Evolutionary multiobjective design of a flexible caudal fin for robotic fish

2015 Bioinspir. Biomim. 10 065006

(http://iopscience.iop.org/1748-3190/10/6/065006)

Bioinspiration & Biomimetics

PAPER

RECEIVED
28 February 2015

REVISED
13 September 2015

ACCEPTED FOR PUBLICATION
22 September 2015

PUBLISHED
25 November 2015

Evolutionary multiobjective design of a flexible caudal fin for robotic fish

Anthony J Clark[1], **Xiaobo Tan**[2] **and Philip K McKinley**[1]

[1] Department of Computer Science and Engineering, Michigan State University, East Lansing, MI, USA
[2] Department of Electrical and Computer Engineering, Michigan State University, East Lansing, MI, USA

E-mail: ajc@msu.edu

Keywords: evolutionary multiobjective optimization, robotic fish, energy usage, compliant fins

Abstract

Robotic fish accomplish swimming by deforming their bodies or other fin-like appendages. As an emerging class of embedded computing system, robotic fish are anticipated to play an important role in environmental monitoring, inspection of underwater structures, tracking of hazardous wastes and oil spills, and the study of live fish behaviors. While integration of flexible materials (into the fins and/or body) holds the promise of improved swimming performance (in terms of both speed and maneuverability) for these robots, such components also introduce significant design challenges due to the complex material mechanics and hydrodynamic interactions. The problem is further exacerbated by the need for the robots to meet multiple objectives (e.g., both speed and energy efficiency). In this paper, we propose an evolutionary multiobjective optimization approach to the design and control of a robotic fish with a flexible caudal fin. Specifically, we use the NSGA-II algorithm to investigate morphological and control parameter values that optimize swimming speed and power usage. Several evolved fin designs are validated experimentally with a small robotic fish, where fins of different stiffness values and sizes are printed with a multi-material 3D printer. Experimental results confirm the effectiveness of the proposed design approach in balancing the two competing objectives.

1. Introduction

Similar to live fish, robotic fish accomplish swimming by deforming their bodies or fin-like appendages. This form of locomotion offers certain key advantages relative to traditional propeller-driven aquatic vehicles. First, robotic fish are potentially more maneuverable, which is critical when operating in cluttered underwater environments [1, 2]. Second, since robotic fish produce very low acoustic noises and exhibit wake signatures similar to those of live fish, they are less intrusive to aquatic ecosystems and offer stealth in security-related applications. Third, with fin/body movements occurring at relatively low frequencies (typically only a few Hz), these systems are less likely to harm aquatic animals or become jammed with foreign objects. Given these characteristics, robotic fish are anticipated to play an important role in environmental monitoring [1], inspection of underwater structures [3], tracking of hazardous wastes and oil spills [4], and the study of natural systems [5–8]. However, while investigations of robotic fish have produced many advances over the past two decades [9–16], robotic fish still do not rival their biological counterparts in terms of swimming abilities.

One approach to improving performance is to integrate flexible, or *compliant*, materials into the morphology (structure) of a robot [17, 18]. Passive, compliant components are intended to partially compensate for actuation capabilities that are primitive relative to those of biological organisms. However, incorporating such materials into a robot poses numerous challenges in the optimal design and control of the system, due to complex material mechanics, hydrodynamics, and interactions between the flexible and rigid elements [19, 20]. Moreover, the interaction between geometry and flexibility further complicates the underlying dynamics. For example, determining the so-called 'optimal' *foil*-shape for robotic fish tail fins depends

on a given performance metric (e.g., speed, power) [21].

Several recent studies have addressed the design of aquatic robots with compliant components [12, 22–24]. For example, Low and Chong [12] used statistical methods to investigate the effect of control and morphological design parameters on the resulting thrust of a robotic fish with a compliant caudal fin. Esposito *et al*[24] performed a similar analysis of a caudal fin with six independently actuated fin rays. While studying a single kinematic parameter, the phase difference between the driving angle at the base of a flexible caudal fin and the fin-bending angle, Park *et al* [22] discovered that maximal thrust occurs at a specific phase difference, even when the morphology of the caudal fin is changed. Although many recent works in this area focus on flexible caudal fins, as evidenced by the studies mentioned above, Daou *et al* [25] have investigated a compliant body, where both the head and caudal fin were rigid.

Existing work on optimizing performance of robotic fish with flexible fins or bodies has typically dealt with a single optimization objective, for example speed or thrust. However, in practical applications robotic fish are often required to meet multiple objectives, either simultaneously or within different tasks or environments. For example, while speed is in general an important specification, energy efficiency (and thus operating duration) is often equally as important. Maneuverability, the ability of the robot to make tight turns or deal with disturbances, is also a particularly significant objective.

In this paper we propose an evolutionary multi-objective optimization (EMO) approach to the design and control of flexible fins for robotic fish. For the purpose of demonstration, we focus on the caudal fin and two competing objectives: speed and power consumption. In particular, we explore the interactions between the stiffness, size, and the control pattern for the flexible fin, and we investigate how the speed performance and energy usage can be balanced. Some situations may require speed be sacrificed for efficiency (e.g., the system may need to return to port due to a low battery), while in other situations speed may be paramount (e.g., the system may need to escape a hazard).

Finding combinations of morphological and control parameters that effectively balance speed and energy consumption is challenging. In robotic fish that incorporate compliant materials, morphological and control parameters are highly interrelated. Specifically, any change to the flexibility of the caudal fin will require a corresponding change to the control signal to ensure that the frequency of oscillation matches the natural motion of the fin.

Evolutionary algorithms (EAs) have proven to be effective for similar problems. For example, Gomez and Miikkulainen evolved neural network controllers to guide *finless* rockets [26]. Generally, EAs operate on a population of individuals, each representing a solution to the problem under study. The *genome* of each individual comprises a set of parameter values. In optimizing a robot, for example, these parameters might include values for the robot's controller as well as characteristics of the robot's morphology (e.g., oscillating frequency of the motor and the dimensions of the fin). The initial population of individuals is either generated randomly or seeded around a point of interest, such as a solution to the problem that is known to be effective. A *fitness function* measures the effectiveness of a solution, such as the speed or distance traveled in a fixed amount of time. In evolutionary robotics (ER) [27–32], the fitness of an individual is often evaluated with respect to one or more tasks performed in a simulated environment. After evaluating all individuals in the population, solutions exhibiting higher fitness are preferentially selected to create the next generation by combining and mutating their parameters with genetic operators (e.g., crossover and mutation). This cycle is typically repeated until one of the following conditions is met: (1) the fitness values plateau, (2) a maximum number of generations is reached, (3) or a maximum amount of real time has passed.

One of the primary benefits of ER methods is that they can optimize both control and morphology, which typically leads to better coupling (i.e., better performance) between behavior and physical form [33–36]. Evolved controllers usually take the form of an array of control parameters (as is done for this study) or use artificial neural networks [26, 37, 38], central pattern generators [39, 40], or genetic programs [41, 42]. Recently, ER has been applied to problems in *soft robotics* [19]. Similar to the robotic fish developed for this study, soft robots comprise (or contain) malleable, flexible components, which are meant to improve performance and/or safety [43–47]. For example, Cheney *et al* [48] evolved locomotion for voxel-based simulated robots, where each robot is made up of a 3D-grid of cubic voxels and each voxel can be evolved with different material properties. Cheney's study demonstrates that evolving soft robots with a generative encoding, based on principles from developmental biology, dramatically improves locomotion when compared to direct encodings.

EMO algorithms operate using similar principles. However, unlike traditional approaches to managing multiple objectives in which fitness values are a weighted sum of different goals (a technique sometimes referred to as scalarization) [49], EMO fitness functions return a sequence of values, where each value represents fitness with respect to a different objective. Instead of locating a single optimal set of parameter values, EMOs converge to a *set* of Pareto-optimal solutions. Individuals belonging to a Pareto front are said to be *nondominated*; that is, each of the solutions is optimal with respect to some combination of the objectives. The most common EMO algorithms, such

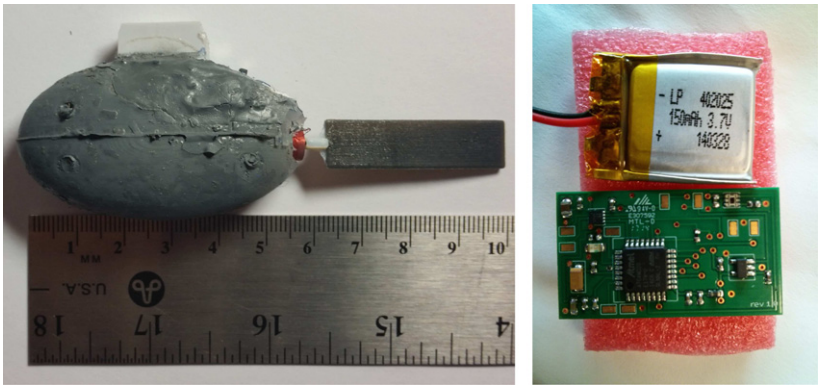

Figure 1. (a) Small robotic fish cast from liquid rubber; the body has been painted gray and the printed caudal fins are detachable. (b) The device's custom PCB and rechargeable battery.

as NSGA-II and SPEA2, use an elitism approach for driving solutions toward the optimal Pareto front, and a niching or crowding mechanism to ensure that the entire set of Pareto-optimal solutions can be found [50, 51]. Indicator-based EMOs also produce a set of solutions, but do so by maximizing an indicator variable, which acts as a single objective for evolving the entire population [52, 53]. One of the most popular indicator variables is *hypervolume* [54]. By maximizing the hypervolume of a population, the algorithm effectively drives search toward the user-defined goals while maintaining a distance between evolved individuals in the solution space. The advantages of EMO algorithms, when compared to single objective EAs and parameter sweeps, include: (1) locating a Pareto front with fewer evaluations, (2) automatically handling constraints, (3) not needing to specify the relative importance among multiple objectives, and (4) automatically sorting solutions according to feasibility and domination. For this study, we apply the NSGA-II algorithm [50], which is widely used in both research and real-world applications. Compared with other EMO algorithms, the main advantages of NSGA-II include a faster sorting operation and a more effective method for maintaining diversity (i.e., reducing premature convergence) [51, 55–57].

Optimized designs obtained via the NSGA-II algorithm are validated on a robotic fish with a flexible caudal fin driven by an electromagnetic actuator. Evolved fin designs are prototyped with a multi-material 3D printer. Experimental results of the robot swimming with different fins demonstrate a trend in the speed performance consistent with the corresponding simulation results, although there is some discrepancy in the exact speed values between the experiments and simulation.

Preliminary investigations related to this work are described in two earlier papers [58, 59]. In [58], we explored how a conventional genetic algorithm could be applied to optimize morphological characteristics, including caudal fin flexibility, and control patterns for a robotic fish prototype. However, that system had limited capabilities (i.e., no sensory feedback, no communication abilities, and a low-power micro-controller), and optimization did not address multiple objectives. In [59], we described our initial approach to applying EMO methods to robotic fish. However, that investigation also used a platform with limited sensing capabilities. The electromagnetically driven robotic fish in this work, shown in figure 1, is smaller, has more computational power and sensing capabilities, and enables a direct calculation of energy usage, as discussed in section 2. We have also refined and enhanced the simulation modeling for flexible fin dynamics, and we have conducted more physical validation trials with the new robotic fish prototype. Finally, whereas in [59] we focused on useful mechanical power, here we are able to optimize for both the robot's speed and the average electrical power expended. As a result, the analysis of Pareto-optimal solutions yields results not apparent in our earlier studies.

The remainder of the paper is organized as follows. In section 2 we describe the robotic fish prototype, simulation model, and optimization algorithm. In section 3 we present the results of evolutionary optimization. In section 4 we provide details for fabricating and testing flexible caudal fins, and report results of the physical validation experiments. Finally, we present our conclusions in section 5.

2. Robotic fish design and simulation

In this section we provide details of the target robotic fish, the simulation model, and the evolutionary algorithm.

2.1. Robotic fish

Figure 1 shows the robotic fish used to test and validate the methods proposed in this paper. The caudal fin is detachable, enabling us to test many fin designs. This

Figure 2. (a) A solidworks model of the robotic fish mold. (b), (c) Two images of the 3D-printed, clear plastic mold used during the casting process.

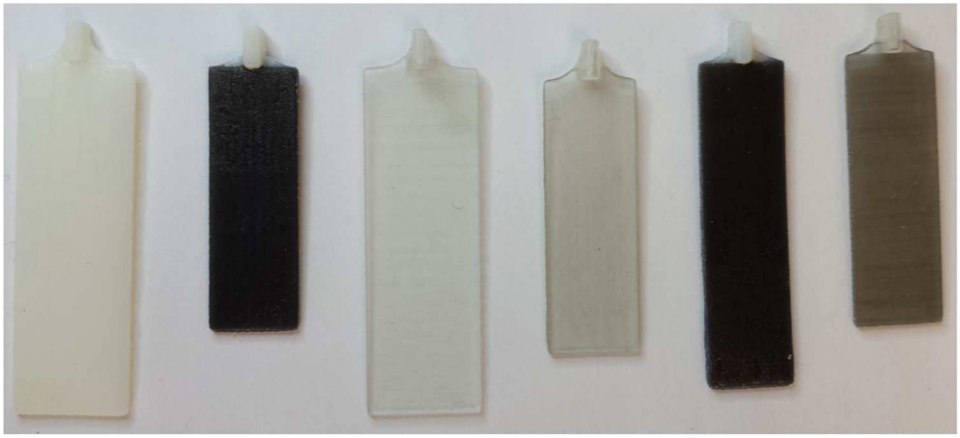

Figure 3. A photograph of several 3D-printed caudal fins. Each fin has different morphological characteristics: length, height, and flexibility.

robotic fish is intended to operate either autonomously or via remote control for up to three hours under normal conditions (i.e., non-continuous communication, average actuator usage). The device is powered by a 150 mAh lithium–ion polymer battery, which provides over two hours of continuous operation under maximum load (including wireless communication, sensing, and actuation).

Body and fin fabrication: The body of the robotic fish is designed to be as small as possible, while incorporating all components necessary for untethered operation. The body is cast from liquid rubber (Smooth-On Ecoflex®00-30), which results in a soft, 'stretchy' form. All electrical and mechanical components are placed in a 3D-printed mold and the liquid rubber is poured around them. The mold (see figure 2) was produced with an Objet350 Connex printer. A photograph of several 3D-printed caudal fins can be seen in figure 3. Details regarding fin design and their fabrication process will be discussed in section 4.

Custom printed circuit board (PCB): To control the robotic fish, we designed a custom PCB, pictured in figure 1(b). The PCB includes a 32-bit ARM microcontroller (Atmel SAM D20), a six-axis inertial measurement unit (IMU) (InvenSense MPU-6050), two light sensors (Intersil ISL29101), and wireless communication (Nordic Semiconductor nRF24L01-+). The microcontroller is capable of executing complex adaptive control algorithms while filtering sensory data. The IMU enables the device to measure its linear and angular accelerations, which can be filtered to provide estimates of velocity. Wireless communication allows the device to be controlled and have its software updated remotely, as well as deliver sensed information to a base computer.

Electromagnetic actuator: The electromagnetic actuator is depicted in figure 4. The actuator comprises a coil of magnet wire (9.5 mm outside diameter), a neodymium permanent magnet, an external centering magnet, and an attachment point for a caudal fin. To operate the actuator, a voltage is applied across the coil's terminals. The coil creates an electromagnetic field that exerts a torque on the permanent magnet, causing its poles to align with the magnetic field. The actuator used in this study has a fixed maximum amplitude of 38° and a centering magnet that causes the actuator to return to its center when no voltage is applied to the coil. Equation (1) is used to calculate the angular acceleration of the permanent magnet resulting from an applied voltage:

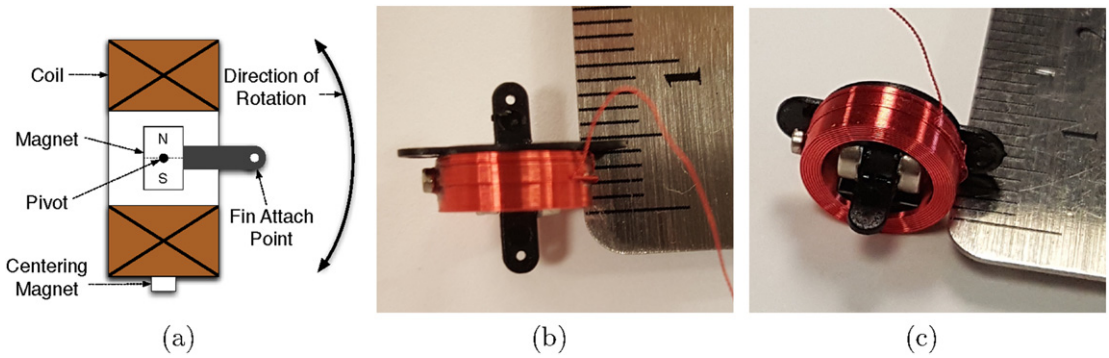

Figure 4. (a) Top-view diagram of the actuator, and (b), (c) photographs of the electromagnetic actuator.

$$\alpha = \frac{V\mu_0 NlR}{I_{\text{effective}}}, \quad (1)$$

where α is the angular acceleration, V is the voltage applied across the coil, μ_0 is the magnetic constant (or the permeability of free space), N, l, and R are the number of turns, length, and resistance of the coil, and $I_{\text{effective}}$ is the moment of inertia for the permanent magnet with an attached fin. This device exhibits a torque of roughly 500 μN m with a volume of 35 mm^3, which yields a torque-per-volume of 15 μN per mm^3. In contrast, a typical commodity micro servo motor provides a torque of approximately 75 mN m with a volume of 2200 mm^3 resulting in 35 μN per mm^3 torque-per-volume. For our purposes, alternatives to the electromagnetic actuator are either too large (commodity servo motors), require higher voltages (piezoelectric motors), or generate less torque (electropolymers and shape memory alloys) [60]. Likewise, systems that need external magnetic fields, such as the microrobots developed for medical applications [61], are too limiting in terms of the types of environments in which a robot could be deployed.

Electromagnetic actuators have been used as motors for robotic fish by a few other research groups. To support their fuel cell studies, Takada *et al* [62] used a similar actuator for a robotic fish; this device was 10 cm in length, had diving capabilities, but did not include any sensing. Shin *et al* [63] used a comparable electromagnetic mechanism for robotic tadpoles, which were less than 3 cm in length but did not include any sensing or complex control capabilities.

Control signal: The actuator is controlled by supplying a positive, negative, or zero voltage to the coil. For this study, we do not consider voltages other than 0, 3.3 or −3.3 volts. This leaves two control parameters for forward thrust: an oscillating frequency and a pulse-width-ratio (PWR). Similar to a duty-cycle, PWR defines the fraction of time the control signal is active during a given period. We note, however, that PWR has negative components, as depicted in figure 5. For the control signal example shown, PWR is set to 0.4, which results in an applied voltage that is active only 40 percent of the period. With this setup, adjusting PWR is the only way to alter energy consumption. For example, a PWR of 0.8 will result in twice the amount of energy consumed when compared to the example signal.

PWR can range from 0 to 1 and the resulting control signal will be a constant zero or a square wave, respectively. How to choose these two parameters (frequency and PWR) depends on the dimensions of the caudal fin and the desired balance between speed and energy consumption. For instance, it is rarely useful to set the PWR to 1, as doing so will result in wasted energy. Specifically, the actuator will consume energy while actively 'pinning' the fin to one side, which does not generate any additional thrust. Moreover, the time required for the fin to reach its maximum amplitude depends on caudal fin dimensions. Since the actuator will always generate the same torque, using a larger fin will result in lower angular acceleration compared to fins with less surface area because the fin is 'pushing' on a larger volume of water. While conducting initial tests of the robotic fish, we found that the actuator was effective only for fins with a surface area less than 5 cm^2.

2.2. Dynamic model

Evolutionary optimization is usually conducted with the aid of a simulation environment, which provides a means to execute a large number of evaluations in a short amount of time. The main drawback of simulation, however, lies with the so-called *reality gap* [64], which arises when solutions that appear to work well in a simulated environment perform differently in a physical environment. In general, higher accuracy simulations tend to have a better chance of crossing the reality gap [65]. In this study, we use a dynamics model developed by Wang *et al* [66] based on Lighthill's large-amplitude elongated body theory of locomotion [67]. This model has proven to be both accurate and computationally efficient [58, 66]. Specifically, simulating 10 s of time takes on average 0.5 s of computation time on a single processor in our

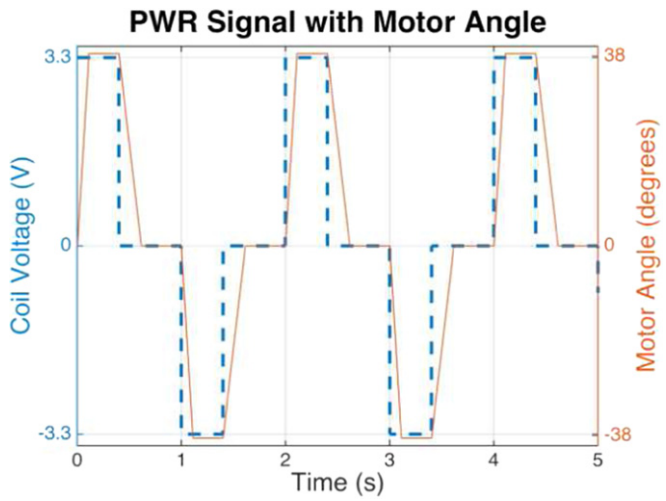

Figure 5. An example of the control voltage signal with the resulting simulated motor angle. The dashed, blue line depicts the control signal with a frequency of 0.5 Hz and a PWR of 0.4 (PWR indicates the fraction of time spent at either 3.3 or −3.3 V). The solid, orange line is the angle of the motor, which has a rise time related to the applied torque, and a fall time relating to the centering torque, due to the centering permanent magnet.

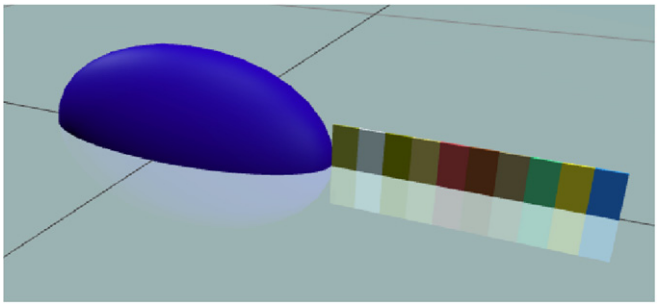

Figure 6. Rendering of the simulated robotic fish. Individual fin segments appear in different colors.

compute cluster; each processor executes at 2.4 GHz and has 64 GBs of RAM.

Wang's model assumes that all motion is constrained to a two-dimensional plane and that fins are rectangular in shape. The model is based on the *added-mass effect*, for which rigid bodies appear more massive due to surrounding water. Specifically, the dynamic model calculates thrust forces as if a volume of water were pushing on the fin in direct opposition to its motion. A critical aspect of simulation is modeling of compliant caudal fin dynamics. Flexibility is modeled as multiple rigid segments connected by springs and dampers. The spring coefficient between two consecutive segments depends on the stiffness of the caudal fin. Individual rigid segments can be seen in the visualization rendered in figure 6. For this study, all simulations were conducted with a ten-segment caudal fin.

Figure 7 depicts the hydrodynamic forces acting on a robotic fish, where a flexible caudal fin is modeled as three rigid segments for the purpose of illustration. The force acting on each fin segment f_i can be calculated independently with

$$\vec{f}_i(\tau) = \begin{pmatrix} f_{i,x}(\tau) \\ f_{i,y}(\tau) \end{pmatrix} = -m\frac{\mathrm{d}}{\mathrm{d}t}\left(v_\perp \hat{n}\right), \qquad (2)$$

where m denotes mass per unit length, τ is the location on the fin where the force acts, and $\hat{n}$ and $v_\perp$, respectively, are the unit direction and velocity perpendicular to the fin. The tip of the final segment experiences an additional force described by

$$\vec{f}_{\mathrm{L}} = \begin{pmatrix} f_{\mathrm{L},x} \\ f_{\mathrm{L},y} \end{pmatrix} = \left[-\frac{1}{2} m v_\perp^2 \hat{m} + m v_\perp v_\parallel \hat{n} \right]_{\tau=L}, \qquad (3)$$

where $\tau = L$ represents the posterior end of the fin, and $\hat{m}$ and $v_\parallel$, respectively, are the unit direction and velocity parallel to the fin. The resulting thrust force F_{T} is simply the sum of all segment forces f_i and the tip force f_{L}. Finally, the details for the evaluation of the drag force F_{D}, lift force F_{L}, and the drag moment M_{D} on the robotic fish body, as well as the rigid body dynamics, can be found in [66].

2.3. Evolutionary optimization

The dynamic simulation model described above takes several parameters, including the frequency and PWR of the control signal and the length, height, and

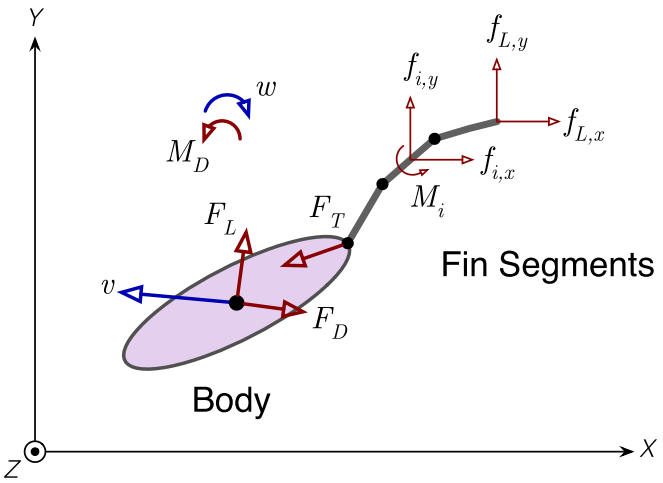

Figure 7. Graphical representation of the simulated hydrodynamics for a robotic fish with a rigid body and flexible caudal fin. Linear velocity v and angular velocity w are the result of thrust force F_T, drag force F_D, lift force F_L, and drag moment M_D. F_T is calculated as the sum of all forces acting on the fin segments.

Table 1. Range of evolved parameters.

	Min	Max
Frequency (Hz)	0.1	5.0
Pulse-width-ratio	0.1	1.0
Fin length (cm)	1.0	4.0
Fin height (cm)	0.3	1.3
Spring constant (Nm rad^{-1})	5e-5	1e-1

flexibility (which dictates the spring and damping coefficients for fin segments) of the caudal fin. Evolved genomes comprise values for these parameters, and the allowable range for each is listed in table 1. Values in the table were determined by testing the physical limitations of the physical robotic fish. For instance, if the frequency is near or above 5 Hz, the caudal fin will not have a sufficient amount of time to rotate before the actuator reverses direction. In effect, the caudal fin simply vibrates, producing very little thrust. The limits on spring coefficients allow for caudal fins to behave similarly to materials as flexible as rubber or as stiff as hard plastic.

NSGA-II configuration: As mentioned in section 1, the NSGA-II algorithm [50] was chosen to conduct the evolutionary multiobjective design. NSGA-II efficiently sorts a combined population of parent and children (created using genetic operators) into different ranks of nondominated Pareto fronts. Selection proceeds by accepting individuals from each rank in sequence until N individuals are selected in total. Crowding distance is used as a tie-breaker when an entire rank cannot be selected (because it would require accepting more than N individuals). Crowding distance is calculated as the Euclidean distance between the fitness vectors of two individuals. This method ensures that the elite (best) individuals are retained because all individuals in the first rank are Pareto-optimal. For evolving genomes comprising real values, NSGA-II requires the user to set the following four parameters (values chosen for our experiments are in parentheses): the probabilities of crossover (90%) and mutation (20%), and the distribution index for both simulated binary crossover (20) and polynomial mutation (20). Starting from values recommended by Deb *et al* [50], these values were determined experimentally and cause the populations to converge in relatively few generations (i.e., within 100 generations, which corresponds to approximately 6400 fitness evaluations).

Fin constraint: Along with evolving the above parameters as real-valued numbers, NSGA-II also accommodates constraints. In our study, the dynamic model is only valid for an elongated fin in which fin length is at least three times the fin height:

$$\text{length} - 3\ \text{height} \geqslant 0, \tag{4}$$

where *length* and *height* refer to the evolved dimensions of the robotic fish caudal fin. In NSGA-II this limitation is configured as a constraint, which enables the algorithm to smoothly follow a gradient from *infeasible* (i.e., a solution that violates a constraint) to *feasible* solutions.

Fitness evaluation: Each individual in the population is evaluated for 10 s of simulation time; however, only the second half of this period determines fitness. This setup allows the robotic fish to reach a cruising speed and final heading, with average speed calculated over the final 5 s. In our preliminary study [59], efficiency was defined as the ratio between useful and total power, where useful power was calculated using the product of the total propulsive force projected onto the trajectory of travel and the travel speed, and the total power was calculated using the sum of all mechanical power exerted by the caudal fin. These calculations require instantaneous power to be calculated at every simulation time step. In contrast, since PWR of the control signal is known, in the current study

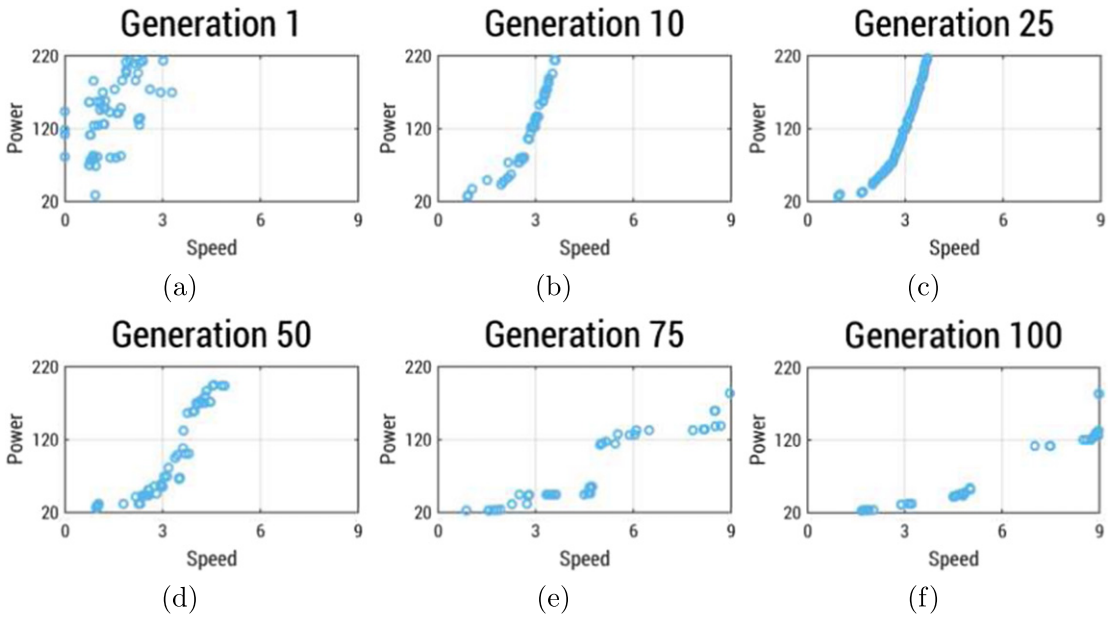

Figure 8. The evolutionary history of a single replicate simulation run. Every individual (including dominated individuals) in the population at a given generation is plotted in each figure. For all figures, the units for speed and power are cm s^{-1} and mW, respectively.

average electrical power is directly calculated using the following equation:

$$P_{avg} = P_{MAX} * \text{PWR}, \quad (5)$$

where P_{avg} and P_{MAX} refer to the average and maximum instantaneous electrical power delivered to the motor, respectively. Since P_{MAX} and PWR are known values, a control signal's energy consumption can be calculated directly.

3. EMO simulation results

We conducted 25 replicate EMO simulations trials, each with populations of 64 individuals evolving for 100 generations. Simulation parameters for the robotic fish dynamics, other than those related to the flexible fin and the control signal, are based on the robotic fish prototype (see section 2.1) to be used for physical validation. Most replicates (23 of 25) converged to nearly identical Pareto fronts. Figure 8 displays the evolutionary history of one replicate simulation. The population is randomly initialized (figure 8(a)), and then evolves toward the final, optimal Pareto front (figure 8(f)).

Figure 9(a) plots the average power and speed of every evolved (feasible) individual from all replicate simulations, with the Pareto-optimal individuals appearing in colors other than light gray. Jitter appears in the final Pareto fronts (some non-gray points appear to be dominated) because replicate simulations do not find identical sets of Pareto-optimal solutions. The results are grouped into three clusters. First is the *energy cluster*, which includes individuals that on average consume low amounts of power. These individuals appear in the lower, left-hand section of the figure, and have speeds less than 5 cm s^{-1} and consume less than 60 mW of power on average. Second, the *speed cluster* includes individuals that are in the middle, right-hand section of the figure, and which consume roughly twice the amount of power but also swim twice as fast. The *local-optimum cluster* is the set of individuals that have converged to a local optimum and appear in the upper, center section of the plot. The replicate run depicted in figure 8 appears to have encountered the same local optimum (shown in plot 8(c)) but was able to find an evolutionary path toward the final Pareto front.

These three clusters are highlighted in figures 9(b)–(d). The closer views show that within each cluster a smaller Pareto front is formed (maximizing speed and minimizing power usage). Inspecting how each of the parameters evolved gives insight into what created these distinct clusters. For example, figure 10 shows that caudal fin morphology (length, height, and flexibility) remains fairly consistent for all Pareto-optimal individuals even across clusters; the data shown in this figure is scaled between 0 and 1 so that each parameter can be plotted on the same axis.

Convergence of morphology parameters indicates that for this particular robotic fish an ideal caudal fin is approximately 3.4 cm in length, 1 cm in height, and has the minimum allowed spring constant (i.e., a flexibility resembling a rubber material). Since the fins differ only slightly among solutions, we can infer that the control parameters account for most of the diversity in the final populations. A pairwise comparison of the parameters' variances using the Brown–Forsythe equality of variances test (and the Bonferroni

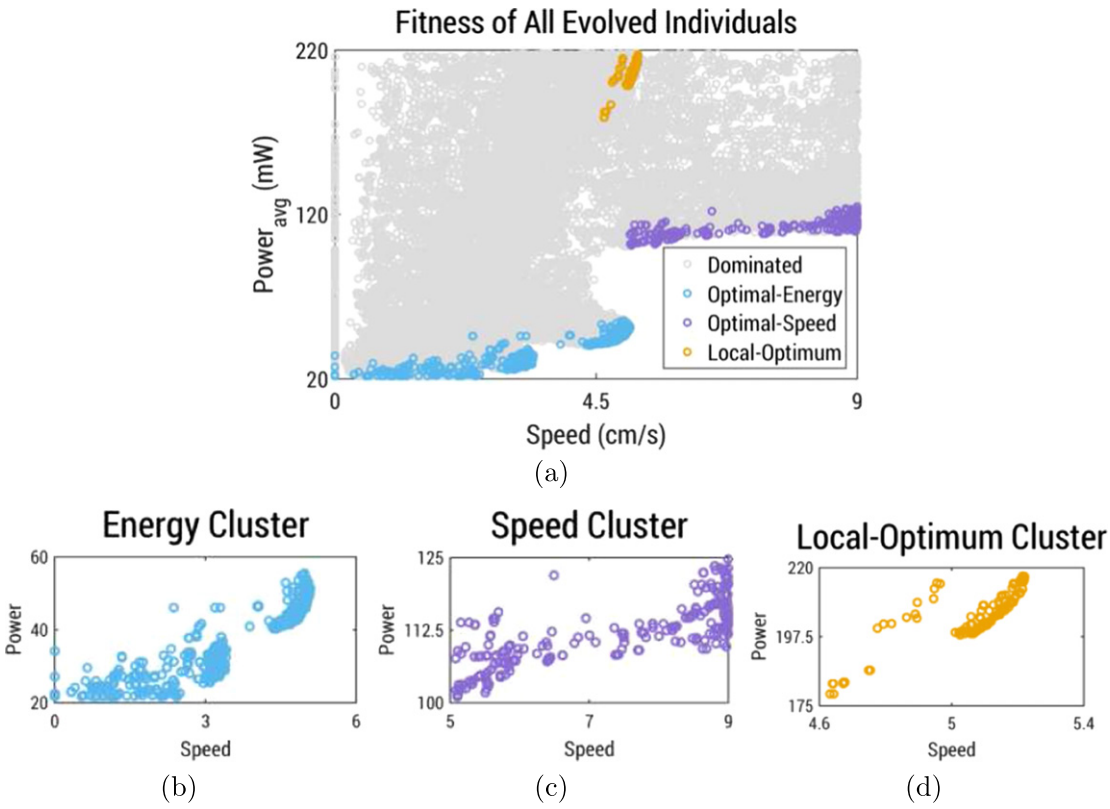

Figure 9. (a) Every evolved individual among all 25 replicate simulations (excluding infeasible solutions). Gray markers denote 'dominated' individuals and non-gray markers denote individuals that were Pareto-optimal for a given replicate experiment. Three clusters have been identified and given different colors. The remaining plots display individuals belonging to the energy cluster (b), speed cluster (c), and local-optimum cluster (d). Please note that the cluster plots do not share the same ranges for their axes.

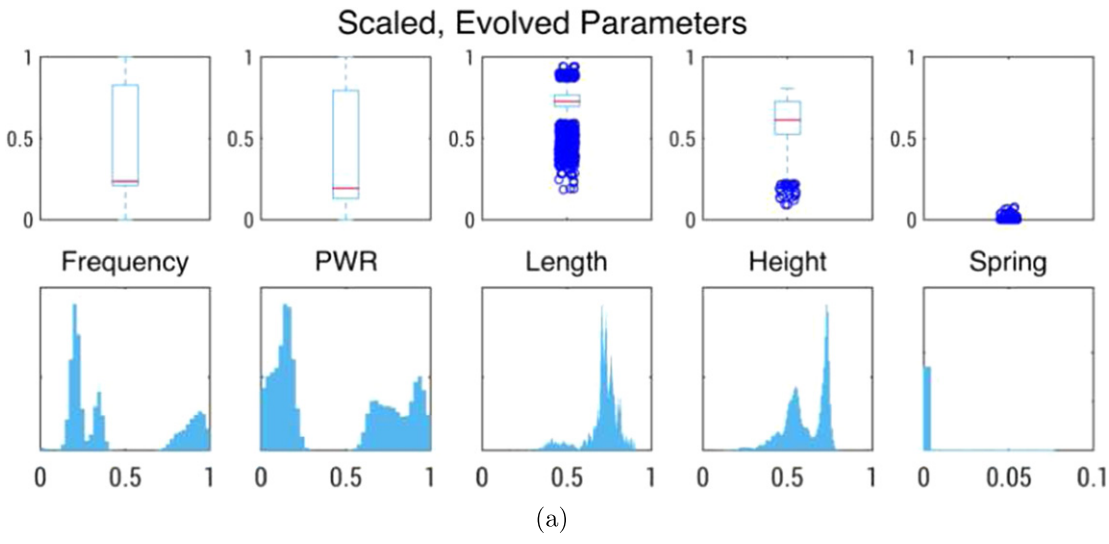

Figure 10. Box plots (top row) and histograms (bottom row) of the distributions of each evolved parameter, scaled between 0 and 1, for all of the Pareto-optimal individuals. For each box-plot, the central red line indicates the median, the light blue box outlines the 25th through 75th percentiles, the dashed light blue vertical lines denote the box whiskers (non-outliers), and the dark blue circles denote outliers. In each histogram, the horizontal axes are scaled parameter values and the height of each bin indicates the density around a given value.

multiplicity correction resulting in a P value of 0.005) shows that the variances in the control parameters (frequency and PWR) are significantly higher than those of the morphological parameters (length, height, and flexibility). Referring again to figures 9(b)–(d), we conclude that three distinct control strategies have been evolved, two of which are practical. Specifically, the energy-based control strategy, which has a

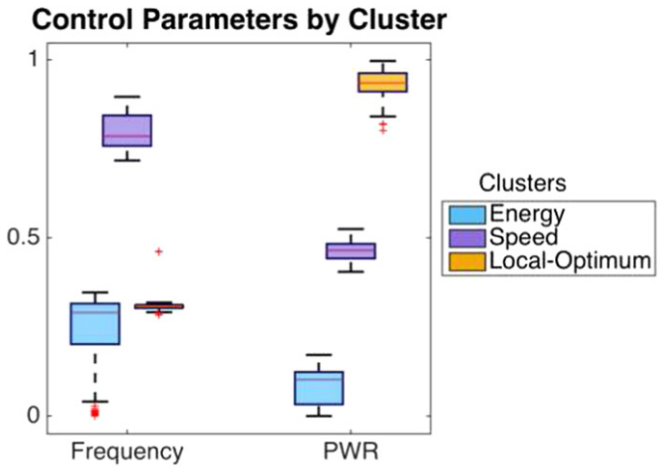

Figure 11. A comparison of control parameters among the three Pareto-optimal clusters. Frequency (right) and the PWR (left) are both increased to achieve faster swimming speeds and decreased to attain lower power consumption. For each box-plot, a central red line indicates the median, a box outlines the 25th through 75th percentiles, a vertical, dashed line denote the box whiskers (non-outliers), and red markers denote outliers.

Table 2. Parameter sets selected for further investigation.

Description	Frequency	PWR	Length	Height	Spring constant
Best speed	3.8 Hz	0.50	3.8 cm	1.0 cm	50 uNm rad^{-1}
Best energy	1.1 Hz	0.12	3.0 cm	1.0 cm	94 uNm rad^{-1}
Dominated	3.0 Hz	0.64	3.9 cm	1.3 cm	48 uNm rad^{-1}
Hand chosen	1.0 Hz	1.00	3.3 cm	1.1 cm	100 uNm rad^{-1}

maximum speed of roughly 5 cm s^{-1}, and the speed-based strategy, which has a minimum average power exerted of approximately 100 mW. This result suggests that, at least for the robot used in this study, morphology can be fixed and the trade-offs between the two objectives can be adjusted online by choosing a different set of parameters from the Pareto front.

Figure 11 compares parameters among the three different control strategies. The PWR of evolved control patterns is lower for lower power-consuming individuals (median value of 0.2), and higher for faster individuals (median value of 0.5). This result is expected, as the only way to minimize energy usage is to reduce the amount of *active* time for the actuator. Active time refers to the duration of time that a voltage is applied to the electromagnetic coil of the robot's actuator. Additionally, faster swimming individuals exhibit higher frequencies (median value of 4.0) than low power-consuming individuals (median value of 1.5).

The frequencies for lower power-consuming individuals suggest that they are sacrificing speed by evolving lower frequencies (i.e., since higher frequencies generally lead to higher speeds but increasing frequency does not affect power consumption). However, as will be discussed later, increasing frequency of these individuals actually reduces speed. As for the local optimum cluster, it appears that NSGA-II located a region in the search space where speed was maximized by increasing the PWR and sacrificing power consumption. Likely due to its exploitative nature, the algorithm was unable to escape this region of the search space in some replicates. That is, NSGA-II expands its current Pareto front by searching the neighborhood around current Pareto-optimal individuals. Therefore, it seems that it is difficult to find an evolutionary trajectory from this local optimum to the actual optimal Pareto-front.

Additionally, Pareto-optimal solutions tend to have either high frequency and high PWR or relatively low values for both parameters. This trend occurs because a high frequency with a low PWR will result in very little fin motion, due to the duration of time that the actuator would remain active. For example, with a frequency of 5 Hz (the maximum allowed value) and a PWR of 0.4, the actuator will remain active for only 40 ms at a time, which is not enough time to rotate the caudal fin for generating thrust. For comparison, the median value for activation duration for all Pareto-optimal solutions (excluding the local optima) is 64 ms with a standard deviation of 10 ms, and the minimum and maximum possible values are 10 and 1000 ms, respectively. Thus, although most solutions require the actuator to be active for relatively short durations, there appears to be a lower limit of approximately 50 ms.

To explore the generality of evolved solutions, we selected three evolved individuals and a solution with hand-chosen parameters to examine further. The values for these parameters are listed in table 2. These

parameter sets include morphologies near the optimal values listed previously, and exhibit a range of control parameter values. The *best speed* and *best energy* parameters refer to evolved individuals near the end points of the Pareto front (the highest speed and lowest power consumption), the *dominated* parameters refer to a randomly selected individual from an early generation that is not near the Pareto front, and *hand chosen* represents parameter values that we have chosen using expert knowledge.

For each fin design shown in table 2, we first conducted a parameter sweep over the control parameters. The purpose of these control parameter sweeps is to evaluate individuals under a range of operating conditions, because for many practical applications it will be useful to dynamically adjust the relative importance of the two objectives. In such a situation, the morphology is fixed and only the control patterns can be adjusted. For example, if a robotic fish is able to charge its batteries via a solar cell, it will be more important to conserve energy when operating in limited sunlight. However, under ideal conditions the robotic fish may be able to sacrifice power consumption for better performance (e.g., swimming speed). Furthermore, even under ideal conditions there may be valid reasons to swim at speeds lower than the maximum value (e.g., if the robotic fish is tracking another object). Thus, during these sweeps all parameters relating to fin morphology were fixed. Results from two experiments (*best speed* and *best energy*) are plotted in figure 12. Plots for *dominated* and *hand chosen* are not shown as they display characteristics similar to those demonstrated by *best energy*. We have not provided data for the parameter sweeps against average power, as average power is directly proportional to PWR.

Considering the parameter sweep plots, a desirable trait is the ability to adjust speed to specific values. The only set of parameters to demonstrate this trait is the *best speed* individual, as shown in figures 12(a), (b). This configuration allows the robotic fish to swim at speeds ranging from 1 to 9 cm s^{-1} by adjusting the frequency or PWR. Additionally, as depicted in figure 12(b), the robotic fish can swim at a range of speeds by adjusting the PWR value, which means that the robotic fish can effectively adjust its average power consumption from 20 to 220 mW and its average speed from 0.2 to 9 cm s^{-1}. Moreover, the apparent linear relationship between PWR and speed allows us to consider the trade-offs between power consumption and swimming speed without worrying about drastic performance changes that can occur under certain conditions. Specifically, we do not need to avoid certain values for PWR due to a nonlinear relationship over a smaller interval. However, any value over 120 mW of average delivered power (corresponding to a PWR of 0.5) appears to waste energy, as average power increases but speed does not. The remaining three sets of parameters (*best energy*, *dominated*, and *hand chosen*) do not exhibit the same ability to adjust speed. However, different speeds may be achieved by these individuals by adjusting both frequency and PWR simultaneously.

We also conducted similar sweeps over the morphological parameters: fin length, height, and flexibility. In these sweeps, the control parameters were fixed. Results are again plotted for the *best speed* and *best energy* parameter values; this data can be found in figure 13. For the best speed individual (figures 13(a)–(c)) it is apparent that the morphology is optimized for speed. Specifically, any change in morphology results in a speed reduction. However, speeds attained by parameter values that lead to lower energy consumption (figures 13(d)–(f)) are less affected by the morphological parameters. That is, changing the morphology results in smaller changes to speed.

4. Experimental validation of evolved designs

One goal of this study is to evolve solutions that can be applied to a physical robot. The increased fabrication complexity of flexible materials can be addressed in part with the aid of rapid prototyping equipment, such as 3D printers, which can decrease the time between design and physical testing [68]. Recent advances in 3D printing technology allow for multiple materials to be jetted simultaneously, which enables the printing of composite materials as employed in this study.

To confirm what we see in simulation carries into reality we selected three evolved designs and one hand-chosen parameter set to validate experimentally. In this section we first describe the method for fabricating caudal fins [59], and then we present results from physical experiments.

4.1. Fin fabrication and testing

In simulation, flexibility of the caudal fin is determined by spring coefficients. However, the flexibility for actual materials is expressed as a physical property such as the Young's modulus. Therefore, we require the following equation, which relates a spring coefficient to the Young's modulus of a 3D-printed component:

$$K_s = \frac{Edh^3}{12l}, \tag{6}$$

where K_s and E refer to the spring coefficient and Young's modulus values, respectively, and d, h, and l represent the height, thickness, and length of a rectangular fin, respectively.

To match the Young's modulus of printed materials with evolved spring coefficients, we needed a way to fabricate a fin for a given Young's modulus value. To do so, we designed composite fins in which flexibility is adjusted by varying the relative thickness of two different materials, as shown in figure 14. The fin

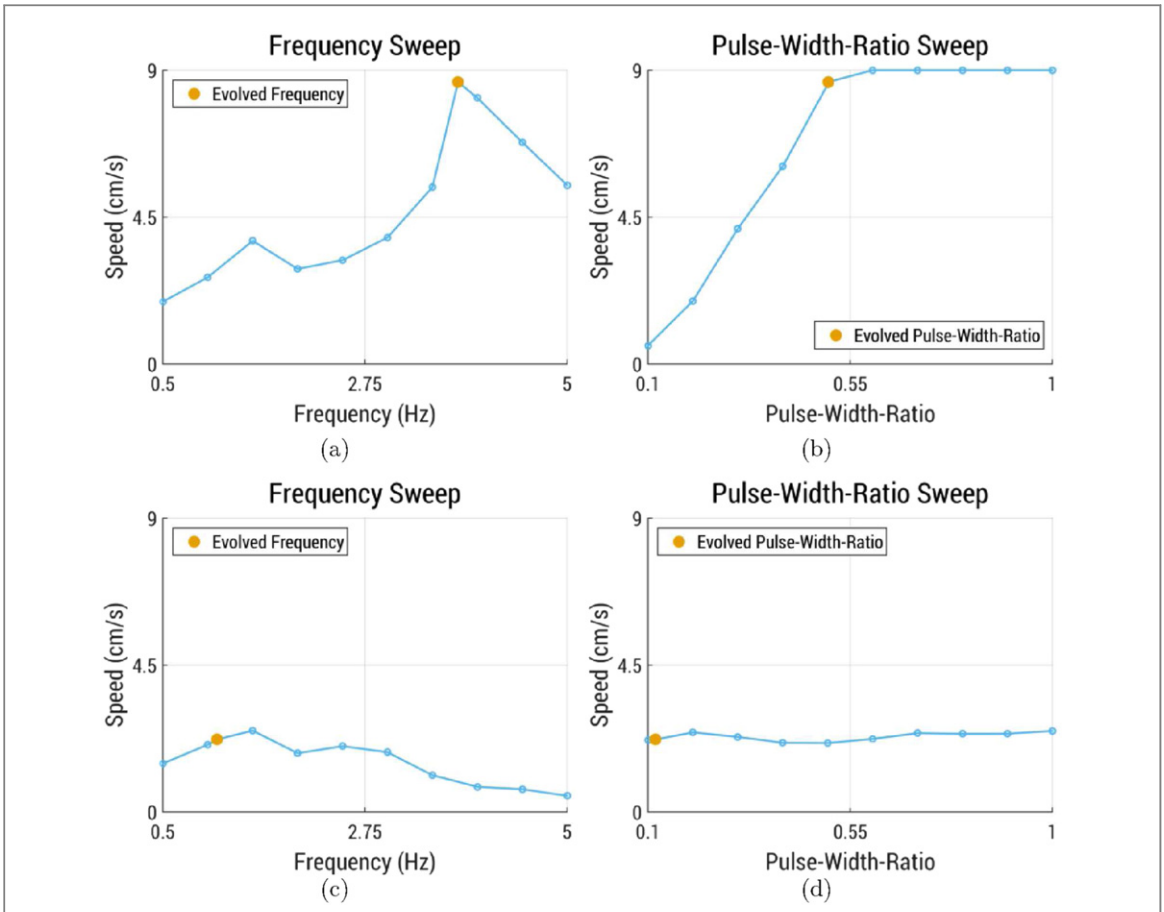

Figure 12. Control parameter sweeps for two sets of parameters: (a), (b) an evolved individual with high speed, and (c), (d) an evolved individual with low energy consumption.

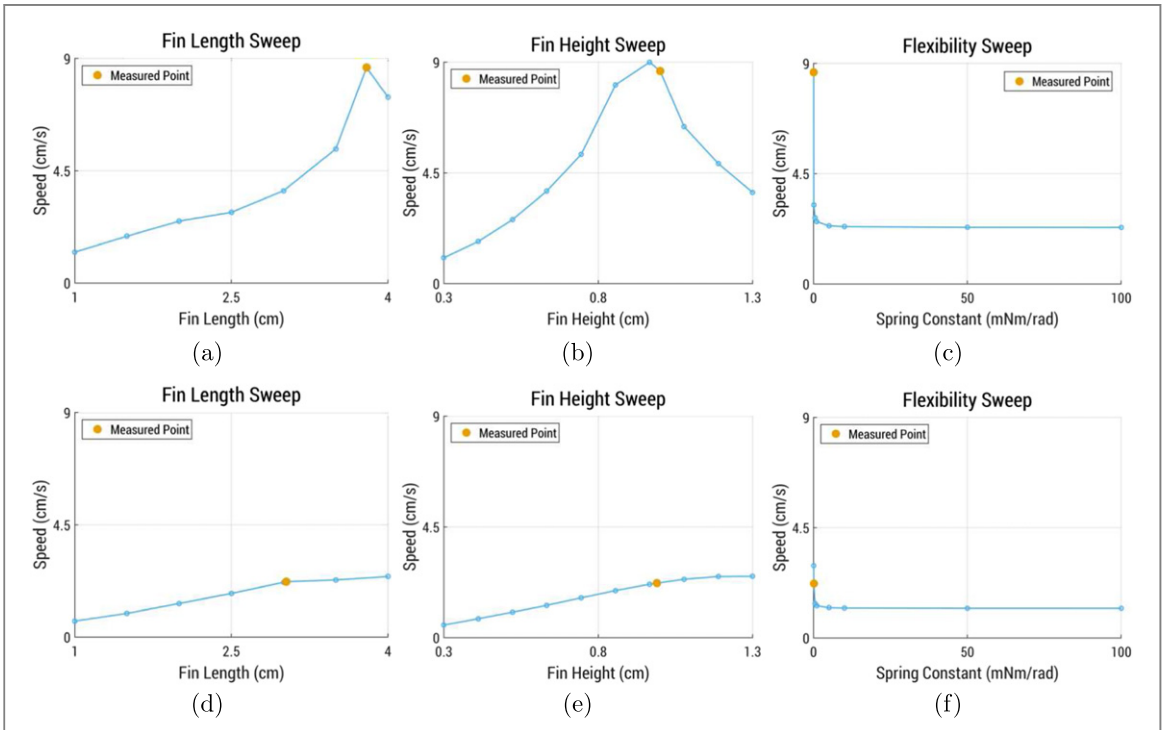

Figure 13. Morphological parameter sweeps for two sets of parameters: (a), (b) an evolved individual with high speed, and (c), (d) an evolved individual with low energy consumption.

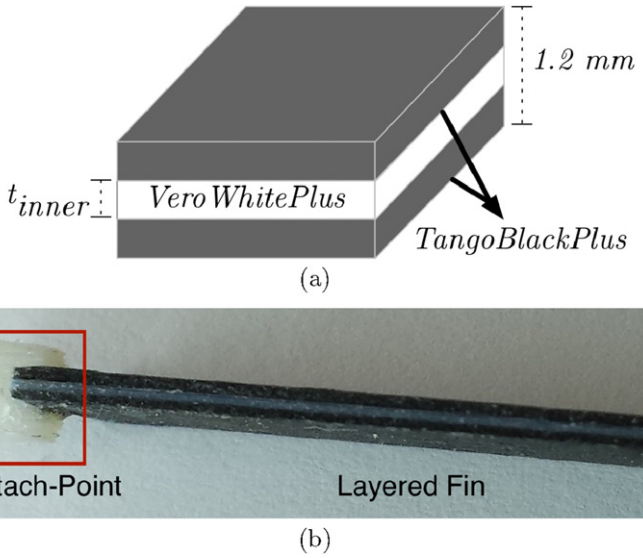

Figure 14. (a) Diagram of a composite material for a 3D-printed flexible caudal fin (note: the caudal fin would be on its side). (b) Top-view photograph of a 3D-printed caudal fin with an inner thickness of 0.38 mm. The overall thickness is a constant 1.2 mm. The effective Young's modulus value for the composite material depends on the relative thickness (t_{inner}) of the inner VeroWhitePlus layer with respect to the two flexible TangoBlackPlus outer layers.

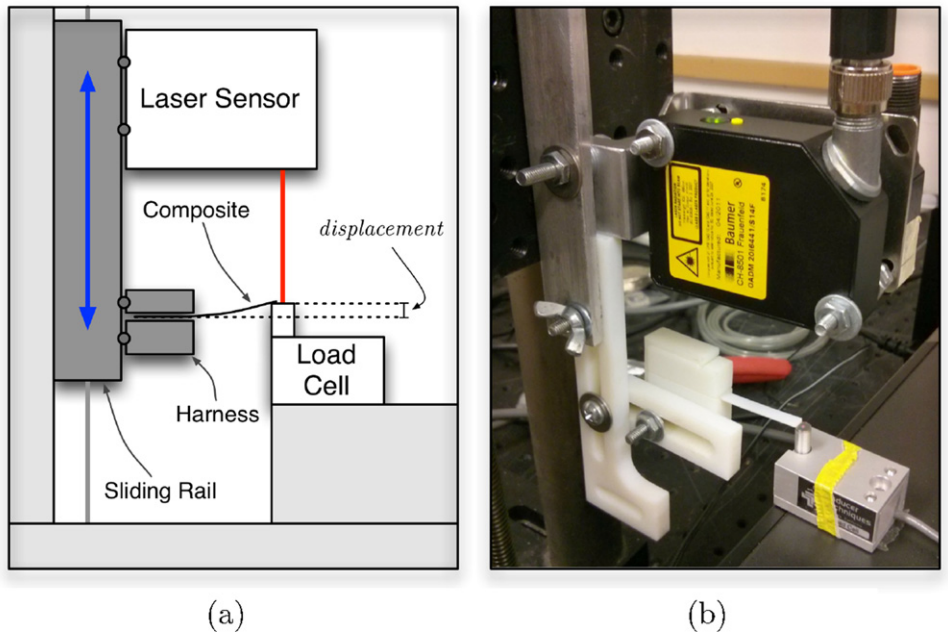

Figure 15. Testing of physical fins. (a) Diagram of the same experiment showing the testing process. (b) Photograph of the experimental setup for measuring the effective Young's modulus of 3D-printed composite materials.

comprises two outer layers of a rubber-like polymer, and an inner layer of a more rigid plastic. When discussing composite materials, flexibility is often referred to as an *effective* Young's modulus to distinguish from uniformly fabricated materials. Thus, specifying the thickness of the inner layer, t_{inner}, and fixing the overall thickness to 1.2 mm limits the range of possible effective Young's modulus values.

To determine the 3D-printable range of effective Young's modulus values, we set up the experiment shown in figure 15 and measured the Young's modulus values for a series of composite materials with different values of t_{inner}. For this experimental setup, the Young's modulus E is evaluated with:

$$E = \frac{L_b^3 P_{\text{L}}}{3 I_b w_{\text{L}}} \tag{7}$$

where L_b and I_b are the length and area of moment inertia of the test composite, respectively, and P_{L} and w_{L} are the load and displacement at the tip of the composite, respectively.

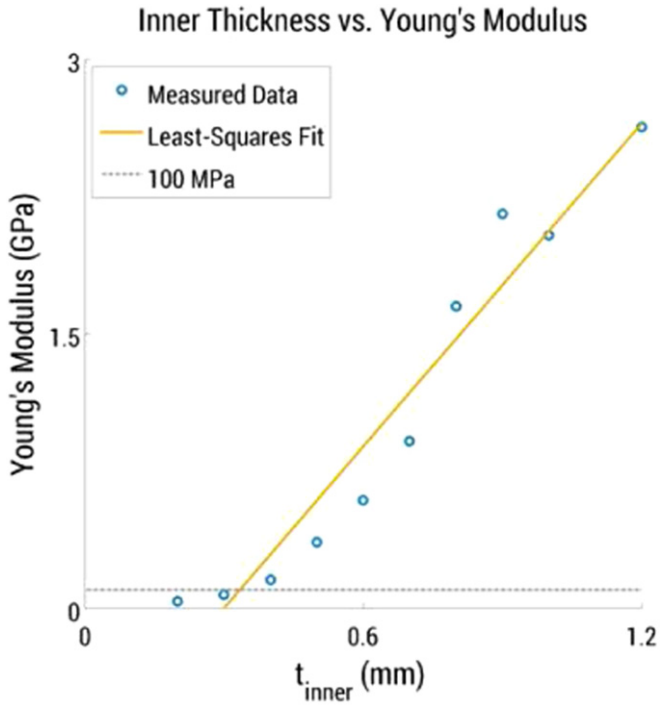

Figure 16. Effective Young's modulus of composite materials for different values of t_{inner}.

As shown in figure 15, a sample of composite material is fixed to a harness while its tip rests against a load cell. Displacement at the tip is adjusted using a sliding rail and measured with a laser sensor. Three replicate sets of load and displacement data are gathered for each composite material, in which each set comprises five data points at different displacements. A least square error method is adopted to find the slope between force and displacement for each of the three sets, and the effective Young's modulus for each composite is evaluated as the average of the three replicate experiments.

Figure 16 plots the results of these experiments for different values of t_{inner}. The evolvable range of spring coefficients corresponds to an effective Young's modulus of approximately 100 MPa to 3 GPa. These values correspond to a range of materials roughly from rubber, which typically has a Young's modulus of 10–100 MPa, to hard plastics, which have a Young's modulus of 1–5 GPa. Here, we use the *best fit* line in figure 16 to find the required t_{inner} value for a given Young's modulus.

4.2. Physical validation

The physical experiments use the same control and morphological parameters as their simulated counterparts (these values are listed in table 2). Making use of the composite fins described in the previous section enables 3D printing of caudal fins with a specified morphology (fin dimensions and flexibility). The fabricated fins are attached to the small robotic fish pictured in figure 1. The robotic fish is then placed in a large water tank, and speed is measured using a method similar to that described in section 2 regarding the evolutionary fitness evaluation. Specifically, the robotic fish speed is measured after it reaches a steady maximum speed. For physical experiments, each reported speed value is the result of averaging data from five trials. Results from these experiments can be found in the table 3.

Table 3. Comparison between speeds acquired in simulation and speeds attained by the small robotic fish in a water tank.

Description	Simulation speed	Measured speed	Absolute error
Best speed	8.6 cm s^{-1}	3.8 cm s^{-1}	4.8 cm s^{-1}
Best energy	2.2 cm s^{-1}	1.9 cm s^{-1}	0.3 cm s^{-1}
Dominated	3.2 cm s^{-1}	3.5 cm s^{-1}	0.2 cm s^{-1}
Hand chosen	2.7 cm s^{-1}	2.9 cm s^{-1}	0.2 cm s^{-1}

For three of the four validated parameter sets, the results show a close relationship between simulation and reality. And for all four sets of parameters, the achieved speeds demonstrate a consistent order between the simulation and the experimental measurement; in other words, ranking the designs by their achieved speeds will result in the same ordering in simulation as in experiments. These observations provide support for the effectiveness of the simulation model and the design approach. On the other hand, we note that the *best speed* evolved parameters result in a considerably faster speed in simulation than in experiments. Analyzing high-speed video of the robotic fish reveals that the physical caudal fin cannot reach as large an amplitude as in simulation. This behavior can likely be attributed to the high frequency

at which the device is operating. As listed in table 2, the frequency for this individual is 3.8 Hz, which is near the upper limit we found to be physically possible during our initial prototyping. Modeling of the electromagnetic actuator will need to be refined to ensure that the simulation environment cannot be exploited by the evolutionary algorithm, as occurs for the *best speed* experiment.

Even with a higher accuracy simulation the reality gap can still be an issue. For this reason, many research groups are investigating explicit methods for countering the reality gap. Two of the most prominent are the 'intelligent trial-and-error' algorithm [69] and self-modeling [70]. As an alternative to these methods, we are currently investigating adaptive control as a method for mitigating effects of the reality gap [71]. In essence, the controller would *adapt* to the differences between simulation and reality.

5. Conclusion

In this study, we proposed an EMO approach to the design of morphology and control for a robotic fish with a flexible caudal. As a way of illustration, we chose two common objectives (maximizing swimming speed and minimizing energy consumption), although our approach would work with any combination of objectives (for example, agility, speed, and energy efficiency). The robotic fish used during validation experiments utilizes an electromagnetic actuator, which enables a direct calculation of energy consumption. Simulation-based optimization allowed us to explore trade-offs between the two competing objectives. Certain evolved and hand-chosen sets of parameters were then selected for physical validation. With the aid of a 3D printer, simulated caudal fins were fabricated to match specifications, and simulation results were validated in a large water tank. Aside from individuals exhibiting the fastest speeds, physical trials support the effectiveness of the proposed evolutionary design approach.

Surprisingly, control parameters (frequency of oscillation and PWR) were found to account for most of the variation among Pareto-optimal individuals. This discovery is intriguing, as it indicates that at least for the robotic fish and objectives employed in this study, most of the engineering effort can be invested in controller design, while less time needs to be spent on morphology. Specifically, morphology can be fixed and control strategies can be designed such that most of the Pareto-optimal solution space can be reached. As a result, the robotic fish can move from high speed, high energy states to low speed, lower energy states that are near the final Pareto-front. Fixing the morphological characteristics and performing parameters sweeps on the control parameters enabled the discovery of individuals that exhibit such generality.

The purpose of this study was to investigate how compliant caudal fins should be matched with control parameters while considering both performance and energy consumption. However, the developed techniques can be applied to the design of other physical systems incorporating soft/flexible components. In our future investigations we intend to focus on four areas. First, we will explore whether different objectives or robotic fish configurations change the variance of parameters in the final population. For instance, does optimizing for agility and efficiency lead to a more varied set of morphological parameters in the set of Pareto-optimal solutions? Second, we will remove the limitations on caudal fin morphology (rectangular shape) by extending our dynamic models to work with fins of other shapes and nonuniform flexibilities. Third, our fitness functions will incorporate more complex tasks (e.g., station keeping) and control paradigms (e.g., feedback controllers), which will encourage the evolution of general robotic fish behaviors. Fourth, we will develop algorithms for adjusting the relative importance of multiple objectives online. For example, under certain conditions it may be more useful to focus on speed, and at other times it will be important to be more energy-efficient. More broadly, the techniques presented in this study are intended to be an initial step toward our goal of producing optimization methods that can be used for any robots incorporating soft/flexible components. Ultimately, we intend to combine these techniques with advanced control algorithms, including self-modeling [69, 72] and adaptive control [71], to design systems that are robust to their uncertain and changing environments.

Acknowledgments

The authors gratefully acknowledge the contributions and feedback on the work provided by Jared Moore, Jianxun Wang, and the BEACON Center at Michigan State University. This work was supported in part by National Science Foundation grants IIS-1319602, CCF-1331852, CNS-1059373, CNS-0915855, and DBI-0939454, and by a grant from the Michigan State University Discretionary Funding Initiative.

References

[1] Tan X 2011 Autonomous robotic fish as mobile sensor platforms: challenges and potential solutions *Mar. Technol. Soc. J.* **45** 31–40

[2] Yu J, Liu L, Wang L, Tan M and Xu D 2008 Turning control of a multilink biomimetic robotic fish *IEEE Trans. Robot.* **24** 201–6

[3] Hover F S, Eustice R M, Kim A, Englot B, Johannsson H, Kaess M and Leonard J J 2012 Advanced perception, navigation and planning for autonomous in-water ship hull inspection *Int. J. Robot. Res.* **31** 1445–64

[4] Wang Y, Tan R, Xing Gu, Wang J and Tan X 2012 Accuracy-aware aquatic diffusion process profiling using robotic sensor networks *Proc. 11th ACM/IEEE Conf. on Information Processing in Sensor Networks* (*Beijing, China*) pp 281–92

[5] Takada Y, Koyama K and Usami T 2014 Position estimation of small robotic fish based on camera information and gyro sensors *Robotics* **3** 149–62

[6] Faria J J, Dyer J R G, Clément R O, Couzin I D, Holt N, Ward A J W, Waters D and Krause J 2010 A novel method for investigating the collective behaviour of fish: introducing robofish *Behav. Ecol. Sociobiology* **64** 1211–8

[7] Long J H Jr, Koob T J, Irving K, Combie K, Engel V, Livingston H, Lammert A and Schumacher J 2006 Biomimetic evolutionary analysis: testing the adaptive value of vertebrate tail stiffness in autonomous swimming robots *J. Exp. Biol.* **209** 4732–46

[8] Phamduy P, Polverino G, Fuller R C and Porfiri M 2014 Fish and robot dancing together: bluefin killifish females respond differently to the courtship of a robot with varying color morphs *Bioinspiration Biomimetics* **9** 036021

[9] Triantafyllou M S and Triantafyllou G S 1995 An efficient swimming machine *Sci. Am.* **272** 64

[10] Anderson J M and Chhabra N K 2002 Maneuvering and stability performance of a robotic tuna *Integr. Comp. Biol.* **42** 118–26

[11] Kato N 2000 Control performance in the horizontal plane of a fish robot with mechanical pectoral fins *IEEE J. Ocean. Eng.* **25** 121–9

[12] Low K H and Chong C W 2010 Parametric study of the swimming performance of a fish robot propelled by a flexible caudal fin *Bioinspiration Biomimetics* **5** 046002

[13] Yu J, Tan M, Wang S and Chen E 2004 Development of a biomimetic robotic fish and its control algorithm *IEEE Trans. Syst. Man Cybern.* **34** 1798–810

[14] Epstein M, Colgate J E and MacIver M A 2006 Generating thrust with a biologically-inspired robotic ribbon fin *Proc. 2006 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems* (*Beijing, China*) pp 2412–7

[15] Lauder G V, Anderson E J, Tangorra J and Madden P G A 2007 Fish biorobotics: kinematics and hydrodynamics of self-propulsion *J. Exp. Biol.* **210** 2767–80

[16] Yu J, Ding R, Yang Q, Tan M, Wang W and Zhang J 2011 On a bio-inspired amphibious robot capable of multimodal motion *IEEE/ASME Trans. Mechatronics* **99** 1–10

[17] Tangorra J L, Lauder G V, Hunter I W, Mittal R, Madden P G A and Bozkurttas M 2010 The effect of fin ray flexural rigidity on the propulsive forces generated by a biorobotic fish pectoral fin *J. Exp. Biol.* **213** 4043–54

[18] Kanso E and Newton P K 2009 Passive locomotion via normal-mode coupling in a submerged spring-mass system *J. Fluid Mech.* **641** 205–15

[19] Hiller J D and Lipson H 2012 Automatic design and manufacture of soft robots *IEEE Trans. Robot.* **28** 457–66

[20] McHenry M J, Pell C A and Long J H 1995 Mechanical control of swimming speed: stiffness and axial wave form in undulating fish models *J. Exp. Biol.* **198** 2293–305

[21] Feilich K L and Lauder G V 2015 Passive mechanical models of fish caudal fins: effects of shape and stiffness on self-propulsion *Bioinspiration Biomimetics* **10** 036002

[22] Park Y-J, Jeong U, Lee J, Kwon S-R, Kim H-Y and Cho K-J 2012 Kinematic condition for maximizing the thrust of a robotic fish using a compliant caudal fin *IEEE Trans. Robot.* **28** 1216–27

[23] Alvarado P V and Youcef-Toumi K 2006 Design of machines with compliant bodies for biomimetic locomotion in liquid environments *J. Dyn. Sys. Meas. Control* **128** 3–13

[24] Esposito C J, Tangorra J L, Flammang B E and Lauder G V 2012 A robotic fish caudal fin: effects of stiffness and motor program on locomotor performance *J. Exp. Biol.* **215** 56–67

[25] Daou H E, Salumäe T, Chambers L D, Megill W M and Kruusmaa M 2014 Modelling of a biologically inspired robotic fish driven by compliant parts *Bioinspiration Biomimetics* **9** 016010

[26] Gomez F J and Miikkulainen R 2003 Active guidance for a finless rocket using neuroevolution *Proc. 5th Annual Conf. on Genetic and Evolutionary Computation* (*GECCO*) (Berlin: Springer) pp 2084–95

[27] Brooks R A 1992 Artificial life and real robots *Proc. First European Conference on Artificial Life* (Cambridge, MA: MIT) pp 3–10

[28] Sims K 1994 Evolving 3D morphology and behavior by competition *Artif. Life* **1** 353–72

[29] Nolfi S and Floreano D 2000 *Evolutionary robotics: The Biology, Intelligence, and Technology of Self-organizing Machines* (Cambridge, MA: MIT)

[30] Lipson H 2005 Evolutionary robotics and open-ended design automation *Biomimetics, Biologically Inspired Technologies* (Boca Raton, FL: CRC) Ch 4 pp 129–55

[31] Floreano D, Husbands P and Nolfi S 2008 Evolutionary robotics *Springer Handbook of Robotics* (Berlin: Springer) pp 1423–51

[32] Bongard J C 2013 Evolutionary robotics *Commun. ACM* **56** 74–83

[33] Chiel H J and Beer R D 1997 The brain has a body: adaptive behavior emerges from interactions of nervous system, body and environment *Trends Neurosci.* **20** 553 – 557

[34] Mautner C and Belew R 2000 Evolving robot morphology and control *Artif. Life Robot.* **4** 130–6

[35] von Haller B, Ijspeert A and Floreano D 2005 Co-evolution of structures and controllers for neubot underwater modular robots *Advances in Artificial Life* (*Lecture Notes in Computer Science* vol 3630) (Berlin: Springer) pp 189–99

[36] Pfeifer R, Lungarella M and Iida F 2007 Self-organization, embodiment, and biologically inspired robotics *Science* **318** 1088–93

[37] Yao X 1999 Evolving artificial neural networks *Proc. IEEE* **87** 1423–47

[38] Clune J, Beckmann B E, Ofria C and Pennock R T 2009 Evolving coordinated quadruped gaits with the HyperNEAT generative encoding *Proc. IEEE Congress on Evolutionary Computation* (*Trondheim, Norway*) pp 2764–71

[39] Reil T and Husbands P 2002 Evolution of central pattern generators for bipedal walking in a real-time physics environment *IEEE Trans. Evolutionary Comput.* **6** 159–68

[40] Baydin A G 2012 Evolution of central pattern generators for the control of a five-link bipedal walking mechanism *Paladyn, J. Behav. Robot.* **3** 45–53

[41] Koza J R and Rice J P 1992 Automatic programming of robots using genetic programming *Proc. of the Tenth National Conf. on Artificial Intelligence* pp 194–201

[42] Kala R 2012 Multi-robot path planning using co-evolutionary genetic programming *Expert Syst. Appl.* **39** 3817–31

[43] Pfeifer R, Lungarella M and Iida F 2012 The challenges ahead for bio-inspired 'soft' robotics *Commun. ACM* **55** 76–87

[44] Majidi C July 2013 Soft robotics: a perspective—current trends and prospects for the future *Soft Robot.* **1** 5–11

[45] Kim S, Laschi C and Trimmer B 2013 Soft robotics: a bioinspired evolution in robotics *Trends Biotechnol.* **31** 287–94

[46] Lipson H 2014 Challenges and opportunities for design, simulation, and fabrication of soft robots *Soft Robot.* **1** 21–7

[47] Rieffel J, Knox D, Smith S and Trimmer B 2014 Growing and evolving soft robots *Artif. life* **20** 143–62

[48] Cheney N, MacCurdy R, Clune J and Lipson H 2013 Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding *Proc. 15th Annual Conf. on Genetic and Evolutionary Computation* (New York: ACM) pp 167–74

[49] Ehrgott M 2006 A discussion of scalarization techniques for multiple objective integer programming *Ann. Oper. Res.* **147** 343–60

[50] Deb K, Pratap A, Agarwal S and Meyarivan T 2002 A fast and elitist multiobjective genetic algorithm: NSGA-II *IEEE Trans. Evolutionary Comput.* **6** 182–97

[51] Zitzler E, Laumanns M and Thiele L 2001 SPEA2: improving the strength Pareto evolutionary algorithm *Technical Report* 103 Computer Engineering and Networks Laboratory (TIK), ETH Zurich, Switzerland

[52] Zitzler E and Künzli S 2004 Indicator-based selection in multiobjective search *Parallel Problem Solving from Nature-PPSN VIII* (Berlin: Springer) pp 832–42

[53] Jiang S, Zhang J, Ong Y-S, Zhang A N and Tan P S 2014 A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm *IEEE Trans. Cybern.* **45** 2202–13
[54] Auger A, Bader J, Brockhoff D and Zitzler E 2012 Hypervolume-based multiobjective optimization: theoretical foundations and practical implications *Theor. Comput. Sci.* **425** 75–103
[55] Coello C A C, Veldhuizen D A V and Lamont G B 2007 *Evolutionary Algorithms for Solving Multi-Objective Problems* vol 242 (Berlin: Springer)
[56] Konak A, Coit D W and Smith A E 2006 Multi-objective optimization using genetic algorithms: a tutorial *Reliab. Eng. Syst. Saf.* **91** 992–1007
[57] Zhou A, Qu B-Y, Li H, Zhao S-Z, Suganthan P N and Zhang Q 2011 Multiobjective evolutionary algorithms: a survey of the state of the art *Swarm Evolutionary Comput.* **1** 32–49
[58] Clark A J, Moore J M, Wang J, Tan X and McKinley P K 2012 Evolutionary design and experimental validation of a flexible caudal fin for robotic fish *Proc. 13th Int. Conf. on the Synthesis and Simulation of Living Systems* (*East Lansing, Michigan, USA*) pp 325–32
[59] Clark A J, Wang J, Tan X and McKinley P K 2014 Balancing performance and efficiency in a robotic fish with evolutionary multiobjective optimization *Proc. 2014 IEEE Int. Conf. on Evolvable Systems* (*ICES*) pp 227–34
[60] Alessi A, Sudano A, Accoto D and Guglielmelli E 2012 Development of an autonomous robotic fish *Proc. 2012 4th IEEE RAS & EMBS Int. Conf. on Biomedical Robotics and Biomechatronics* (*BioRob*) (Piscataway, NJ: IEEE) pp 1032–7
[61] Okada T, Guo S, Xiao N, Qiang F and Yamauchi Y 2012 Control of the wireless microrobot with multi-DOFs locomotion for medical applications *Proc. 2012 Int. Conf. on Mechatronics and Automation* (*ICMA*) (Piscataway, NJ: IEEE) pp 405–10
[62] Takada Y, Araki R, Nakanishi Y, Nonogaki M, Ebita K and Wakisaka T 2010 Development of small fish robots powered by small and ultra-light passive-type polymer electrolyte fuel cells *J. Robot. Mechatronics* **22** 150–7
[63] Shin B H, Lee K-M and Lee S-Y 2015 A miniaturized tadpole robot using an electromagnetic oscillatory actuator *J. Bionic Eng.* **12** 29–36
[64] Jakobi N, Husbands P and Harvey I 1995 Noise and the reality gap: the use of simulation in evolutionary robotics *Advances in Artificial Life* (*Lecture Notes in Computer Science* vol 929) (Berlin: Springer) pp 704–20
[65] Koos S, Mouret J-B and Doncieux S 2010 Crossing the reality gap in evolutionary robotics by promoting transferable controllers *Proc. 12th Annual Conf. on Genetic and Evolutionary Computation* (*GECCO*) (New York: ACM) pp 119–26
[66] Wang J, McKinley P K and Tan X 2012 Dynamic modeling of robotic fish with a flexible caudal fin *Proc. ASME 2012 5th Annual Dynamic Systems and Control Conf. joint with the JSME 2012 11th Motion and Vibration Conf.* (*Ft. Lauderdale Florida, USA*) pp 203–12
[67] Lighthill M J 1971 Large-amplitude elongated-body theory of fish locomotion *Proc. R. Soc. London* **179** 125–38
[68] Richter C and Lipson H 2011 Untethered hovering flapping flight of a 3D-printed mechanical insect *Artif. Life* **17** 73–86
[69] Cully A, Clune J, Tarapore D and Mouret J-B 2015 Robots that can adapt like animals *Nature* **521** 503–7
[70] Bongard J and Lipson H 2014 Evolved machines shed light on robustness and resilience *Proc. IEEE* **102** 899–914
[71] Clark A J, McKinley P K and Tan X 2015 Enhancing a model-free adaptive controller through evolutionary computation *Proc. 2015 Conf. on Genetic and Evolutionary Computation, GECCO '15* (New York: ACM) pp 137–44
[72] Bongard J, Zykov V and Lipson H 2006 Resilient machines through continuous self-modeling *Science* **314** 1118–21