3D Snake Locomotion and Food Sensing using Neural Networks

Khyati Pawde
704590958
khyatipawde@cs.ucla.edu

Mansi Shah
504591572
mansishah@cs.ucla.edu

Sakshi Goplani
504562678
sakshigoplani@cs.ucla.edu

Abstract:* *In this paper we present a neural network method to allow an artificially evolved snake to survive in a randomly generated environment with the ability to locomote and sense food. The setting is based on a very popular mobile game called 'Snake'. We have implemented the snake genotype using neural networks simulated using the software-Framsticks that helps bring the virtual snake to life with movement and behavior resembling that of a real snake.

Keywords: Genotype, Framsticks, Neural Networks, Artificial Life, Snake

1. Introduction:

Artificial life [1] aims to analyze and create virtual creatures by studying real biological organisms. The best means of understanding artificial life forms for humans is through intelligible visualizations. Making use of Framsticks for the purpose of simulating the genotype of a snake serves this purpose. Framsticks is a software that is used for open-ended evolution of artificial life creatures and thus avoids limitations for the simulation of a neural network that supports different topologies and dimensions. The virtual creatures developed using Framsticks have "brains" and "bodies" that interact with each other and the environment. Thus Framsticks supports development and simulation of life-like sophisticated creatures and their phenomena. The paper is organized to describe the simulation model, evolution, the neural network for a slithering snake in both land and water environments.

2. Structure and Neural Network:

The rules for natural behavior that emerge from simulated creature's way of living are already

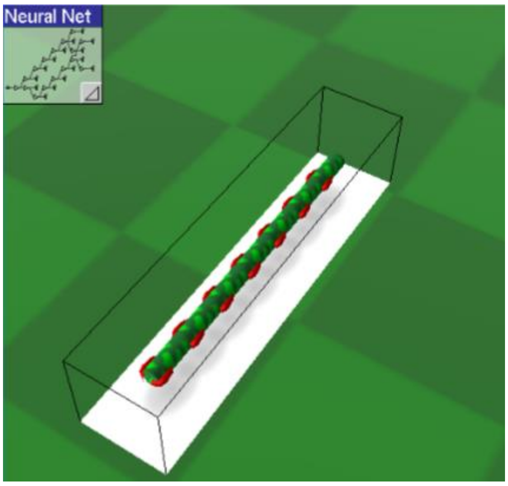

Figure 1: Original Snake Structure

made available in the Framsticks [1] simulation environment. The evolution simulator acts as the main simulator and maintains records of all the simulated genotypes.

A three-dimensional simulator is used to simulate and evaluate the genotypes. It supports various complex stimuli affecting the creature for dynamic development.

We simulated a snake genotype using the three-dimensional simulator initially in the land environment and later moved on to test its behavior for locomotion and orientation.

Figure 1 shows the original unevolved body of the snake's genotype simulated using Framsticks. The artificial life creature in Framsticks [2] mainly consists of the body and brain. As the name suggests the body of the genotypes are modelled using sticks as the basic element. This type of simulation is known to be Finite Element method. These sticks can vary in length, strength, weight, friction, etc. The brain is made up of Neurons and Connectors. The neuron interacts with the body or behaves specifically like a receptor or effector. These are placed on the stick to improve the sensing capabilities of the genotype. Muscles are capable of bending and rotating that help alter the relative positions of the adjacent sticks. Thus these aspects make the movement and behavior of our genotype seem realistic

```
X[Sin][-1:0.891,=:0.834][|-1:-1][|-2:1]X[|-2:1,!:01][|-2:1,!:.01]X[|-2:1,!:.01][|-2:1,!:.01]X[|-2:1,!:.01][|-2:0.887,!:.01]X[|-2:1,!:.01][|-2:-2.729,!:.01]X[|-1:3.077][|-3:1,1:-5.351][|-3:1,!:0.102]X[|-2:1,!:.01][|-2:1,!:.01]
```

Figure 2: Original Snake Genotype

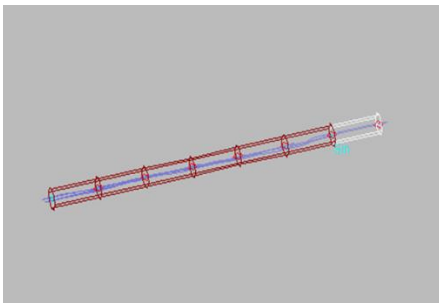

Figure 3: Original Snake Framsticks Structure

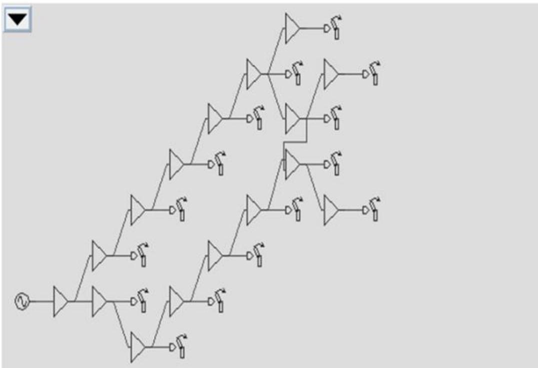

Figure 4: Original Snake Neural Network

The original genotype and the weights of the corresponding neural network can be seen in Figure 2 and 4. Each Framsticks element is denoted by an X followed by the syntax for specifying the type of muscle associated with the stick as well as the neural network connections and the weights.

In the syntax shown,

- | - represents a bending muscle
- = - represents the quantity of inertia in the muscle
- ! - specifies the force acting on the muscle
- Remaining (x:y) pairs - represent neural network connections where x is relative position of the neuron and y is the weight of the network edge.
- [Sin] - represents a sinusoidal wave generator which provides as stimulus for movement of the Snake

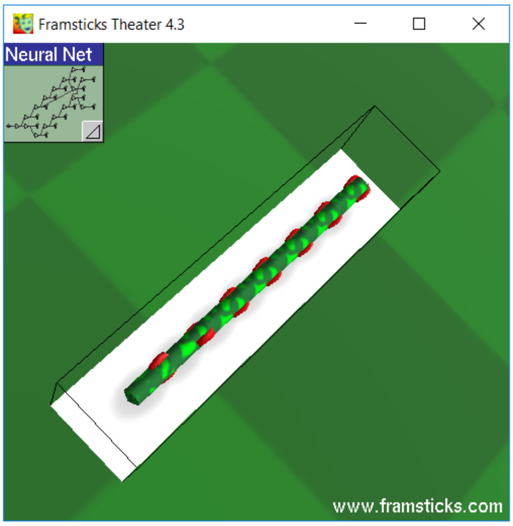

Figure 5: Evolved Snake Structure

Figure 5 shows the evolved body of the Snake's genotype from generation 17. Visually there is slight difference from that shown in Figure 1 except that the first two muscles are rotated and thus give a hood like appearance to the Snake which can be elevated in simulation.

```
X[Sin][-1:0.891, =:0.834][|-1:-1][|-2:1]X[|-2:1, !:01, s:0][|-2:1.588, !:.01]X[|-2:1, !:.01][|-2:1, !:.01]X[|-2:1.669, !:.01][|-2:1.512, !:.01]X[|-2:1, !:.01][|-2:-4.538, !:.01]RX[|-1:2.485][|-3:1, 1:-5.351][|-3:1, !:0.102, -9:1]RRX[|-2:1, !:.01][|-2:3.348,!:.01]
```

Figure 6: Evolved Snake Genotype

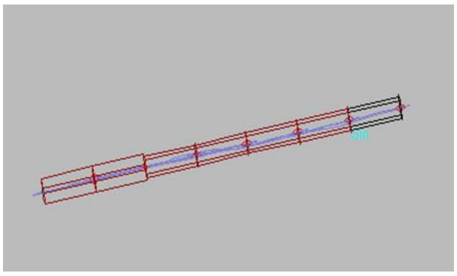

Figure 7: Evolved Snake Framsticks Structure

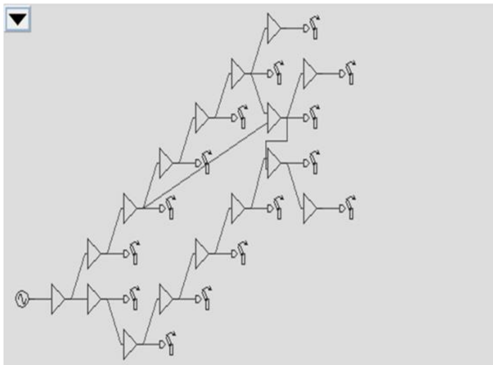

Figure 8: Evolved Snake Neural Network

The genotype of the same can be seen in Figure 6. The most significant changes worth noting is that there is a [s] sensor in one of the muscles. This helps sensing in the energy emanating from the environment and which in turn changes the neural network behaviour (refer Figure 8) to direct the Snake in direction of the source of energy (in this case, Food). Apart from that, certain weights and values are changed as a result of mutations over the generations.

3. Environment:

Initially, the artificial world of the snake consisted of a standard solid surface with a fence forming a boundary around it to constrict the snake to a stipulated space. The ground has a grid on it which makes it easy to detect the movement of the snake. The food is randomly dispersed on the surface, with new food appearing to replace the food that has previously been consumed by the snake so that a minimum threshold of food is always present on the surface. For the water snake, a similar controlled environment was developed, the only difference being that the surface had a water level (modified from -1 to 1 in the parameters). Thus, the friction automatically reduced and the snake was able to move more freely resulting in an increase velocity and realistic movement.

4. Experiment Definition:

The variables and rules for the simulator were defined using a script known as the experiment definition. This script helped define the behavior of the Framsticks system for the purpose of creating the objects in the simulated world, the interactions among these objects(snake and food), and the evolution. The evolution is a steady-state (one-at-a-time) selection model, where a single genotype is inserted into the gene pool at a time [2].

The "variables" defined in the script were used to initialize the parameters for the various contributing factors to the simulation i.e. creatures, environment, state and world. Some parameters for each of these include:

- **Creatures:** *name* (Snake), *energy* (25,000), death (0/1), *nnsim* (Neural Net Simulation) etc.
- **Experiment:** *initialgen* (genotype of initial generation), *maxcreated* (maximum creates that can be created and exist simultaneously), *rotation*, *creath* (creature height) etc.
- **State:** *totaltestedcr* (total number of creatures evaluated so far), *stepType* (normal simulation), *stepsCounter* (number of steps in current life cycle) etc.
- **World:** *worldtyp* (surface type, flat surface in our case), *worldbnd* (boundaries set), *wrldsiz* (size of our world) etc.

The "procedures" defined in the script were used to simulate life events [2]. The procedures and the events associated with them were as follows:

- **onExpDefLoad:**
  - Create a gene pool known as "Genotype".
  - Create 2 populations "Creatures" and "Food"
- **onExpInit:**
  - Empty all previous gene pools and populations.
  - Place the beginning genotype in "Genotype".
- **onStep:**
  - *Food:* If the amount of food is below the minimum threshold, then add more food to the environment.
  - *Creature:* If the genotype is dead due to exhaustion of it's energy, replace it with a new organism with a minimum reserve of energy.
- **onBorn:**
  - Move the food/creature to a randomly chosen place in the world.
  - The starting energy of the object is set according to it's type. (*Food:* 200 , *Snake:* 25,000)
- **onKill:**
  - If a creature has died, save it's performance in "Genotypes" which would possibly create a new genotype and cause evolution. The new generation would learn from the experiences of the previous generation and mutate accordingly.
- **onFoodCollision:**
  - Send energy from the "Food" to the "Creature"
- **on[X]Collision:**
  - On collision with a fence, turn around and move towards the nearest food source.

5. Simulation & Results:

For the simulation and evolution following steps were taken:

- **Simulation of the original snake**: Without any evolution, when simulated the original snake slithers in an amateur manner. Moreover its velocity is almost zero and it appears to be immobile. Also, without the ability to move or consume energy, it's lifespan is minimal

- **Evolution**: Due to poor performances of the original snake, we created our experiment definition to modify certain parameters which would help with evolution of the snake and it's genotype. We mainly defined the fitness value of the snake as a combination of its velocity and lifespan. As a result the future generations with higher fitness will have a higher velocity and larger lifespan.
  - Firstly we evolved till approximately 90-100th generation, however the creatures no longer resembled neither the body nor the movements of a snake even though they had a high velocity and were good at sensing and eating food.

  - Next, we evolved again but this time only till 20 generations to get a desirable trade off

  - between fitness and similarity to the ancestors. From these, we chose one particular snake model from generation 17 with the best fitness value.

- **Performance:** Next we simulated the original and evolved snake in a series of different environments (with or without water) and compared their performance. The simulations can be viewed in the video attached with the report.

- **Results:** Following were the prominent results observed in the simulation:
  - With or without water, the evolved snake is better in sensing, approaching and consuming food for gaining energy..
  - The original snake is almost unable to move or consume food on ground due to high friction acting between the body and the ground surface, and also due to lack of its ability to sense food as an energy source.
  - However, in water, even the slithering motion of the original snake appears most natural and effortless, the reason for which, we believe is the reduced resistance from friction. In our simulation, the same snake despite not being able to sense food, consumes one particle which came across in its path.
  - Another interesting observation in the water was that the evolved snake on being blocked by the boundary of the environment does not stop, but simply tries twice and eventually turns around and away from the boundary on its own. This was very interesting to observe because nowhere in the evolution was such a criteria specified and the snake simply adapted to the environment and learnt in real time.

6. Conclusion and Future Scope:

The main challenge in this project was to model a Snake whose locomotion techniques is starkly different from other species with limbs. We successfully managed to achieve a realistic model which can be effectively utilised for games and other animation applications.

Some interesting future work would be to simulate an environment with obstacles such as wall which the snake should learn to avoid. In such a case, the snake would have to move away from such obstacles to avoid losing energy/death and still find it's path towards the food. This is an interesting modification of the current scenario where the snake would go towards an object to gain energy.

7. References:

[1] Maciej Komosinski and Szymon Ulatowski." Framsticks: Towards a Simulation of a Nature-like World, Creatures and Evolution1" Institute of Computing Science, Poznan University of Technology

[2] Maciej Komosinski." The Framsticks system: versatile simulator of 3D agents and their evolution" Institute of Computing Science, Poznan University of Technology, Poznan, Poland