**A food foraging C. elegans robot**

Tom Portegys1, Shane Gingell2,

1OpenWorm Foundation, Boston MA, USA; 2Out of the BOTS, Nambour, Queensland, Australia

**Keywords:** Robotics, Computational Biology, *Caenorhabditis elegans*

Abstract

One of the goals of the OpenWorm project is to promote awareness of the biology of the C. elegans nematode worm. In the robotics subproject, this takes the form of creating physical implementations of the worm which approximate prevalent scientific models. The downside of this is that C. elegans is, of course, not a robot. However, simulating the worm in robotic form manifests a tangible aspect that pure software models do not possess. A worm robot also sidesteps working with the actual worm, a transparent 1mm organism, which requires special conditions and equipment, such as lighting and microscopes. Another aim of the robotics effort is to foster crossover education in biology, robotics, and coding. This might take the form of either specifying or producing kits of parts that can be assembled by students in school settings or by generally interested parties. This paper discusses other biological robotics efforts and describes in detail the latest OpenWorm robots produced in conjunction with *Out of the BOTS* robotics.

Keywords: robotics, C. elegans, OpenWorm

1. Introduction

The OpenWorm project’s stated goal is the creation of a virtual C. elegans. In a true spirit of scientific and technical curiosity, however, it can be viewed as an umbrella of initiatives that both further this primary aim but also support a number of secondary goals and activities. The education committee is chartered with promoting education about computational biology using C. elegans as a model subject, for example. The robotics subproject in turn aims to produce a robotic worm that can be assembled and programmed by students and other interested parties. This combines coding, robotics, and computational biology and is intended to serve as a classroom learning project.

Another purpose of the robot project is to be able to encounter and solve problems that often are not visible to, or are easily bypassed by entirely simulated solutions. And in doing so suggest possible biological mechanisms within C. elegans. For example, in designing a mechanism for food foraging, one must take into account that the worm senses salt gradients as it moves in its environment [1]. Although it is not entirely clear how this works in the worm, a method that senses a signal gradient in a moving robot could trigger ideas about further biological research on this topic.

The noted physicist Richard Feynman famously wrote, "What I cannot create, I do not understand". There is an invaluable back-and-forth flow of ideas and insights between analysis and synthesis. For example, the invention of the steam engine helped discover thermodynamics principles. In a robotic implementation, there are factors such as friction, noise, and general analog irregularities that are in some ways more in kinship with a physical body, even one made of animal tissue, than a pure simulation.

The biological roboticist Jeff Krichmar put it this way: ‘The often used phrase, ‘‘understanding through building’’, implies that one can get a deep understanding of a system by constructing physical artifacts that can operate in the real-world. In building and studying neurobiologically inspired robots, scientists must address theories of neuroscience that couple brain, body, and behavior.’ [2]. Arguments from the field of Artificial Intelligence can also be brought to bear here, where robotics researchers such as Brooks [3], Hoffmann and Pfeifer [4] have opined that true authenticity can only be achieved by machines that have sensory and motor skills and are connected to the world through a body.

Furthermore, the National Science Foundation’s Special Report on Robots and Biology [5] reports that, “Evolution has produced a wide range of intelligent, mobile sensor units in the form of living organisms ranging from insects to humans. Compared to current robots, insects and other animals often have much more flexible and efficient control of their movements. Some researchers study organisms to develop better robots, while others build robots to better understand the organisms.” Numerous projects are leveraging nature to guide the development of robotics. For example, an interdisciplinary team of researchers is using a grant to study how the complex movements of octopuses can be used to design a bio-inspired, soft-armed, autonomous robot [6].

Considering the educational rationale of the robotic C. elegans, robots as classroom projects are gaining popularity and credibility as approaches for studying technology and science, integrating activities that explore technological models of scientific phenomena. To explain difficult abstract concepts in science and math subjects, teachers are using robotics kits that include various types of robots, including quadcopters, robotic arms, etc. [7]. Cuperman and Verner [8] have implemented this in a method where a learner inquires into a biological phenomenon and develops its representation in the form of a robotic model.

The C. elegans robot was demonstrated as part of a public program of events associated with the conference that was hosted by OpenWorm at the Sainsbury Wellcome Centre for Neural Circuits and Behaviour in London, UK on January 31st 2018. In May of 2019 it went on display at the Ars Electronica Center in Linz, Austria [9].

# 2. Description

This paper describes two versions of the C. elegans robot. The first is a Raspberry Pi [10] processor implementation, and the second is an ESP32 MicroPython system on a chip (SoC) [11] implementation. Both implementations are a combined effort of the authors. Gingell spearheaded the design and initial build and Portegys built the version used for the demo as well as supplying software to run the simulation. Future collaborations might include other interested parties.

The 3D models, code, assembly instructions and parts lists are available at:

<https://github.com/openworm/robots>

## 2.1 Raspberry Pi implementation

**Hardware**

Figure 1 shows a top view of the robot with major components denoted. Figure 2 shows a side view. The robot consists of 9 articulated segments, each segment mounted on a pair of wheels. Locomotion is achieved, as it is in C. elegans, by moving in a snakelike manner that relies on surface friction. The wheels are hence not powered and exist to provide a suitable contact surface with the ground. Each segment is a 3D printed component that articulates with its neighbors via servos. The electronic components are mounted on platforms fastened to several of the segments. The Raspberry Pi Zero W (RPi) [10] is a full-fledged microprocessor that included wireless communication capability. A PWM board distributes power and controls signals from the RPi to the servos. Each servo is capable of maintaining a specified angular position that translates to inter-segment angular positioning.

# 

Figure 1 – Top view of robot.

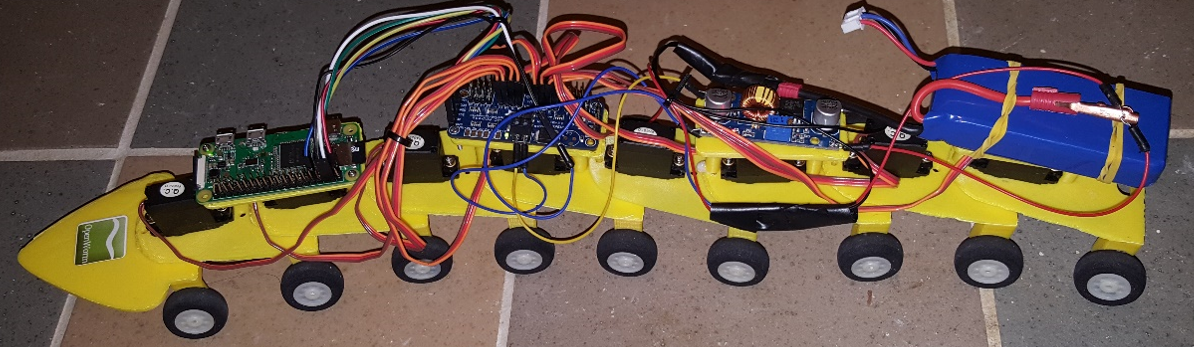


Figure 2 – Side view of robot.

Figure 3 shows the designs for the three 3D printed parts. These parts are specified in a common .stl file format that is editable and portable to most 3D printers. On the left is the segment part. In the center is the head that is envisioned to be mounted with sensors for food foraging and touch. On the right is one of the platforms for mounting the electronic components.

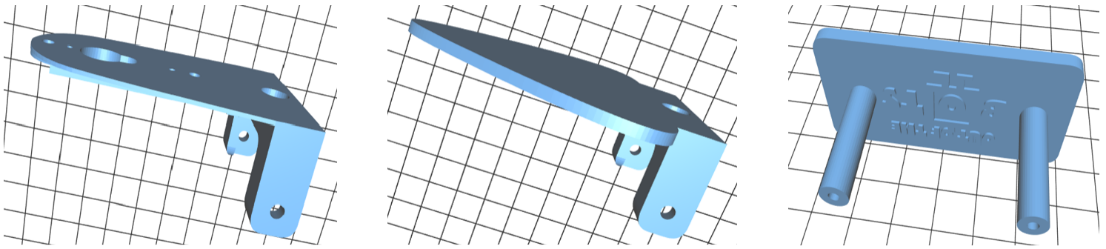


Figure 3 – 3D printed body parts.

Figure 4 shows how the segments are articulated. A servo is mounted on the front top of the segment with its geared shaft extending into an aperture in the next forward segment. An arm secured to the gear sits in a recession in the segment that transfers the angular movement of the gear into angular movements between the segments.

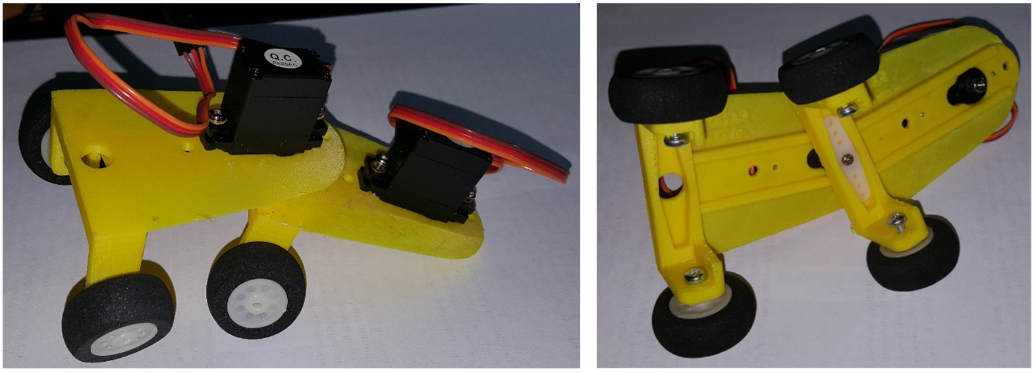


Figure 4 – Segment subassembly.

###### Programming and control

## The RPi replays a recorded sequence of segment angle positions partially shown in Figure 5. Each line is processed at a set interval of time, and the angles are directly distributed to the servos. It can be seen that the anterior segments receive signals or greater magnitude than the posterior segments. The overall effect is to achieve an undulating forward motion.

[78,-0.63,-0.65,-0.62,-0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[219,-0.63,-0.65,-0.62,-0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[375,-0.63,-0.65,-0.62,-0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[531,0.63,0.00,-0.62,-0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[672,0.63,0.65,-0.62,-0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[812,0.63,0.65,0.00,-0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[937,0.63,0.65,0.62,-0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[1047,0.63,0.65,0.62,0.00,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[1156,0.63,0.65,0.62,0.58,-0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[1344,0.63,0.65,0.62,0.58,0.55,-0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[1515,-0.63,0.65,0.62,0.58,0.55,0.51,-0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[1672,-0.63,0.65,0.62,0.58,0.55,0.51,0.00,-0.44,-0.41,-0.37,-0.34,-0.30],

[1781,-0.63,0.00,0.62,0.58,0.55,0.51,0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[1906,-0.63,-0.65,0.62,0.58,0.55,0.51,0.48,-0.44,-0.41,-0.37,-0.34,-0.30],

[2031,-0.63,-0.65,0.62,0.58,0.55,0.51,0.48,0.44,-0.41,-0.37,-0.34,-0.30],

[2140,-0.63,-0.65,0.00,0.58,0.55,0.51,0.48,0.44,-0.41,-0.37,-0.34,-0.30],

[2265,-0.63,-0.65,-0.62,0.58,0.55,0.51,0.48,0.44,0.41,-0.37,-0.34,-0.30],

[2375,-0.63,-0.65,-0.62,0.58,0.55,0.51,0.48,0.44,0.41,0.37,-0.34,-0.30],

[2547,-0.63,-0.65,-0.62,-0.58,0.55,0.51,0.48,0.44,0.41,0.37,0.34,0.30],

[2656,-0.63,-0.65,-0.62,-0.58,0.55,0.51,0.48,0.44,0.41,0.37,0.34,0.30],

[2875,0.63,-0.65,-0.62,-0.58,-0.55,0.51,0.48,0.44,0.41,0.37,0.34,0.30],

Figure 5 – Sample recorded angular movement data.

The data is recorded from an off-line run of a simulation of the worm developed by Boyle, Berri, and Cohen [12]. This simulation is a fine-grained model of the worm’s neuromuscular system that achieves locomotion. Unfortunately the simulation entails a computation-intensive physics engine that is too demanding for the RPi, hence the need for the recording/replay operations. An Android smartphone app is also available that contains the simulator functionality [13].

The recording is transferred to the RPi via the *scp* (secure copy) command. To run the robot, the user logins in via *ssh* and executes a Python script that initializes the servos and processes the recorded file line-by-line.

A short video of the robot in action can be viewed here: <https://www.youtube.com/watch?v=z1zARL2_4oM>

## 2.2 ESP32 implementation

The ESP32 is a system on a chip (SoC) [11] that supports a MicroPython interface. The board contains both control and power distribution connections for the servos as well as connections to support other peripherals, such as sensors. This simplifies the design by dispensing with the PWM and voltage regulator boards required by the RPi implementation. The implementation can be seen in Figure 6.

Two improvements were done the ESP32 implementation. First, although the 3D printed body segments remain fully compatible, these were laser cut for a sturdier body. Second, an ultrasonic distance sensor was added to allow the robot to forage for simulated food by seeking it out by proximity. This is an approximation of how C. elegans follows chemical gradients in its environment to find the nearest food source [1].



Figure 6 – ESP32 implementation with ultrasonic distance sensor.

The ultrasonic sensor has a wide (30+ degrees) view field. To achieve successful navigation toward the nearest object, a distance measurement is taken at 45 degrees from center as the robot swings its head from side to side. An increment is added to the segment servo angles depending on which side the object is. The following algorithm accomplishes locomotion combined with distance sensing:

turn\_angle\_delta = 0

for each step:

# Set servo angles.

for each segment:

servo\_angle = angles\_array[step][segment] + turn\_angle\_delta

set\_servo(segment, servo\_angle)

# Check right sensor distance.

if head swing moving right and head angle is 45 degrees:

right\_distance = activate\_sensor()

# Check left sensor distance.

if head swing moving left and head angle is -45 degrees:

left\_distance = activate\_sensor()

# Determine turn direction.

if right\_distance <= goal\_distance or left\_distance <= goal\_distance:

print("food found")

break

else if right\_distance > max\_valid\_distance and left\_distance > max\_valid\_distance:

turn\_angle\_delta = 0

print("food forward")

else if right\_ping\_distance < left\_ping\_distance:

turn\_angle\_delta = right\_turn\_angle\_delta

print("food right")

else if left\_ping\_distance < right\_ping\_distance:

turn\_angle\_delta = left\_turn\_angle\_delta

print("food left")

else:

turn\_angle\_delta = 0

print("food forward")

Videos of the robot can be found here:

Moving without sensor: <https://www.youtube.com/watch?v=ta4kQ2Gs8iM>

Foraging food to right: <https://www.youtube.com/watch?v=OIZyOVygVxw>

Foraging food to left: <https://www.youtube.com/watch?v=BD-n1SASU4I>

## 2.3 Future implementations

Future implementations could include:

* Touch response. C. elegans responds to touch on its nose by reversing direction.
* A Python Notebook that will provide a programming interface to the robot. This can also support software exercises related to C. elegans, such as how the neural network can be optimized to achieve sensorimotor coordination for touch response, for example.

Both implementations of the robot can be constructed by someone with moderate technical skills after obtaining the parts from various vendors. A more ambitious plan is to pre-order and box the parts for single purchase availability.

# 3. Conclusion

We have presented robotic implementations of C. elegans that are intended to adhere to and demonstrate the behavioral biology of the worm as closely as possible. Future versions will continue this design goal. We hope to produce a robot kit that is suitable for students to learn about the fascinating biology of the worm as well as to hone skills in robotics and coding.

# References

1. Izquierdo, E. and Beer, R. D. (2015). "[An Integrated Neuromechanical Model of Steering in C. elegans](https://mitpress.mit.edu/sites/default/files/titles/content/ecal2015/978-0-262-33027-5-ch040.pdf)", *ECAL15*.
2. Krichmar, J.L., Conradt, J., and Asada, M. (2015). “Neurobiologically Inspired Robotics: Enhanced Autonomy through Neuromorphic Cognition”. *Neural Networks*, 72, 1-2.
3. Brooks, R. (1999). Cambrian Intelligence: The Early History of the New AI. Cambridge MA: The MIT Press.ISBN 0-262-52263-2.
4. Hoffmann, M. and Pfeifer, R. (2011). “The implications of embodiment for behavior and cognition: animal and robotic case studies”, in W. Tschacher & C. Bergomi, ed., 'The Implications of Embodiment: Cognition and Communication', Exeter: Imprint Academic, pp. 31-58.
5. Website: <https://www.nsf.gov/news/special_reports/robotics/robots.jsp>
6. Fisher, R., et al. (2017). “Octopus-Inspired Autonomous Arms for Soft Robots with Adaptive Motions,” Office of Naval Research grant. <http://phoenixmed.arizona.edu/about/news/researchers-study-biology-octopuses-develop-autonomous-robots>
7. Schwartz, K. (2014). “Robots in the Classroom: What Are They Good For?” <https://www.kqed.org/mindshift/35611>
8. Cuperman, D. and Verner, I. M. (2013). “Learning through creating robotic models of biological systems”, *International Journal of Technology and Design Education,* Volume 23, Issue 4, pp 849–866.
9. Ars Electronica Center: <https://www.linztourismus.at/freizeit/linz-entdecken/museen-und-ausstellungen/ars-electronica-center/>
10. Raspberry Pi: <https://www.raspberrypi.org/>
11. ESP32 system on a chip: <http://esp32.net>
12. Boyle, J., Berri, S., and Cohen, N. (2012). “[Gait modulation in C. elegans: an integrated neuromechanical model](http://www.frontiersin.org/Computational_Neuroscience/10.3389/fncom.2012.00010/abstract)”, *Front. Comput. Neurosci*.
13. Worm simulator Android smartphone app: <http://wormworx.portegys.com>