, "Swimming Euglena respond to confinement with a behavioural change enabling effective crawling," *Nat. Phys.* **15**, 496–502 (2019).
- <sup>10</sup>C. E. Walczak and D. L. Nelson, "Regulation of dynein-driven motility in cilia and flagella," *Cell Motility and the Cytoskeleton* **27**, 101–107 (1994).
- <sup>11</sup>C. R. Reid, T. Latty, A. Dussutour, and M. Beekman, "Slime mold uses an externalized spatial "memory" to navigate in complex environments," *Proceedings of the National Academy of Sciences of the United States of America* **109**, 17490–17494 (2012).
- <sup>12</sup>J. Bongard, "Morphological change in machines accelerates the evolution of robust behavior," *Proceedings of the National Academy of Sciences of the United States of America* **108**, 1234–1239 (2011).
- <sup>13</sup>P. McMillen and M. Levin, "Collective intelligence: A unifying concept for integrating biology across scales and substrates," *Communications Biology* **7** (2024), 10.1038/s42003-024-06037-4.

- <sup>14</sup>Q. Wang, “Optimal strokes of low Reynolds number linked-sphere swimmers,” *Applied Sciences (Switzerland)* **9** (2019), 10.3390/app9194023.
- <sup>15</sup>A. C. H. Tsang, P. W. Tong, S. Nallan, and O. S. Pak, “Self-learning how to swim at low Reynolds number,” *Phys. Rev. Fluids* **5**, 074101 (2020).
- <sup>16</sup>B. Hartl, M. Hübl, G. Kahl, and A. Zöttl, “Microswimmers learning chemotaxis with genetic algorithms,” *Proceedings of the National Academy of Sciences* **118**, e2019683118 (2021), <https://www.pnas.org/doi/pdf/10.1073/pnas.2019683118>.
- <sup>17</sup>I. Jebellat, E. Jebellat, A. Amiri-Margavi, A. Vahidi-Moghaddam, and H. Nejat Pishkenari, “A Reinforcement Learning Approach to Find Optimal Propulsion Strategy for Microrobots Swimming at Low Reynolds Number,” *Robotics and Autonomous Systems* **175**, 104659 (2024).
- <sup>18</sup>A. Najafi and R. Golestanian, “Simple swimmer at low Reynolds number: Three linked spheres,” *Phys. Rev. E* **69**, 062901 (2004), arXiv:0402070 [cond-mat].
- <sup>19</sup>S. Kim and S. J. Karila, *Microhydrodynamics: Principles and Selected Applications* (Dover Publications Inc., Mineola, New York, 2005).
- <sup>20</sup>S. Muiños-Landin, A. Fischer, V. Holubec, and F. Cichos, “Reinforcement Learning of Artificial Microswimmers,” *Sci. Robotics* **6**, eabd9285 (2021).
- <sup>21</sup>K. Hornik, M. Stinchcombe, and H. White, “Multilayer feedforward networks are universal approximators,” *Neural networks* **2**, 359–366 (1989).
- <sup>22</sup>A. Li, Y. X. Luo, Y. Liu, Y. Q. Xu, F. B. Tian, and Y. Wang, “Hydrodynamic behaviors of self-propelled sperms in confined spaces,” *Engineering Applications of Computational Fluid Mechanics* **16**, 141–160 (2022).
- <sup>23</sup>A. Mordvintsev, E. Randazzo, E. Niklasson, and M. Levin, “Growing neural cellular automata,” *Distill* **5** (2020), 10.23915/distill.00023.
- <sup>24</sup>S. Pontes-Filho, K. Walker, E. Najarro, S. Nichele, and S. Risi, “A unified substrate for body-brain co-evolution,” in *From Cells to Societies: Collective Learning across Scales* (2022).
- <sup>25</sup>J. H. Holland, *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence*. (1975).
- <sup>26</sup>M. Zbigniew, “Genetic algorithms + data structures= evolution programs,” *Computational statistics* **24**, 372–373 (1996).
- <sup>27</sup>Z. Michalewicz and M. Schoenauer, “Evolutionary Algorithms for Constrained Parameter Optimization Problems,” *Evolutionary Computation* **4**, 1–32 (1996).
- <sup>28</sup>S. Katoch, S. S. Chauhan, and V. Kumar, “A review on genetic algorithm: past, present, and future,” *Multimedia Tools and Applications* **80**, 8091–8126 (2020).
- <sup>29</sup>D. Ha, “Evolving stable strategies,” *blog.otoro.net* (2017).
- <sup>30</sup>P. W. Anderson, “More is different,” *Science* **177**, 393–396 (1972), <https://www.science.org/doi/pdf/10.1126/science.177.4047.393>.
- <sup>31</sup>M. Levin, “The computational boundary of a “self”: Developmental bioelectricity drives multicellularity and scale-free cognition,” *Frontiers in Psychology* **10** (2019), 10.3389/fpsyg.2019.02688.
- <sup>32</sup>S. Mishra, W. M. Van Rees, and L. Mahadevan, “Coordinated crawling via reinforcement learning,” *Journal of the Royal Society Interface* **17** (2020), 10.1098/rsif.2020.0198.
- <sup>33</sup>B. Hartl, S. Risi, and M. Levin, “Evolutionary implications of self-assembling cybernetic materials with collective problem-solving intelligence at multiple scales,” *Entropy* **26** (2024), 10.3390/e26070532.
- <sup>34</sup>B. Nasouri, A. Vilfan, and R. Golestanian, “Efficiency limits of the three-sphere swimmer,” *Physical Review Fluids* **4**, 1–9 (2019).
- <sup>35</sup>N. Meyer-Vernet and J.-P. Rospars, “How fast do living organisms move: Maximum speeds from bacteria to elephants and whales,” *American Journal of Physics* **83**, 719–722 (2015).
- <sup>36</sup>A. Daddi-Moussa-Ider, M. Lisicki, and A. J. Mathijssen, “Tuning the upstream swimming of microrobots by shape and cargo size,” *Phys. Rev. Appl.* **14**, 024071 (2020).
- <sup>37</sup>E. Gutman and Y. Or, “Optimizing an undulating magnetic microswimmer for cargo towing,” *Physical Review E* **93**, 063105 (2016).
- <sup>38</sup>B. Felderhof, “Collinear swimmer propelling a cargo sphere at low Reynolds number,” *Physical Review E* **90**, 053013 (2014).
- <sup>39</sup>M. Levin, “Darwin’s agential materials: evolutionary implications of multiscale competency in developmental biology,” *Cellular and Molecular Life Sciences* **80** (2023), 10.1007/s00018-023-04790-z.
- <sup>40</sup>D. Patra, S. Sengupta, W. Duan, H. Zhang, R. Pavlick, and A. Sen, “Intelligent, self-powered, drug delivery systems,” *Nanoscale* **5**, 1273–1283 (2013).
- <sup>41</sup>R. Golestanian and A. Ajdari, “Analytic results for the three-sphere swimmer at low Reynolds number,” *Phys. Rev. E* **77**, 036308 (2008).
- <sup>42</sup>F. M. Kievit and M. Zhang, “Cancer nanotheranostics: improving imaging and therapy by targeted delivery across biological barriers,” *Adv. Mater.* **23**, H217–H247 (2011).
- <sup>43</sup>A. V. Singh, M. H. D. Ansari, P. Laux, and A. Luch, “Micro-nanorobots: important considerations when developing novel drug delivery platforms,” *Expert Opin. Drug. Deliv.* **16**, 1259–1275 (2019).
- <sup>44</sup>J. Cooke, “Scale of body pattern adjusts to available cell number in amphibian embryos,” *Nature* **290**, 775–778 (1981).
- <sup>45</sup>C. G. Langton, *Artificial life: An overview* (MIT Press, 1997).
- <sup>46</sup>C. Fields and M. Levin, “Regulative development as a model for origin of life and artificial life studies,” *Biosystems* **229**, 104927 (2023).
- <sup>47</sup>W. S. McCulloch and W. Pitts, “A logical calculus of the ideas immanent in nervous activity,” *The Bulletin of Mathematical Biophysics* **5**, 115–133 (1943).
- <sup>48</sup>M. L. Minsky, *Theory of neural-analog reinforcement systems and its application to the brain-model problem* (Princeton University, 1954).
- <sup>49</sup>M. Minsky, “Steps toward artificial intelligence,” *Proceedings of the IRE* **49**, 8–30 (1961).
- <sup>50</sup>F. Rosenblatt *et al.*, *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms*, Vol. 55 (Spartan books Washington, DC, 1962).
- <sup>51</sup>M. Minsky and S. Papert, “Perceptron: an introduction to computational geometry,” The MIT Press, Cambridge, expanded edition **19**, 2 (1969).
- <sup>52</sup>D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “Learning representations by back-propagating errors,” *nature* **323**, 533–536 (1986).
- <sup>53</sup>Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *Nature* **521**, 436–444 (2015).
- <sup>54</sup>I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).
- <sup>55</sup>R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction* (The MIT Press, 2018).
- <sup>56</sup>J. Von Neumann, A. W. Burks, *et al.*, “Theory of self-reproducing automata,” *IEEE Transactions on Neural Networks* **5**, 3–14 (1966).

- <sup>57</sup>X. Li and A. G.-O. Yeh, “Neural-network-based cellular automata for simulating multiple land use changes using gis,” *Int. J. Geogr. Inf. Sci.* **16**, 323–343 (2002).
- <sup>58</sup>Y. Tang and D. Ha, “The sensory neuron as a transformer: Permutation-invariant neural networks for reinforcement learning,” in *Advances in Neural Information Processing Systems*, edited by A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan (2021).