Artificial Life with Cognitive Sciences Labs

Report 4

Evolutionary Design in Framsticks

Dmytro Romaniv, 151958, group 1
Łukasz Sztukiewicz, 151959, group 1
Michał Wiliński, 151938, group 1

24th January, 2022

Abstract

In this experiment, we researched different combinations of evolutionary parameters to evolve a basic Framstick creature into our desired fast hopper. Computations were done using batch simulating with scripts. Our exploration helped us in understanding the meaning of different genetic operators, morphology changes and how they influence experiments and their convergence.

Contents

1 Evolutionary goal

For this experiment, our evolutionary goal of choice is to engender a grasshopperlike creature with great vertical and horizontal velocity. We tried to achieve that goal by maximizing the above-mentioned velocities. We had some failed attempts which made us change our evolutionary goal multiple times but after thousands of iterations, we can finally present the results of our hard work.

2 Default and modified sets of parameters

2.1 Default parameters

- Gene pool: 200;

- Selection:

Unchanged	10%
Mutated	80%
Crossed over	10%
Selection rule	fitness proportional
Delete genotypes	randomly

- Custom fitness function:

```
return 0.0+this.velocity*20.0+this.vertvel*20.0;
```

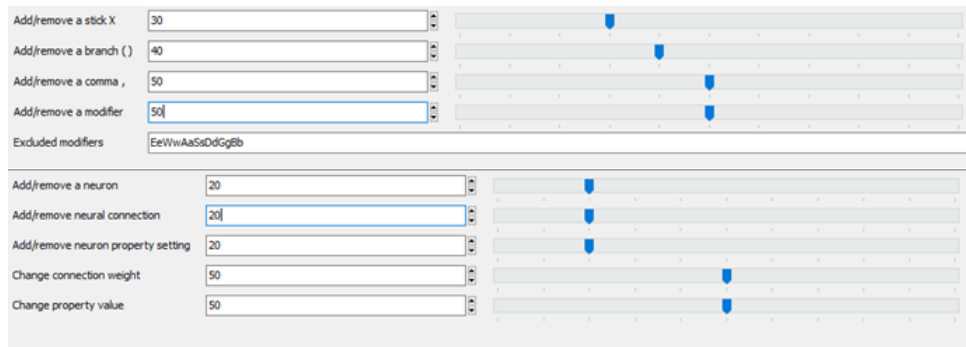- Termination criterion: 30 000 evaluations;

- Encoding: f1;

1

Figure 1: Default parameters of f1 encoding for the experiment.

2.2 Modification 1

Modifications:

- No crossover;

- Selection - unchanged 40%;
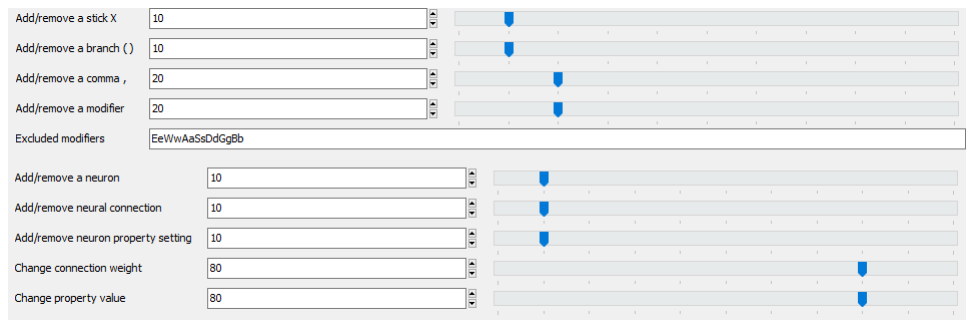
- Selection - mutated 60%;

Figure 2: Modified parameters of f1 encoding for the experiment (mod 1).

2.3 Modification 2

Modifications:

- Gene pool - raised from 200 to 1000;

3   Experiment

Firstly, we tried creating an initial population with the below script, but when we investigated convergence of basic initial genotype compared to our code-generated, we found out that we worsened convergence, so we returned to using basic initial genotype.

```python
import random
def random_creature():
        parts = ["R", "C", "L", "M", "c", "RR", "MM"]
        num_parts = random.randint(3,10)
        neurons_position = random.sample(range(num_parts),random.randint(0,num_parts))
        neuron_count=0
        split_stack = []
        creature="("
        for j in range(num_parts):
         #chance to add split
         if random.randint(0,100)<30:
           creature+="("
           split_stack.append("(")
          if random.randint(0,100)<60:
           creature+=","*random.randint(1,2)
         #add modificators to part
         creature+= "".join(random.sample(parts, random.randint(1,3)))
         #add part
         creature+="X"*random.randint(1,3)
         #adding neuron
         neuron=""
         valid_places = list(range((-neuron_count),len(neurons_position)-neuron_count))
         if j in neurons_position and neuron_count!=len(valid_places):
          neuron_count+=1
          neuron +="["
          #rotation muscle or bending
          neuron += "".join(random.sample(["@", "|"], 1))
          #add to body
          for k in sorted(random.sample(valid_places, random.randint(1,len(valid_places)))):
           neuron +=f"{k}:{random.randint(0,10000)/1000},"
          neuron=neuron[:-1] #legacy code
          neuron+="]"
         creature+=neuron
         #chance to close split
         if len(split_stack)!=0 and random.randint(0,100) <20:
          creature+="),"
          split_stack = split_stack[:-1]
         #adding splits
         for _ in range(len(split_stack)+1):
           creature+=")"
        return creature
```

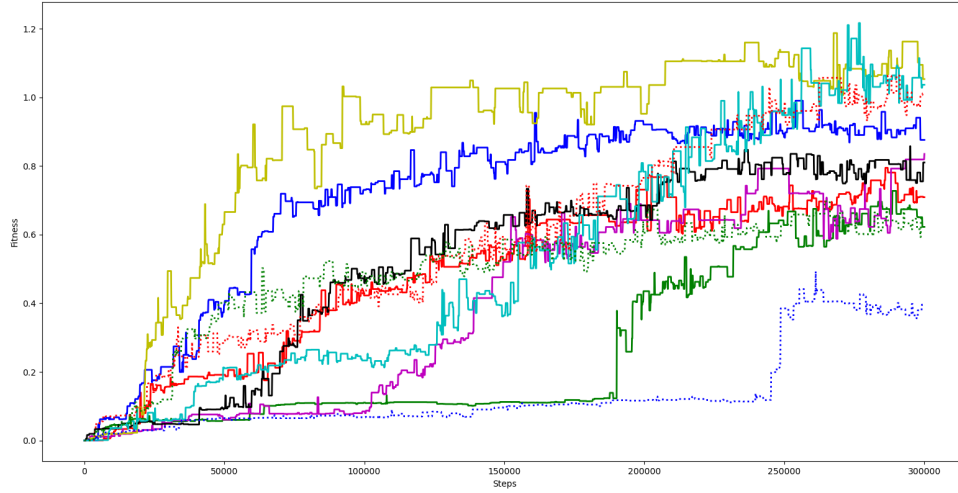By experimenting we got the following results:

3.1 Default

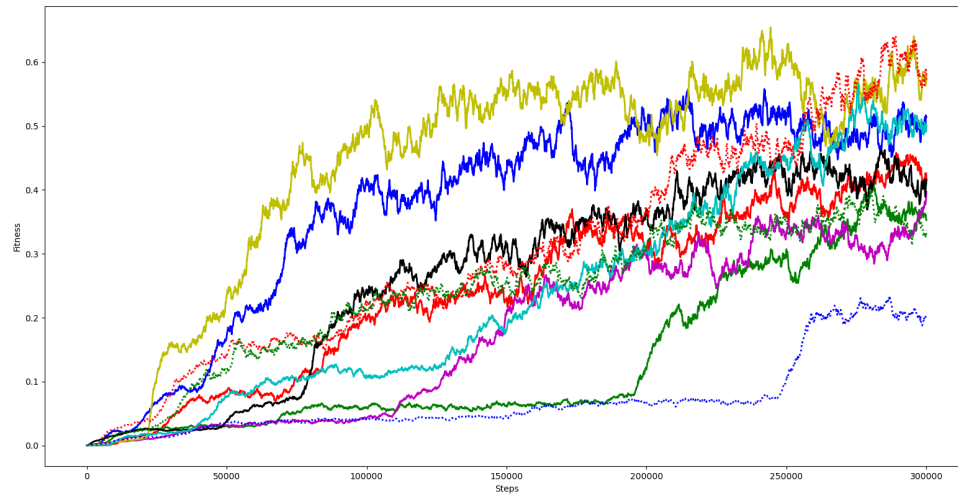Figure 3: Maxes plot of 10 runs for the default params.

Figure 4: Averages plot of 10 runs for the default params.
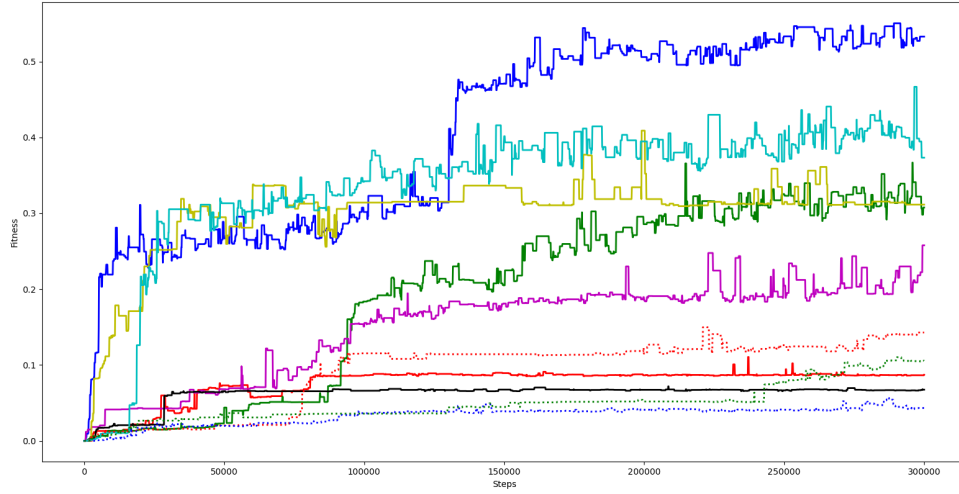
3.2 Modification 1

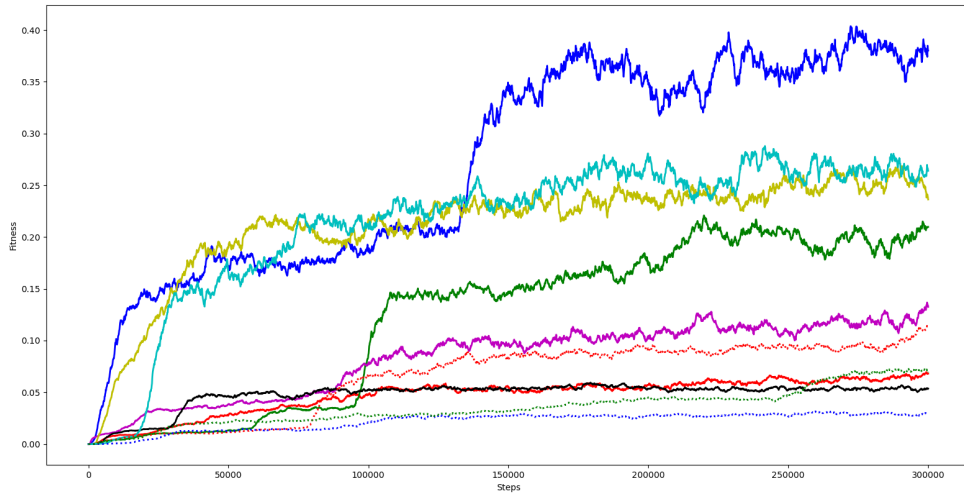Figure 5: Maxes plot of 10 runs for modification 1.

Figure 6: Averages plot of 10 runs for modification 1.
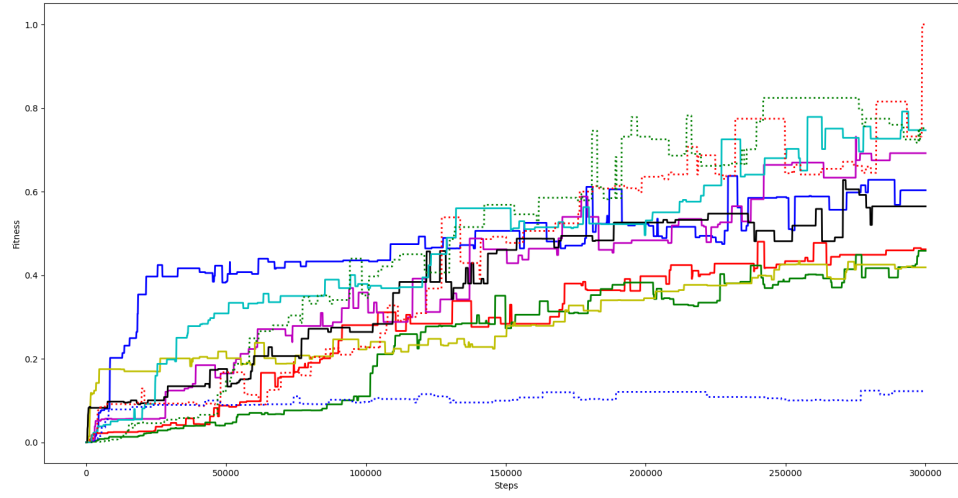
3.3   Modification 2

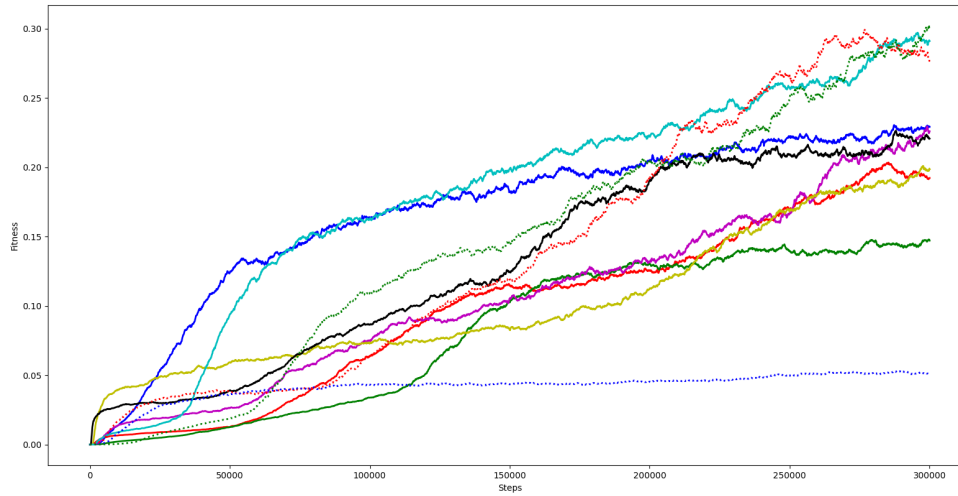Figure 7: Maxes plot of 10 runs for modification 2.

Figure 8: Averages plot of 10 runs for modification 2.

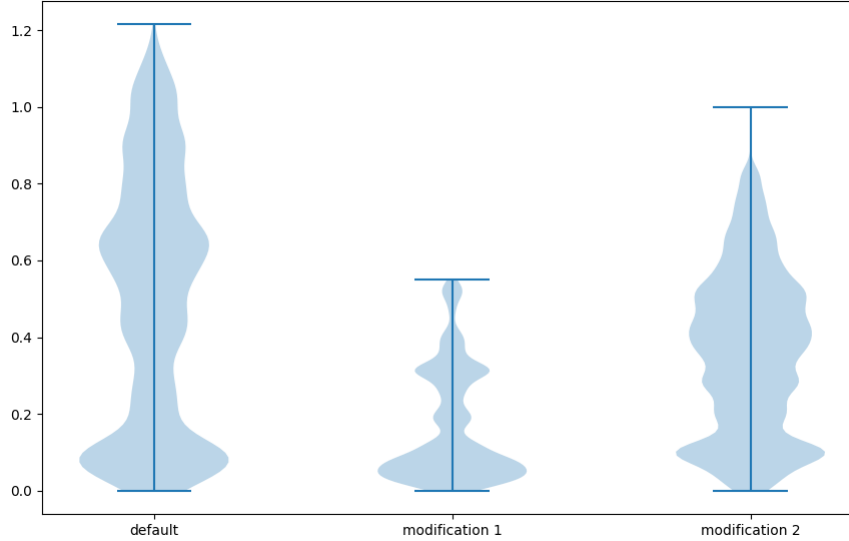Violin plot for comparing fitnesses of different parameter combinations:

Figure 9: Violin plot for the comparison of parameters

4 Quantitive conclusions

Based on the violin plot and fitness-in-time plots we can say that:

- There was a significant breakthrough in fitness at least once during each run (as visible on figure 3). It can be caused by successful mutation i.e. adding working bending muscle.

- The evolutionary goal is not very hard so the oftentimes average fitness has steady, linear growth (as visible on figure 8).

- This is not a surprise that sometimes invalid genotypes occur or that the minimal fitness drops to near-zero values because not all mutations and crossovers are perfect. Bad ones can totally destroy working solutions and produce such outputs.

- There is room for improvement, the experiment can be run longer and produce much more interesting solutions. The chosen stopping criterion seems to be too strict.

- There are many short plateaus, places with no improvement, in each run but our default parameters seem to cope well with those local optima (as visible on figure 5).

- Modification 1 sometimes gets stuck at local optima and cannot break through and performs much worse than default parameters (as visible on figure 5).

- Modification 2 is a simple gene pool increase but the fitness drops compared to the initial gene pool setting (as visible on figure 7).

5   Qualitative conclusions

What is the behavior of the best creatures?
The best creatures try to use jump as a moving factor. They tend to jump at 20-30 degrees to travel as much distance as possible.

Do they have some special genotypes?
The best creatures are minimalistic, avoiding unnecessary mass. Usually, they consist of 3-5 parts and 2-3 muscles forming a stick that can bend and rotate in 2/3 its length to break out of the ground.

Are the results in line with your expectations? If not, why?
Yes, the results are not exactly in line with our expectations, especially since we thought that jumping at 45 degrees is optimal from a mathematical point of view.

Most representative creatures (uploaded on YouTube, click on its name to open hyperlink):

- Framstick 1

- Framstick 2

- Framstick 3

List of Figures