Evolution of 3-Dimensional Soft Robots

Corbin Frisvold

Jim Thorpe Area School District

Feb 29, 2020

Background

Why did I choose this project?

1. Intersection of Computer Science, Mathematics, and Biology.
2. Interest in 3D Printing, 3D Design, and evolutionary design.
3. Expansion upon work of MIT's Karl Sims

Terminology

1. Soft Robots
2. Compositional Pattern Producing Network
3. Fitness

Objectives

1. Develop a way to generate and simulate soft robots
2. Evaluate fitness of said robots, and explore 'peaks'
3. Create an evolutionary algorithm to create and refine robots

Materials

Materials used for the program

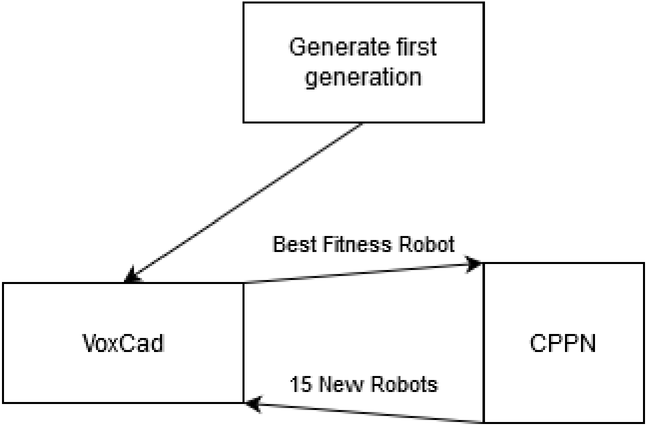
1. Python 2.7
    1. NumPy
2. VoxCad Soft Robotics Library
3. QT 5.14.1

Programming and Methods

There were three main sections to my program

1. Setup VoxCad with QT
2. Implement physics engine into Python
3. Design a CPPN for evolution

Program Flow



Generate first generation

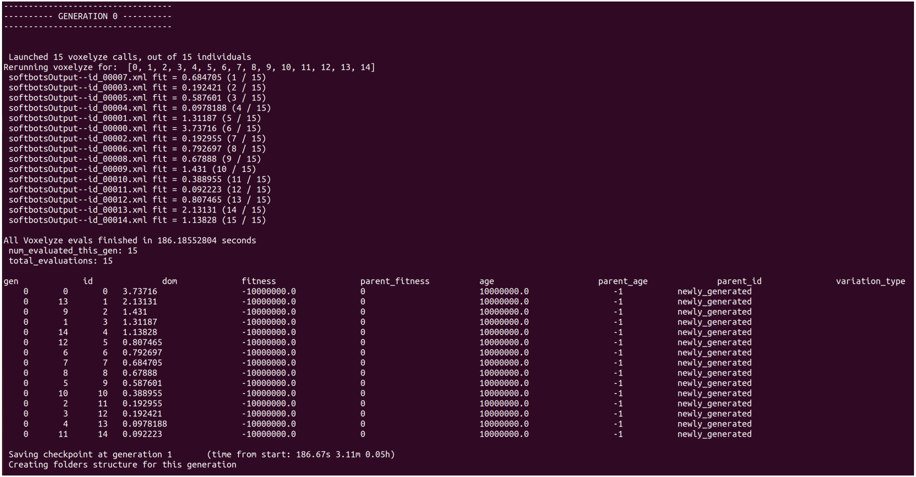
Best Fitness Robot

VoxCad

CPPN

15 New Robots

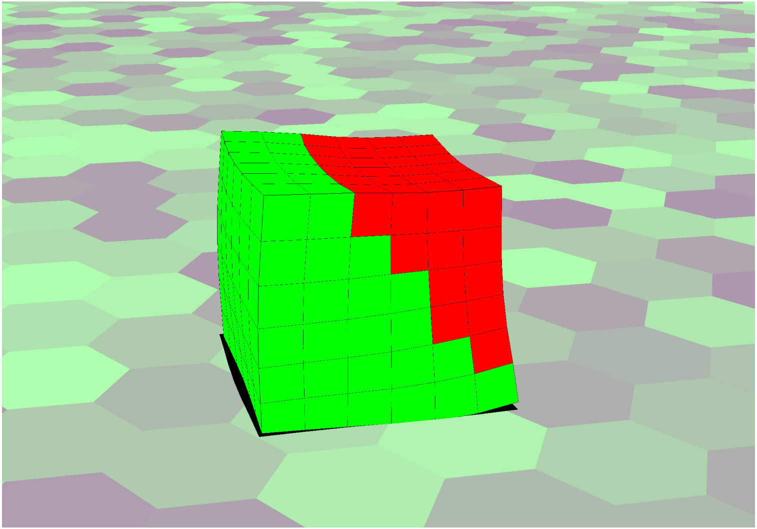
Final Program



```
----------------------------------
---------- GENERATION 0 ----------
----------------------------------


 Launched 15 voxelyze calls, out of 15 individuals
Rerunning voxelyze for:  [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
 softbotsOutput--id_00007.xml fit = 0.684705 (1 / 15)
 softbotsOutput--id_00003.xml fit = 0.192421 (2 / 15)
 softbotsOutput--id_00005.xml fit = 0.587601 (3 / 15)
 softbotsOutput--id_00004.xml fit = 0.0978188 (4 / 15)
 softbotsOutput--id_00001.xml fit = 1.31187 (5 / 15)
 softbotsOutput--id_00000.xml fit = 3.73716 (6 / 15)
 softbotsOutput--id_00002.xml fit = 0.192955 (7 / 15)
 softbotsOutput--id_00006.xml fit = 0.792697 (8 / 15)
 softbotsOutput--id_00008.xml fit = 0.67888 (9 / 15)
 softbotsOutput--id_00009.xml fit = 1.431 (10 / 15)
 softbotsOutput--id_00010.xml fit = 0.388955 (11 / 15)
 softbotsOutput--id_00011.xml fit = 0.092223 (12 / 15)
 softbotsOutput--id_00012.xml fit = 0.807465 (13 / 15)
 softbotsOutput--id_00013.xml fit = 2.13131 (14 / 15)
 softbotsOutput--id_00014.xml fit = 1.13828 (15 / 15)

All Voxelyze evals finished in 186.18552804 seconds
 num_evaluated_this_gen: 15
 total_evaluations: 15

gen             id              dom             fitness                 parent_fitness          age                     parent_age              parent_id               variation_type
    0       0       0   3.73716                 -10000000.0             0                       10000000.0                 -1           newly_generated
    0      13       1   2.13131                 -10000000.0             0                       10000000.0                 -1           newly_generated
    0       9       2   1.431                   -10000000.0             0                       10000000.0                 -1           newly_generated
    0       1       3   1.31187                 -10000000.0             0                       10000000.0                 -1           newly_generated
    0      14       4   1.13828                 -10000000.0             0                       10000000.0                 -1           newly_generated
    0      12       5   0.807465                -10000000.0             0                       10000000.0                 -1           newly_generated
    0       6       6   0.792697                -10000000.0             0                       10000000.0                 -1           newly_generated
    0       7       7   0.684705                -10000000.0             0                       10000000.0                 -1           newly_generated
    0       8       8   0.67888                 -10000000.0             0                       10000000.0                 -1           newly_generated
    0       5       9   0.587601                -10000000.0             0                       10000000.0                 -1           newly_generated
    0      10      10   0.388955                -10000000.0             0                       10000000.0                 -1           newly_generated
    0       2      11   0.192955                -10000000.0             0                       10000000.0                 -1           newly_generated
    0       3      12   0.192421                -10000000.0             0                       10000000.0                 -1           newly_generated
    0       4      13   0.0978188               -10000000.0             0                       10000000.0                 -1           newly_generated
    0      11      14   0.092223                -10000000.0             0                       10000000.0                 -1           newly_generated

 Saving checkpoint at generation 1       (time from start: 186.67s 3.11m 0.05h)
 Creating folders structure for this generation
```

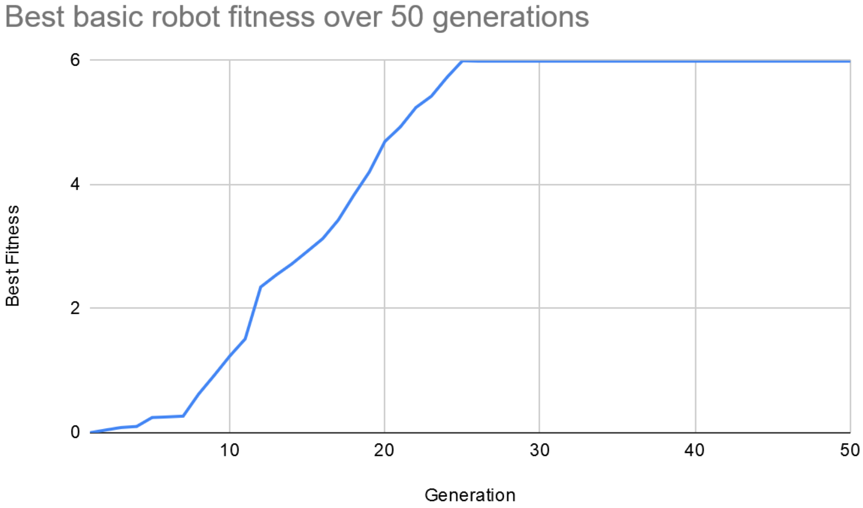
Basic Robot

Begin with a set of 15 6 x 6 x 6 cube robots, evolve its shape alone.



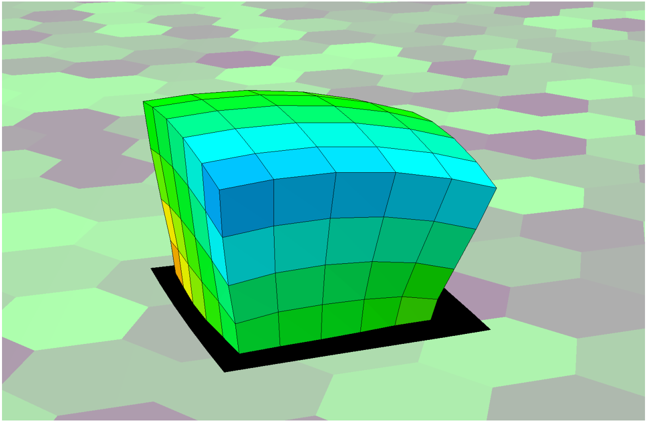
Basic Robot Evolution

After 50 generations, best fitness was 5.9961, plateau after 25



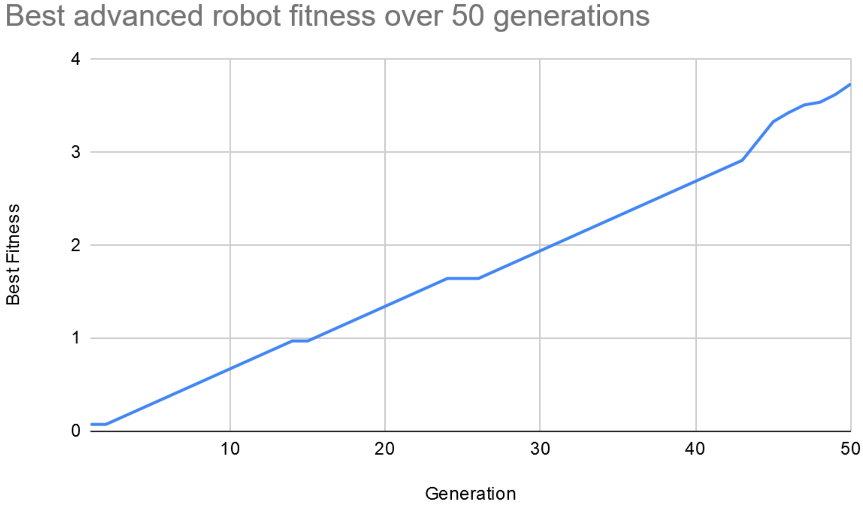
Advanced Robot

Begin with another 15 basic robots, except allow them to change densities, material stiffness, and size.



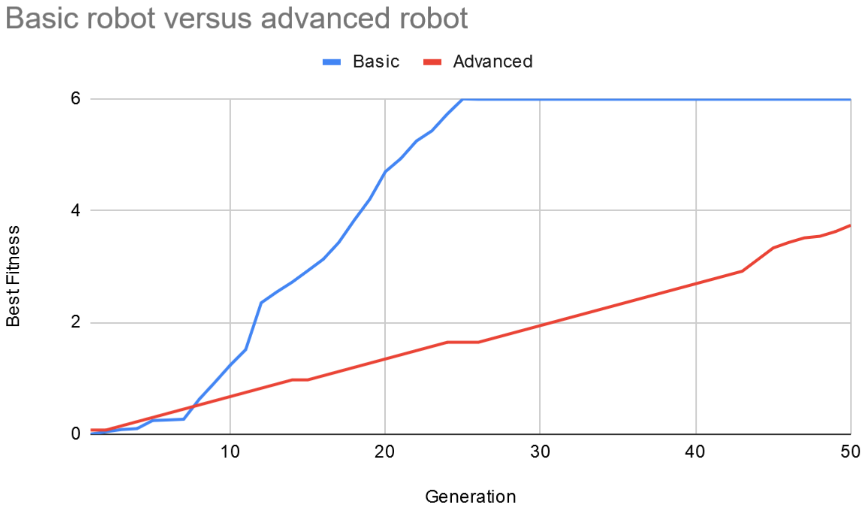
Advanced Robot Evolution

After 50 generations, best fitness was 3.7372, didn't plateau during trials.



Analysis: Basic vs Advanced Robot

Look at the Basic Robot vs. Growth Robot



Analysis: Factors Affecting Evolutionary Plateau

1. Size and shape
2. Control over material types
3. Control over material densities
4. Possibility of environment

Possible Errors

1. Physics engine error
    1. Speed increase
    2. Large batch size
2. Material wear
3. Environment
    1. Temperature
    2. Drag

Real World Applications

1. Military
    1. Nuclear extraction
2. Medical
    1. Surgery
    2. Physical therapy
3. Civilian
    1. 3D-Printing
    2. Research

Conclusion

1. The program successfully simulates and evolves robots
2. Generally user friendly, libraries can be cleaner
3. VoxCad interface makes analysis easy, all data in XML
4. Wide range of applications, from military to household
5. Possible future improvements include:
   1. More environmental factors, currently only temperature
   2. Material wear simulation
   3. Other neural network types may work better for evolution

Resources

1. Dynamic Simulation of Soft Multimaterial 3D-Printed Objects (2014) Jonathan Hiller and Hod Lipson
2. https://www.python.org/
3. http://www.numpy.org/
4. https://www.qt.io/
5. https://www.creativemachineslab.com/voxcad.html

Questions?