Evolution of Fin Undulation on a Physical Knifefish-inspired Soft Robot

Frank Veenstra*
IT University of Copenhagen
Copenhagen, Denmark
frve@itu.dk

Jonas Jørgensen*
IT University of Copenhagen
Copenhagen, Denmark
jjoe@itu.dk

Sebastian Risi
IT University of Copenhagen
Copenhagen, Denmark
sebr@itu.dk

ABSTRACT

Soft robotics is a growing field of research and one of its challenges is how to efficiently design a controller for a soft morphology. This paper presents a marine soft robot inspired by the ghost knifefish that swims on the water surface by using an undulating fin underneath its body. We investigate how propagating wave functions can be evolved and how these affect the swimming performance of the robot. The fin and body of the robot are constructed from silicone and six wooden fin rays actuated by servo motors. In order to bypass the reality gap, which would necessitate a complex simulation of the fish, we implemented a Covariance Matrix Adaptation Evolution Strategy (CMA-ES) directly on the physical robot to optimize its controller for travel speed. Our results show that evolving a simple sine wave or a Fourier series can generate controllers that outperform a hand programmed controller. The results additionally demonstrate that the best evolved controllers share similarities with the undulation patterns of actual knifefish. Based on these results we suggest that evolution on physical robots is promising for future application in optimizing behaviors of soft robots.

KEYWORDS

Soft Robotics, Evolutionary Algorithms, Covariance Matrix Adaptation Evolutionary Strategy, Evolution of Physical Systems

1 INTRODUCTION

Despite recent advances in evolutionary robotics, the reality gap [15] is still a prevalent issue. Especially in the emerging field of soft robotics it becomes more difficult to simulate the physical properties of soft materials accurately [25]. In cases where this was accomplished successfully, it required high computational power and complex algorithms [6]. For aquatic robots, the integration of flexible materials can lead to increased performance by the principle of morphological computation, i.e. by exploiting that dynamic interactions with the environment can be useful for achieving a desired behavior efficiently. The complex mechanics of silicone and its hydrodynamic interactions are, however, computationally heavy to simulate, especially when the morphology is driven by multiple actuators. For these reasons, we propose an evolutionary approach of directly evolving physical systems [24] as a feasible alternative method to evolve efficient behavior of a bio-inspired soft robot. To the best of our knowledge, this is the first paper to use an evolutionary algorithm to evolve a behavior directly on a physical soft robot without prior simulation.

Soft robots have been proposed for a number of applications that include exploration and search and rescue operations. For such tasks high maneuverability is usually necessary. Since the family of ghost knifefish (Apteronotidae) contain examples of dexterous aquatic animals capable of high multidirectional maneuverability at low speeds [20], we chose this fish as our model whose control will be subjected to evolution. Knifefish are able to produce thrust in many directions by undulating a single anal fin located underneath the body. By generating propagating waves across their fin they can easily move backwards and forwards depending on the directionality of the wave [9]. Vertical thrust is accomplished through sending counter-propagating waves towards and away from the center of the fin canceling out longitudinal forces. In undulatory swimming the thrust is produced through a reaction force on the fluid adjacent to the body or fin surface. Bending of the body part, in our case the fin, enables wave propagation. The combination of the lateral forces produced on both sides of the fin should cancel out each other to produce a net forward thrust [2].

1.1 Evolution of Soft Robots

The evolutionary robotics approach to soft robotics has thus far only been implemented in simulation environments such as VoxCAD [3, 4, 8, 16] or off-the-shelf physics engines where morphologies are represented by tetrahedral meshes and the controls and morphology have been evolved [23]. Computational power is, however, a major constraint when using simulations. Computational requirements scale proportionally to the amount of tetrahedra and voxels simulated, usually exponentially, increasing the computational power needed when more are used. Morphologies found through the VoxCAD approach have only been replicated physically by means of soft volumetrically expanding materials that require changes in the pressure of the surroundings for actuation [14].

Controllers for simulations of existing partially soft morphologies have also been evolved in simulation environments and in some cases transferred to hardware. A genetic algorithm with a "lumped" dynamic model simulation has been used to evolve the gait of a soft caterpillar-inspired robot and has resulted in an increase in performance of a physical prototype [26]. In another instance, both an objective-based and a novelty-driven (*novelty search* [17]) approach have been utilized to optimize the design of a crawling

*These authors contributed equally to this work.

octopus by discovering self-stabilizing dynamic gaits [7]. A differential evolution algorithm was used to optimize a model-free adaptive controller (MFAC) in a simulation of a robotic fish with a flexible caudal fin [5]. For the same morphology an evolutionary multiobjective optimization technique (NSGA-II algorithm) found morphological and control parameters in simulation that maximize the swimming speed and minimize the power usage with subsequent validation in hardware [5]. However, in this approach it was found that the "best speed" parameters were considerably faster in simulation than in the experiments due to hardware limitations. This illustrates that although reasonable performance can be transfered from the simulation to reality, discrepancies are still persistent. In the above examples the evolution of soft robot morphologies and controllers was made possible by confining the search space to highly abstracted morphologies (fish where only a simple tail is flexible, caterpillar-like shapes) or by decomposing the morphologies into a finite number of voxels. While such approaches have yielded interesting results, they are still lacking in relation to realizing the full potential of soft robotics technology as they limit the design space to very simple or highly abstracted shapes. By evolving the controller in the physical hardware instead, one is able to reap the benefits of having both a bio-inspired design that mimics a natural model closely and an automated discovery of its most optimal behavior.

1.2 Knifefish-inspired Swimming Robots

Due to their unique morphology, knifefish have served as inspirations for a number of research robots. Building on the work of Low et al. [18, 19], Siahmansouri et al. constructed an untethered robot with 6 fin rays capable of regulating the direction and depth of swimming by moving the fin relative to a buoyancy tank [28]. Curet et al. built a knifefish-inspired robot with 32 individually actuated fin rays and were able to show that its optimal actuation parameters were similar to the ones of the black ghost knifefish [10]. They were also able to generate upward forces on the robot with counter-propagating undulation waves [9]. Sfakiotakis et al. devised a linear slide equipped with a fin composed of 8 individual fin rays and implemented open-loop velocity control and closed-loop position control [27].

A common denominator of the previous work on knifefish-inspired robots is the use of sinusoidal functions as an undulation pattern for the fin. This occurs despite the fact that a sine function is only an approximation of the actual undulation pattern of the species, which could be reproduced more accurately[30]. The design of our robot also departs from the earlier work as it is an integrated silicone morphology constructed with contemporary soft robotics fabrication techniques. This approach simplifies the fabrication of the fin and fin rays significantly. Moreover, elasticity is added to the fin, which has been hypothesized to be a means of increasing energy efficiency [18].

2 METHODOLOGY

We designed a soft swimming robot with a single undulating fin inspired by the anatomy of the black ghost knifefish[1]. To be able to evaluate its swimming speed with different motion patterns, we constructed the experimental setup shown in Figure 1. As we only evolve the forward swimming speed, the robot is fixed on a linear slide. It is not submersible and kept at a level of neutral buoyancy. The robot (E) is placed in the water surface of a 100×40×40cm aquarium. It is tethered with power and signal cables for its 6 servo motors. It is attached to a cart (F) with four ball-bearing wheels that is mounted on a T-slot beam linear slide (C) atop the aquarium. A plastic attachment piece (D) connects the cart to the linear slide and prevents the robot from turning. The slide is equipped with two IR sensors to measure when the beginning and end of the slide has been reached. For the evaluation of an undulation pattern, the robot starts on the left side of the track at the first IR sensor. During evaluation a swimming pattern is played on the robot and an ultrasonic distance sensor (A) measures the distance to a plastic plate (B) on the cart. The cumulative sum of the distance readings are used directly as the fitness value for the undulation pattern that was evaluated.

2.1 Mechanical Design of the Robot

The main parts of the robot are its hull, frame, and fin rays (see Figure 2) [2]. The hull and fin of the robot were constructed from Ecoflex 00-30 silicone (Young's modulus approx. 0.1 MPa, Shore A hardness 00-30) [22]. The uncured material was degassed after mixing and poured into a three part 3D printed mold (two sides and one inner part). The inner mold part holds the fin rays in place during casting and blocks out a compartment for the rigid inner frame, which was mounted after casting. The inner frame is constructed from laser cut acrylic parts that were glued together. The servo motors are held in place with bolts and nuts.

Six bamboo sticks (approx. diam. 3mm) serve as fin rays. With 6 fin rays it is theoretically possible for the robot to hover and to move forward, backward, up, and down by generating traveling and counter-propagating waves [9]. Each fin ray is attached to a servo motor via a servo bracket. The servo motors used were initially six H-KING HK 15148 mini servo motors. Due to malfunctions three of them were replaced with two TowerPro SG90 and one EMAX ES08AII. The servo motors are connected to the fin rays with a crank-like mechanism (Figure 3). The angle of a fin ray ϕ as a function of the servo angle α is given by:

$$\phi(\alpha) = \tan^{-1}\left(\frac{\sin(\alpha)\ \cdot 21}{30 - \cos(\alpha)\ \cdot\ 21}\right) \tag{1}$$

where the constant 21 is the distance (in mm) from the center of rotation of the servo to the piston that connects to the fin ray and the constant 30 the distance from the center of rotation of the servo to the approximate center of rotation of the fin ray (see Figure 3).

This equation, however, does not take into account the additional angular deflection caused by slack between the pistons and the fin ray, and the elasticity of the soft body resisting rotation (see Figure 4). The maximum angular excursion was therefore close to 28 degrees instead of the approximately 45 degrees that were calculated when not taking into consideration these issues.

[1]A video of the robot and our setup can be found at https://youtu.be/3XjgZbs0t2g

[2]The CAD files for the design can be accessed at https://cad.onshape.com/documents/51d2c0394f6e3aa7b3fc06b3

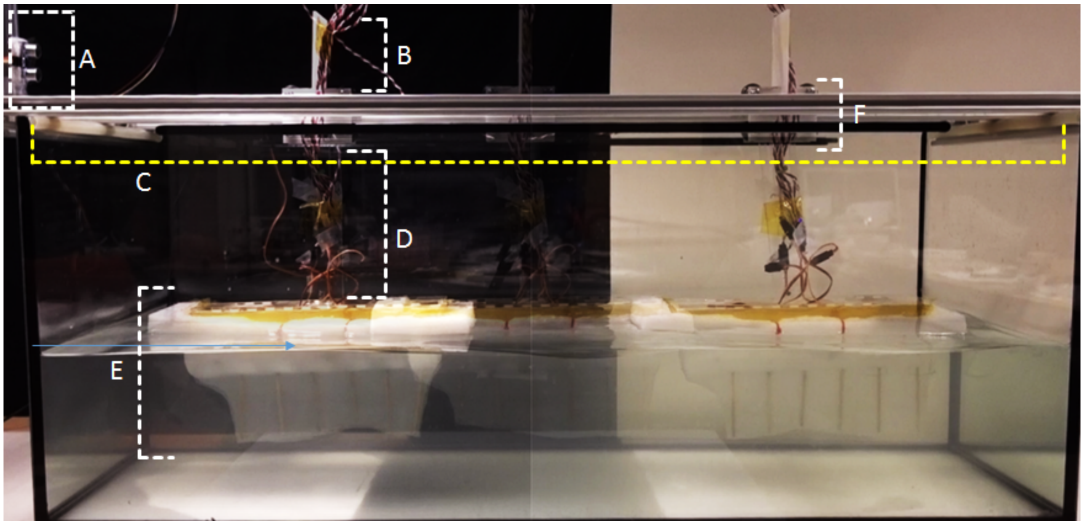


Figure 1: Experimental setup. (A) Ultrasonic distance sensor, (B) plastic plate for bouncing back the sound of the ultrasonic sensor (C) T-Slot linear slide, (D) plastic plate connecting the robot (E) to the cart (F). The evolutionary goal is to move the robot as fast as possible along the slide from the left to the right side of the aquarium.

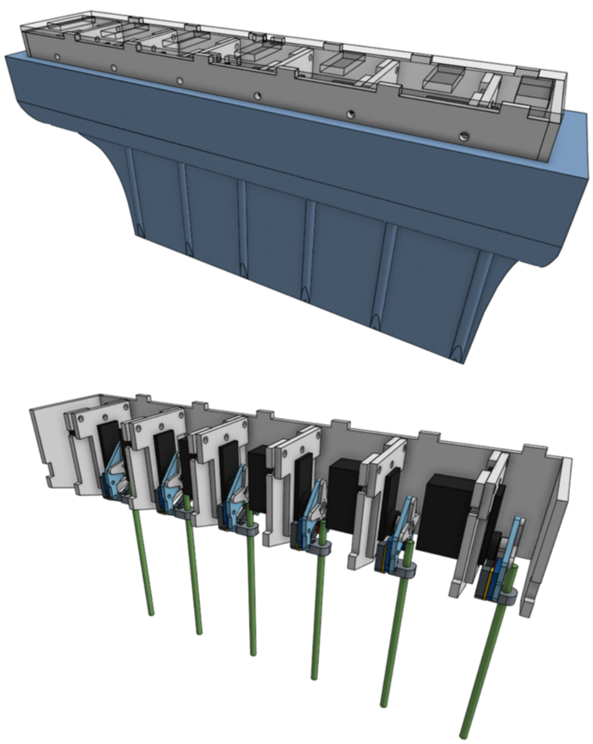


Figure 2: CAD design of the robotic knifefish. The white parts represent the laser cut acrylic parts, the blue part is the silicone part (top), the black parts depict the 6 servo motors that were used to actuate the fin rays. The bamboo sticks that serve as fin rays are displayed in green. The robot's full dimensions are 272x60x136mm and the fin is 70mm high and 210mm long. The fin rays are each spaced 40mm apart.

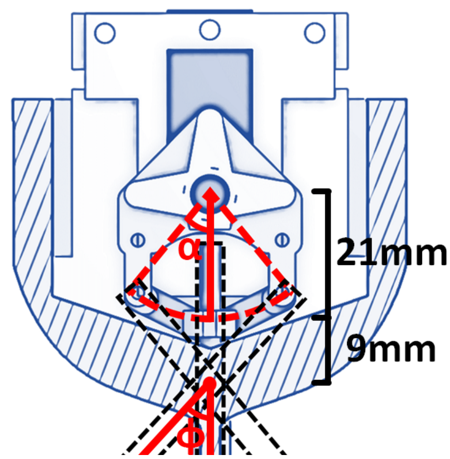


Figure 3: Cross section of the robot fish design. The red arc depicts the range of motion from the center of rotation of the servo motor to the plastic part that is connected to the crank mechanism. The red dot at the bottom in the hull depicts the approximate center of rotation of the fin ray.

2.2 Evolutionary Experiments

In our pre-experiments we implemented a generational evolutionary algorithm without crossover to create the genome for our robot controller. Due to the long evaluation time of the generational evolutionary algorithm, and servos being prone to overheating, we decided to implement Covariance Matrix Adaptation Evolutionary Strategy (CMA-ES) [12, 13] instead, to quickly find the basin of attraction and thereby speed up the evolutionary process[3].

[3]Our full implementation and the source code of the evolutionary algorithm and Arduino code can be found at https://github.com/FrankVeenstra/Knifefish_GECCO2018

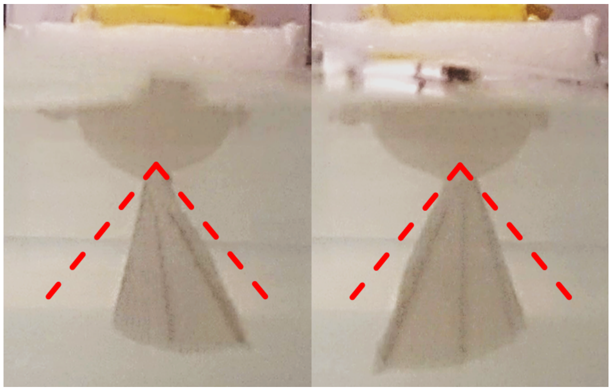


Figure 4: Angular deflection of the fin. Front view of the robot showing the angular deflection of the fin. The actual maximum angle of the fin can be seen to be less than the calculated angle (red dashed lines)

2.2.1 Encoding. The genome we created for an individual is composed of a string of 15 bytes. Each triple of three bytes translates into a sinusoidal function with a specific frequency, phase, and amplitude. The total of five sine functions are summed to yield the first five terms of a standard Fourier series. With this function we can approximate an arbitrary continuous periodic function and use it as a fin undulation pattern on the robot to be evaluated. The mutable parameters were the amplitude, phase, and frequency of each sinusoidal function. These parameters are converted into servo angles α_n for the 6 servo motors with the following function:

$$\alpha_n(t) = (\frac{g_1}{255} \cdot \theta_{max}) \cdot sin((g_3 \cdot t) + (g_2 \cdot n)) \quad (2)$$

where g_1, g_2 and g_3 represent the mutable parameters of a genome triple as bytes. θ_{max} is the maximum angle that the servo motors are allowed to move. n stands for adjacent servo motor numbers (values from 0 to 5) and t represents the time steps.

2.2.2 Evolutionary Algorithm. The evolutionary approach was divided into a control system and an evolutionary algorithm. The evolutionary algorithm made use of functions from the Distributed Evolutionary Algorithms in Python (DEAP) library which included an implementation of CMA-ES [11]. The CMA-ES implementation implemented a population size of 10 and ran for 20 generations. We found that CMA-ES was able to find similar solutions in 20 generations as running a normal generational evolutionary algorithm for 100 generations which was advantageous for limiting the duration of the experiments. Our CMA-ES implementation included an initial standard deviation value of 50 and a centroid value of 125 for every gene (half the max value of the bytes in the genome).

2.2.3 Controller System. An Arduino Mega 2560 controlled the robot by actuating the servo motors and received the sensor readings of the ultrasonic distance and infrared sensors. Through serial communication a genome is uploaded from a PC running the evolutionary algorithm to the Arduino Mega. The Arduino Mega evaluates an individual using the genome it received. This evaluation consists of:

(1) Move robot to the starting position (by using a manually coded swimming behavior)
(2) Move the servos to a central position and wait for six seconds (this delay was implemented to prevent overheating of servos and reduce waves in the tank)
(3) Evaluate genome for 10 seconds
(4) Send back a fitness value based on the distance the robot has traveled within the 10 seconds

All steps take roughly between 20-30 seconds for one individual depending on how far the robot was able to swim. When the same genome was evaluated multiple times the error difference in fitness was negligible (standard deviation of samples of size 4 was less than 1% for each run). Each individual is therefore only evaluated once.

A 20 ms delay was inserted between each time step for updating the servo angles. 500 time steps were done for each individual. The fitness value of each individual is calculated as a summation of the ultrasound distance measurements at every consecutive update of the servo positions. At each time step the ultrasonic distance sensor initiates a sound pulse and measures the time difference between the pulse and echo. This time interval becomes higher the further the robot moves away from its initial position. The fitness value for a controller that is not moving the robot lies around $100 \cdot 10^4$. At the start of the evaluation of a genome, the entire wave pattern for each servo is calculated for each time step. This requires six arrays to store 500 byte values derived from the genome. Although this takes up a lot of memory on the Arduino Mega, it circumvents doing calculations on the spot that might have caused an additional delay between every time step. A small delay is, however, caused by the ultrasonic sensor which requires an 8 microsecond delay for measuring the distance.

2.2.4 Experiments. Since earlier examples of robotic knifefish have been able to swim with only a single sinusoidal wave function as a control signal for the fin, we conduct experiments where the genome is reduced to three bytes that translate into the frequency, phase, and amplitude of a single sine function. We test if evolution is able to efficiently optimize these three parameters for increased swimming speed. Our second set of evolutionary experiments evaluate functions that are generated from all 15 mutable parameters, and yield the first five terms of a Fourier series. This is done to see whether an arbitrary periodic function can increase the performance compared to a single sine wave. For both sets of experiments we also test whether evolution will find swimming behaviors similar to the ones of actual knifefish, and if the performance of the evolved controllers can rival a manually programmed controller.

For both the sinusoidal and the Fourier series approach, 5 evolutionary runs were done with the exact same hardware setup. Since the slightest change in hardware and the environment can influence the evolutionary runs drastically, all the 10 runs were done consecutively. A manually coded swimming behavior is used as a baseline to compare with the evolved controllers. This behavior was the fastest swimming behavior we were able to find by manually adjusting the genome parameters during a two hour trial session with the platform. Its control function is:

$$\alpha_n = 40 \cdot sin((64 \cdot t) + (100 \cdot n)) \quad (3)$$

These control parameters correspond to a genome with the following three bytes: 255 for the amplitude, 64 for the phase, and 100 for the frequency.

2.3 Comparing Behaviors of the Robot with Actual Knifefish

Bale et al. [1] found that a diverse group of aquatic animals that use median/paired fin swimming, including knifefish, have evolved a similar optimal swimming strategy. More specifically, the result of dividing the length of an undulation on the fin by the mean amplitude of undulations along the fin, during steady swimming, consistently yields around 20. This wavelength, which maximizes the force generated by the body and the swimming speed, is referred to as the optimal specific wavelength (OSW). We therefore calculate the specific wavelength (SW) of our evolved undulation patterns to compare them with the swimming behaviors of knifefish. The SW is calculated by dividing the wavelength of undulation λ by the average amplitude of oscillation $\tilde{a}$. In general, this average amplitude $\tilde{a}$ is given by

$$\tilde{a} = h_{mean} \sin(\theta_{max}^{avg})/2 \quad (4)$$

Where θ_{max}^{avg} is the mean maximum angle of excursion of the fin rays and h_{mean} is the mean height of the fin.

3 RESULTS

3.1 Performance Analysis

After running CMA-ES for 20 generations using the sinusoidal and the Fourier series approaches, different wave patterns were acquired. Both evolutionary progressions of the 5 runs of each approach (Figure 5) evolved decent swimming behaviors though the Fourier series evolutionary progressions seem to have more variation in performance and did not plateau as clearly as the sinusoidal evolutionary progression. This corresponds to a larger, perhaps more convoluted, search space when evolving Fourier series.

The periodic control signals that have evolved in the sinusoidal approach are similar to each other while the best individuals of the Fourier series exhibit more erratic wave patterns (Fig. 6). Looking at the individual wave patterns and their corresponding fitness values, the best individual evolved in the Fourier series has a significantly higher fitness value than the others.

In Table 1 we compare the evolved swimming behaviors of our best candidates to see if the OSW ratio also applies here. The approximate wavelengths of the traveling waves have been obtained from ventral view video recordings of the robot with the best candidates and the manually coded behavior controlling its swimming. The average amplitude of oscillation was calculated from Equation 4 using a maximum angular excursion of 28 degrees (derived from video recordings) and that the fin height is 7 cm. The average travel speeds were also measured from video recordings (of the manual behavior and the best evolved individuals being replayed on the robot). Our inspiration the black ghost knifefish has a SW of 18.03 [1]. From Table 1 it can be seen that the best evolved sinusoidal controller has a specific wavelength of 16, i.e. it approximates, but is lower then, the optimal specific wavelength found by Bale et al. Although our manually programmed controller has a SW of 17 and comes closest to the actual knifefish, in reality it performed considerably worse than most of the evolved controllers (see Table 1).

Table 1: Specific Wavelengths and Travel Speeds of Behaviors. The evolved behaviors resulted in wave patterns with varied wavelengths and speeds. (Wavelength of Four. (Run 4) has been omitted as the wave function was to erratic for it to be measured from video recordings.)

Genome	Wavelength	SW	Speed (cm/s)
Manual	28 cm	17	3
Sine (Run 1)	26 cm	16	8
Sine (Run 2)	23 cm	14	6
Sine (Run 3)	26 cm	16	6
Sine (Run 4)	23 cm	14	6
Sine (Run 5)	24 cm	15	8
Four. (Run 1)	26 cm	16	4
Four. (Run 2)	26 cm	16	2
Four. (Run 3)	24 cm	15	5
Four. (Run 4)	-	-	5
Four. (Run 5)	22 cm	13	1

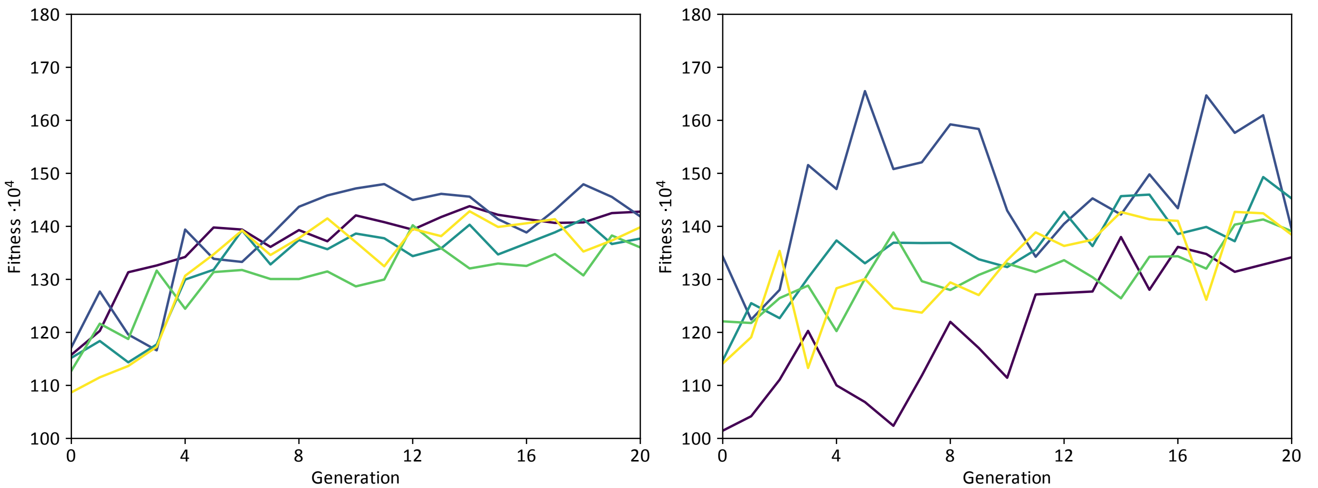
Being able to evolve wave patterns to control the swimming behavior of the robot is of limited use if their phenotype cannot be reproduced. Since the robot was slightly worn down after a lot of different experiments and several malfunctioning servo motors had been replaces, we evaluated the performance of the evolved wave patterns again. When comparing the evolved Fourier series wave patterns with the evolved sinusoidal wave patterns it can be seen that the sinusoidal wave patterns also outperform the manually encoded wave pattern significantly in terms of fitness value (Figure 7). Though this could have been caused by many factors, it seems that a sinusoidal function is a more robust general approach that might be suboptimal but resilient to morphological/environmental change

3.2 Phenotypic Analysis

To analyze the type of behaviors that evolved, the position of the tip of each fin ray was tracked in the best evolved individuals using footage taken from a ventral view of the robot (Figure 8). This tracking was done to analyze the actual undulation patterns across the fin as opposed to the calculated control patterns. Looking at the best evolved individuals from both the Fourier series and the sinusoidal approach, the wave propagates strikingly similar along the fin of both individuals. The phase and frequency are different for the two individuals but the sinusoidal wave pattern generates roughly the same wavelength as the Fourier series only with a higher frequency. The sinusoidal wave pattern makes roughly six undulations while the Fourier series makes five within the same time interval.

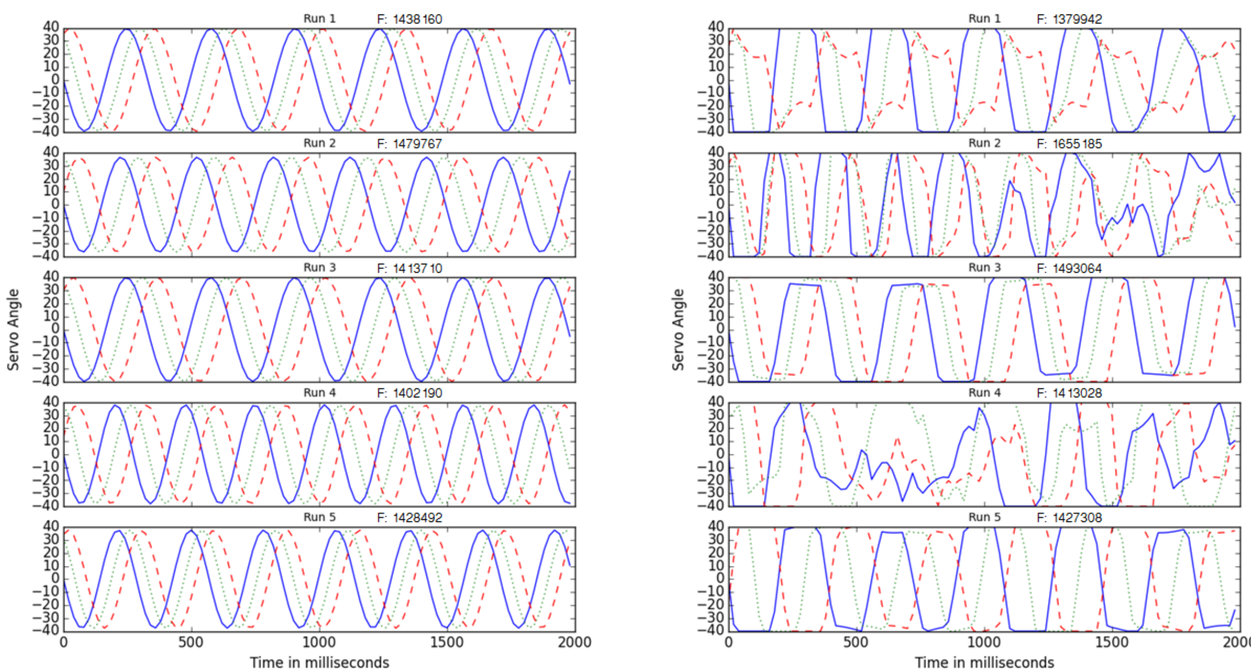
4 DISCUSSION

CMA-ES proved an efficient method for automatically evolving the swimming behavior of our soft robot whose morphology was inspired by the ghost knifefish. Although the search space was quite small, failing hardware was a problem that in general makes



(a) **(b)**

Figure 5: Evolutionary progressions of five runs . The sinusoidal approach (a) and the Fourier series approach (b) showing the maximum fitness (hall of fame) of the evolutionary runs.



(a) **(b)**

Figure 6: Evolved control wave patterns. The best evolved wave patterns in 5 distinct evolutionary runs using the sinusoidal approach (a) and the Fourier series approach (b). The graphs show two seconds of a resulting wave from each genome. The blue line represents the trajectory of the first servo motor while the green dotted and red dashed lines depict the positions of servos two and three respectively. The trajectories of servo four, five and six are not depicted. The difference in the wave of different servos visible in some of the Fourier series is due to including potentially high frequencies and querying the function every 20ms.

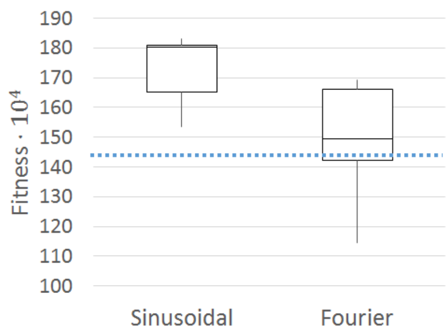


Figure 7: Performance difference between the best evolved sinusoidal and Fourier series individuals. The box plot shows the quartiles of the best individuals of the 5 runs of the sinusoidal approach and the Fourier series approach. These results were obtained from replaying the best genomes of the different approaches using a patched up version of the robotic fish (i.e. where the servo motors had been replaced). The blue dotted line represents the baseline performance of our manually encoded genome.

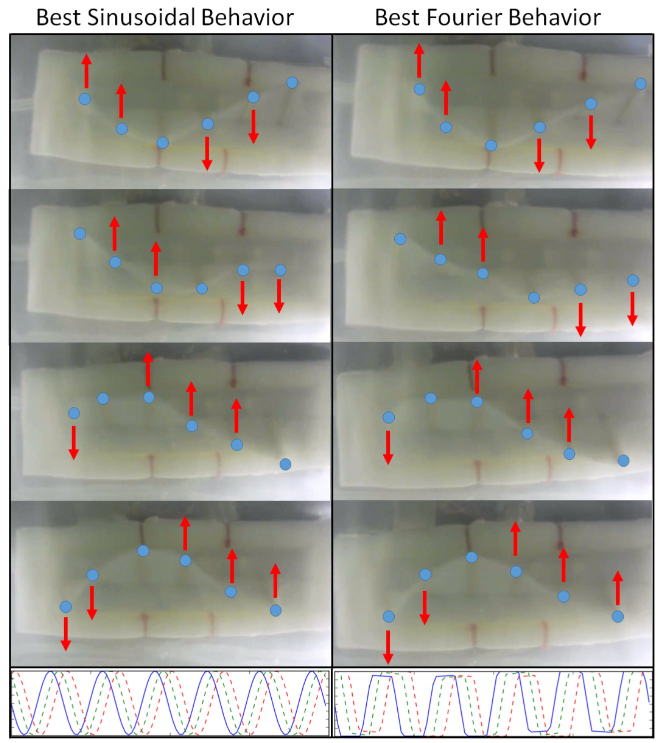


Figure 8: Evolved robot wave patterns. The wave patterns of the best (highest fitness) evolved sine wave and Fourier series seen from below. Both propagating waves are almost identical to one another and have a wavelength that is slightly longer than the length of the fin. The blue dots illuminate the tips of the fin rays while the red arrows depict the motion of the individual fin rays. The function plots below correspond to the fin undulations depicted above and are the best *reproducible* evolved wave patterns shown in Figure 6 (Sine (Run 1) and Fourier (Run 3))

evolving physical robots arduous. Predefining the controller by only utilizing periodic wave functions and only running CMA-ES for a brief period was enough to generate efficient swimming behavior. One of the main challenges when evolving physical robots is about how to deal with malfunctioning hardware. Considering a death toll of 17 servo motors during these experiments, using CMA-ES seemed a lot more viable compared to initial experiments with a generational evolutionary algorithm that took almost five times longer to get to results compared to the CMA-ES approach.

The robotic platform presented in this paper is constrained by predefined functions and the limited movement sets acquired in the evolutionary runs. However, the presented robot fish could potentially evolve many different behaviors that the knifefish is also capable of. This could make it a viable model for autonomous underwater vehicles. A next submersible iteration of the fish could evolve vertical thrust by sending counter-propagating waves towards and away from the center of the fin canceling out longitudinal forces as discussed by [9]. A selection of these behaviors could be evolved and encapsulated in a fixed environment, removing manual programming of the behavioral repertoire.

Zoological studies of knifefish kinematics have shown that the wavelength of the propagating wave varies across the fin during steady swimming [30]. Given that the swimming behavior of the knifefish has been optimized through natural evolution, implementing this feature in the encoding of the controller could probably lead to better performance. This could be accomplished by using a *compositional pattern-producing network* (*CPPN*) [29] with servo number and time as inputs. A similar approach has previously been used successfully to generate the oscillatory controller for a quadruped robot [21]. To discover a greater variety of controllers that perform well, novelty search [17] or other diversity enhancing methods can also be applied instead of a goal directed approach which is often prone to premature convergence or over-fitting. Another aspect worthy of further inquiry is the materials used for the fin. It is possible that a material with another elastic modulus might better exploit the interactions with the water to facilitate the emergence of dynamics that aid the swimming.

With our robotic platform we are able to automatically evolve the behavior of an intuitively functional soft robot using CMA-ES. Considering the increasing advances of automated manufacturing methods and readily available materials to create detailed robots with various features, we think this evolutionary approach on physical soft robots can become viable as a tool for directly optimizing the behavior of the physical systems.

5 CONCLUSION

In this paper we demonstrated that evolving the controller for a knifefish-inspired soft robot is feasible directly on the physical robot. The majority of the evolved behaviors outperformed a hand-designed controller in terms of speed. Additionally evolution was able to exploit the dynamical properties of the flexible material to produce feasible swimming strategies for the robot that have similar phenotypes but different genomes. We posit that evolutionary experiments on physical robots, which have so far only been applied to traditional rigid robots, are especially relevant for soft robots that are difficult to simulate computationally. In the future the presented approach could be combined with more explorative search methods such as novelty search and different fish models, to solve tasks for which even a simple hand-designed controller is an infeasible option.

ACKNOWLEDGEMENTS

This project has received funding from "Flora Robotica", a European Union's Horizon 2020 research and innovation program under the FET grant agreement, no. 640959.

REFERENCES

[1] Rahul Bale, Izaak D. Neveln, Amneet Pal Singh Bhalla, Malcolm A. MacIver, and Neelesh A. Patankar. 2015. Convergent Evolution of Mechanically Optimal Locomotion in Aquatic Invertebrates and Vertebrates. *PLOS Biology* 13, 4 (Apr 2015). https://doi.org/10.1371/journal.pbio.1002123

[2] A.A. Biewener. 2003. *Animal Locomotion.* Oxford University Press.

[3] Nicholas Cheney, Jeff Clune, and Hod Lipson. 2014. *Evolved Electrophysiological Soft Robots.* Vol. 14. 222–229. https://doi.org/10.7551/978-0-262-32621-6-ch037

[4] Nick Cheney, Robert MacCurdy, Jeff Clune, and Hod Lipson. 2013. *Unshackling evolution: evolving soft robots with multiple materials and a powerful generative encoding.* ACM, 167–174.

[5] Anthony J. Clark, Philip K. McKinley, and Xiaobo Tan. 2015. Enhancing a Model-Free Adaptive Controller Through Evolutionary Computation. In *Proceedings of the 2015 Annual Conference on Genetic and Evolutionary Computation (GECCO '15).* ACM, New York, NY, USA, 137–144. https://doi.org/10.1145/2739480.2754762

[6] E. Coevoet, A. Escande, and C. Duriez. 2017. Optimization-Based Inverse Model of Soft Robots With Contact Handling. *IEEE Robotics and Automation Letters* 2, 3 (July 2017), 1413–1419. https://doi.org/10.1109/LRA.2017.2669367

[7] F. Corucci, M. Calisti, H. Hauser, and C. Laschi. 2015. Evolutionary discovery of self-stabilized dynamic gaits for a soft underwater legged robot. In *2015 International Conference on Advanced Robotics (ICAR).* 337–344. https://doi.org/10.1109/ICAR.2015.7251477

[8] Francesco Corucci, Nick Cheney, Hod Lipson, Cecilia Laschi, and Josh Bongard. 2016. Material properties affect evolutions ability to exploit morphological computation in growing soft-bodied creatures. (Jul 2016). https://mitpress.mit.edu/sites/default/files/titles/content/alife16/ch043.html

[9] Oscar M. Curet, Neelesh A. Patankar, George V. Lauder, and Malcolm A. MacIver. 2011. Aquatic manoeuvering with counter-propagating waves: a novel locomotive strategy. *Journal of The Royal Society Interface* 8, 60 (Jul 2011), 1041–1050. https://doi.org/10.1098/rsif.2010.0493

[10] Oscar M. Curet, Neelesh A. Patankar, George V. Lauder, and Malcolm A. MacIver. 2011. Mechanical properties of a bio-inspired robotic knifefish with an undulatory propulsor. *Bioinspiration & Biomimetics* 6, 2 (2011). https://doi.org/10.1088/1748-3182/6/2/026004

[11] Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner, Marc Parizeau, and Christian Gagné. 2012. DEAP: Evolutionary Algorithms Made Easy. *Journal of Machine Learning Research* 13 (jul 2012), 2171–2175.

[12] Nikolaus Hansen, Sibylle D. Müller, and Petros Koumoutsakos. 2003. Reducing the Time Complexity of the Derandomized Evolution Strategy with Covariance Matrix Adaptation (CMA-ES). *Evol. Comput.* 11, 1 (March 2003), 1–18. https://doi.org/10.1162/106365603321828970

[13] Nikolaus Hansen and Andreas Ostermeier. 2001. Completely Derandomized Self-Adaptation in Evolution Strategies. *Evol. Comput.* 9, 2 (June 2001), 159–195. https://doi.org/10.1162/106365601750190398

[14] J. Hiller and H. Lipson. 2012. Automatic Design and Manufacture of Soft Robots. *IEEE Transactions on Robotics* 28, 2 (Apr 2012), 457–466. https://doi.org/10.1109/TRO.2011.2172702

[15] Nick Jakobi, Phil Husbands, and Inman Harvey. 1995. *Noise and the reality gap: The use of simulation in evolutionary robotics.* Springer, Berlin, Heidelberg, 704–720. https://doi.org/10.1007/3-540-59496-5_337

[16] Sam Kriegman, Nick Cheney, Francesco Corucci, and Josh C. Bongard. 2017. A Minimal Developmental Model Can Increase Evolvability in Soft Robots. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO '17).* ACM, New York, NY, USA, 131–138. https://doi.org/10.1145/3071178.3071296

[17] Joel Lehman and Kenneth O Stanley. 2011. Abandoning objectives: Evolution through the search for novelty alone. *Evolutionary computation* 19, 2 (2011), 189–223.

[18] K. H. Low. 2009. Modelling and parametric study of modular undulating fin rays for fish robots. *Mechanism and Machine Theory* 44, 3 (Mar 2009), 615–632. https://doi.org/10.1016/j.mechmachtheory.2008.11.009

[19] K. H. Low and A. Willy. 2006. Biomimetic Motion Planning of an Undulating Robotic Fish Fin. *Journal of Vibration and Control* 12, 12 (Dec 2006), 1337–1359. https://doi.org/10.1177/1077546306070597

[20] M. A. MacIver, E. Fontaine, and J. W. Burdick. 2004. Designing future underwater vehicles: principles and mechanisms of the weakly electric fish. *IEEE Journal of Oceanic Engineering* 29, 3 (Jul 2004), 651–659. https://doi.org/10.1109/JOE.2004.833210

[21] Gregory Morse, Sebastian Risi, Charles R Snyder, and Kenneth O Stanley. 2013. Single-unit pattern generators for quadruped locomotion. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation.* ACM, 719–726.

[22] Bobak Mosadegh, Panagiotis Polygerinos, Christoph Keplinger, Sophia Wennstedt, Robert F. Shepherd, Unmukt Gupta, Jongmin Shim, Katia Bertoldi, Conor J. Walsh, and George M. Whitesides. 2014. Pneumatic Networks for Soft Robotics that Actuate Rapidly. *Advanced Functional Materials* 24, 15 (Apr 2014), 2163–2170. https://doi.org/10.1002/adfm.201303288

[23] John Rieffel, Davis Knox, Schuyler Smith, and Barry Trimmer. 2013. Growing and Evolving Soft Robots. *Artificial Life* 20, 1 (Feb 2013), 143–162. https://doi.org/10.1162/ARTL_a_00101

[24] John Rieffel, Jean-Baptiste Mouret, Nicolas Bredeche, and Evert Haasdijk. 2017. Introduction to the Evolution of Physical Systems Special Issue. *Artificial Life* 23, 2 (2017), 119–123. https://doi.org/10.1162/ARTL_e_00232

[25] Daniela Rus and Michael T Tolley. 2015. Design, fabrication and control of soft robots. *Nature* 521, 7553 (2015), 467.

[26] Frank Saunders, Ethan Golden, Robert D. White, and Jason Rife. 2011. Experimental verification of soft-robot gaits evolved using a lumped dynamic model. *Robotica* 29, 6 (2011), 823–830. https://doi.org/10.1017/S0263574711000014

[27] Michael Sfakiotakis, John Fasoulas, Manolis M. Kavoussanos, and Manolis Arapis. 2015. Experimental investigation and propulsion control for a bio-inspired robotic undulatory fin. *Robotica* 33, 5 (2015), 1062–1084. https://doi.org/10.1017/S0263574714002926

[28] Mohsen Siahmansouri, Ahmad Ghanbari, and Mir Masoud Seyyed Fakhrabadi. 2011. Design, Implementation and Control of a Fish Robot with Undulating Fins. *International Journal of Advanced Robotic Systems* 8, 5 (2011), 60. https://doi.org/10.5772/50898 arXiv:https://doi.org/10.5772/50898

[29] Kenneth O. Stanley. 2007. Compositional Pattern Producing Networks: A Novel Abstraction of Development. *Genetic Programming and Evolvable Machines* 8, 2 (June 2007), 131–162. https://doi.org/10.1007/s10710-007-9028-8

[30] Eric D. Youngerman, Brooke E. Flammang, and George V. Lauder. 2014. Locomotion of free-swimming ghost knifefish: anal fin kinematics during four behaviors. *Zoology (Jena, Germany)* 117, 5 (Oct 2014), 337–348. https://doi.org/10.1016/j.zool.2014.04.004