Genetic Programming in Swift for Human-competitive Evolution

Petr Mánek, supervisor: František Mráz
Faculty of Mathematics and Physics, Charles University in Prague

Objectives

1. Implement a genetic programming library in the Swift programming language.
2. Demonstrate the usage of the library by applying it to sample problems.

Genetic Algorithms

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Swift Programming Language

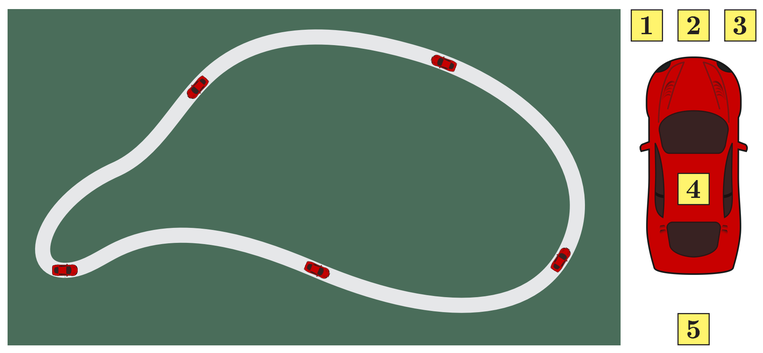
The Swift programming language has been unveiled in 2014 by the Apple Corporation. Since then, it has been widely adopted by software developers and computer engineers, succeeding Objective-C as the main programming language used for application development on the Apple mobile device platform. Building on proven coding paradigms, such as generics and strongly-typed objects, Swift strives to be a modern, concise and safe alternative to popular languages like Python or C++ while attempting to maintain comparable performance in terms of computational speed and memory management.

Architecture

Properties

Self-driving Car Simulation

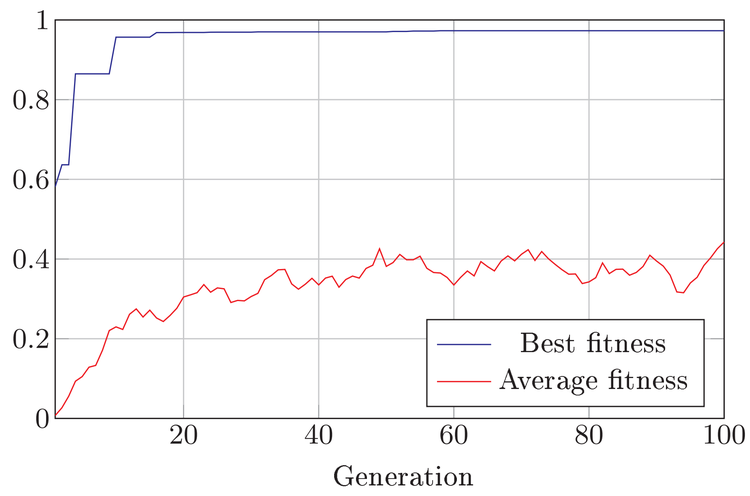
The presented library was used to evolve a control program for a self-driving car. The environment assumed Newtonian physics model (ignoring friction forces) and the parameters of the experimental car were modelled to match a real vehicle. The road was a randomly generated closed Bézier curve, which was detectable by 5 sensors positioned in the front, middle and the back of the car (see the picture below).



The car was controlled by a 3-layer feedforward neural network with 10 neurons in the hidden layer, whose connection weights were encoded in the genotype. Every program was evaluated in 5 independent 1-hour simulations, which tracked the total distance driven over the road. The fitness function was defined as

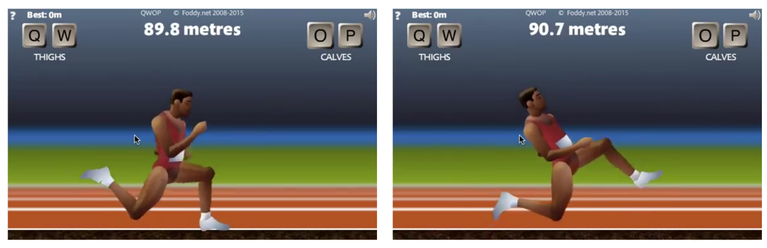
$$f(\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_5) = \frac{1}{5d_{max}} \sum_{i=1}^{5} \hat{d}_i$$

where $\hat{d}_1, \hat{d}_2, \ldots, \hat{d}_5$ denote the total distances driven over the road in the individual simulations and d_{max} denotes the maximum achievable distance given the simulation parameters. The best control program after 100 generations was able to stay on the road in approximately 70% of simulations.



QWOP Player

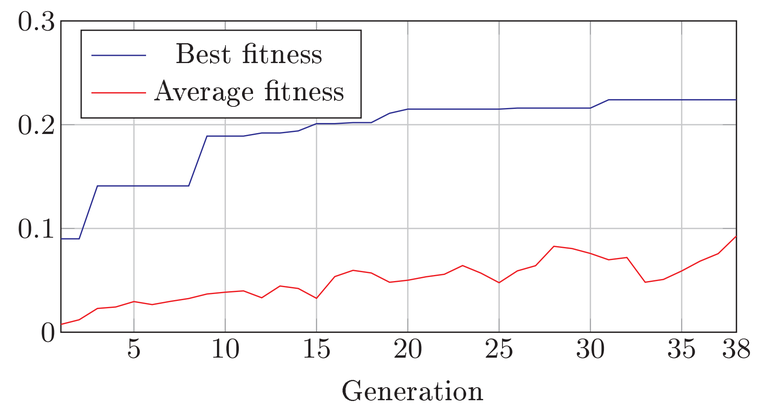
QWOP is a popular online game, in which the player drives an athlete to finish a 100-meter sprint race as fast as possible. QWOP's difficulty is caused by its control scheme, which only allows the player to move the athlete by contracting individual muscle groups within his body. The challenge of the game is in that sense comparable to the problem of evolving bipedal gaits in physical robots.



The presented library was used to evolve an artificial QWOP player and partially replicate human-competitive results achieved by **TODO**. In every generation, 80 game strategies were generated and encoded as simple programs (genotype strings), then evaluated by the fitness function

$$f(d_1, d_2, \ldots, d_n) = \frac{1}{100n} \sum_{i=1}^{n} d_i$$

where $d_1, d_2, \ldots, d_n$ denote the distances achieved in n trial 30-second runs. The best strategy after 38 generations was able to complete the race in approximately 152 seconds.



Conclusions

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Donec odio elit, dictum in, hendrerit sit amet, egestas sed, leo. Praesent feugiat sapien aliquet odio. Integer vitae justo. Aliquam vestibulum fringilla lorem. Sed neque lectus, consectetuer at, consectetuer sed, eleifend ac, lectus. Nulla facilisi. Pellentesque eget lectus. Proin eu metus. Sed porttitor. In hac habitasse platea dictumst. Suspendisse eu lectus. Ut mi mi, lacinia sit amet, placerat et, mollis vitae, dui. Sed ante tellus, tristique ut, iaculis eu, malesuada ac, dui. Mauris nibh leo, facilisis non, adipiscing quis, ultrices a, dui.

References

Morbi luctus, wisi viverra faucibus pretium, nibh est placerat odio, nec commodo wisi enim eget quam. Quisque libero justo, consectetuer a, feugiat vitae, porttitor eu, libero. Suspendisse sed mauris vitae elit sollicitudin malesuada. Maecenas ultricies eros sit amet ante. Ut venenatis velit. Maecenas sed mi eget dui varius euismod. Phasellus aliquet volutpat odio. Vestibulum ante ipsum primis in fau-