ISSN No:-2456-2165

Teaching AI to Play Games using Neuroevolution of Augmenting Topologies

Dr. T. Ramaswamy[1], Y. Pranati[2], C. V. Sindhu Lahari[3], S. Sanjana[4]
[1]Associate Professor, Dept. of ECE, Sreenidhi institute of Science and Technology, Telangana, India
[2]B.Tech(ECE),Sreenidhi institute of Science and Technology, Telangana, India
[3]B.Tech(ECE),Sreenidhi Institute of Science and Technology, Telangana, India
[4]B.Tech(ECE),Sreenidhi Institute of Science and Technology, Telangana, India

Abstract:- The paper is a demonstration of using the NEAT algorithm to allow artificial neural networks to evolve through a genetic algorithm to play the game of Flappy Bird. So it builds a most optimal network by itself by adding nodes or connections, or changing the weights of the connections, or enabling or disabling certain connections as and when required to accomplish the task at hand i.e the game of FlappyBird. We provide the inputs to the algorithm so as to create an optimal neural network that completes the task, in our case the game of Flappy bird.

Keywords:- *NEAT Algorithm, Artificial Neural Networks, Genetic Algorithm.*

I. INTRODUCTION

Genetic algorithms are numerical optimization algorithms inspired by both natural selection and natural genetics. The method is a general one, capable of being applied to an extremely wide range of problems. These algorithms are not complex and the code which is written applying the algorithms is simple. NEAT (NeuroEvolution of Augmenting Topologies) is a genetic algorithm that creates artificial neural networks by undergoing evolution.

II. BLOCK DIAGRAM

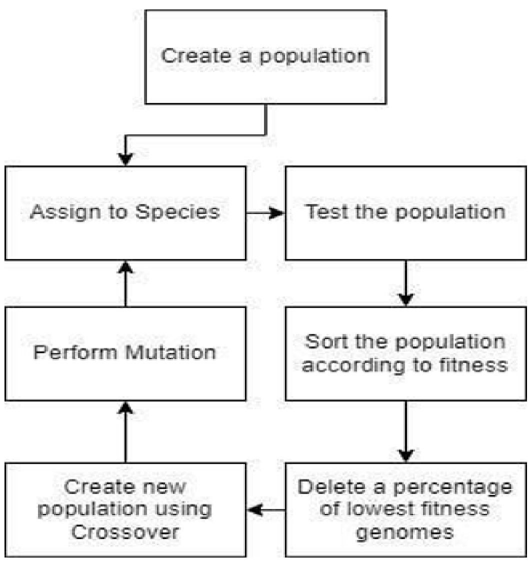

Fig 1 Block-Diagram for NEAT Algorithm

III. LITERATURE SURVEY

The optimization algorithm (or optimizer) is in charge of carrying out the learning process in a neural network. There are many different optimization algorithms. All have different characteristics and performance in terms of memory requirements, processing speed, and numerical precision.

Gradient Descent is the most basic but most used optimization algorithm. It's used heavily in linear regression and classification algorithms.

Backpropagation in neural networks also uses a gradient descent algorithm. Backpropagation refers to a technique from calculus to calculate the derivative (e.g. the slope or the gradient) of the model error for specific model parameters, allowing model weights to be updated to move down the gradient. As such, the algorithm used to train neural networks is also often referred to as simply backpropagation.

IV. WORKING

In feed-forward neural networks, the number of input, output,and hidden nodes are fixed as well as the connections between the nodes(usually being fully connected i.e all nodes in one layer are connected to all the nodes in the other layer). We test the network and make changes to the weights of the connections to get the correct output or expected output.

In the current implementation of NEAT, a population of individual genomes is maintained. Each genome contains a specific number of input and output nodes and one bias node to ensure that more importance is given to certain features than others. To evolve a solution to a problem, the user must provide a fitness function that assigns a single real number as the fitness(performance) of a genome indicating the quality of that particular genome: better ability to solve the problem means a higher fitness score. The algorithm progresses through several generations, with each generation being produced by reproduction and mutation of the fittest genomes of the previous generation. The reproduction and mutation operations may add nodes and/or connections to genomes, so as the algorithm proceeds genomes (and the neural networks they produce) may become more and more complex.

V. TRAINING AND TEST ENVIRONMENT

We are training the neural network to play the game of FlappyBird. The aim of the game is that the bird must pass through the gaps in the pipes. The bird cannot touch the pipes and alsomust not go beyond the top and bottom boundaries of the testenvironment. If done so, the bird will be considered as "dead".

Every bird in the test environment has a parameter called fitness which is a measure of its performance. The fitness ofa bird is incremented by an amount 'x', when the bird is traversing in the region where the gaps in the pipes are present and when the bird crosses the pipe, the fitness is incremented by 10 times of that of 'x'. Hence as the numberof pipes the bird traverses through increases, the fitness also increases considerably.

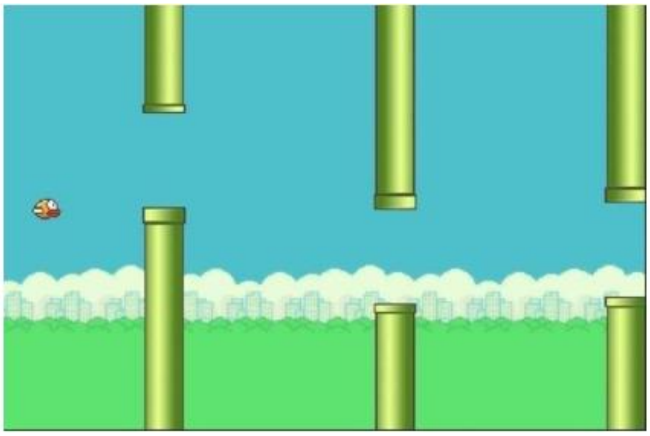

Fig 2 Test Environment (Flappy Bird)

Creating a population based on required population size. In this case we have a population size of 1000. The number of inputs that we provide to the genome(network) is 7 which include Bird's x position, Bird's y position, Bird's velocity, Top pipe's x position, Top pipe's y position, Bottom pipe's y position, and memory (it is the information regarding the previous action and parameters ofthe bird or genome).

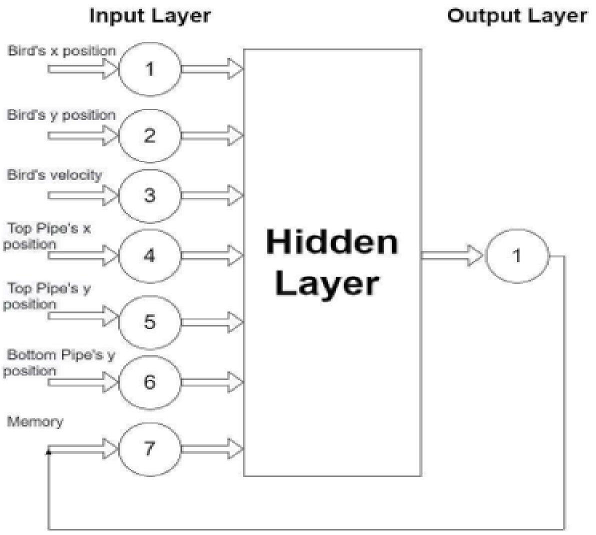

Fig 3 Network Structure

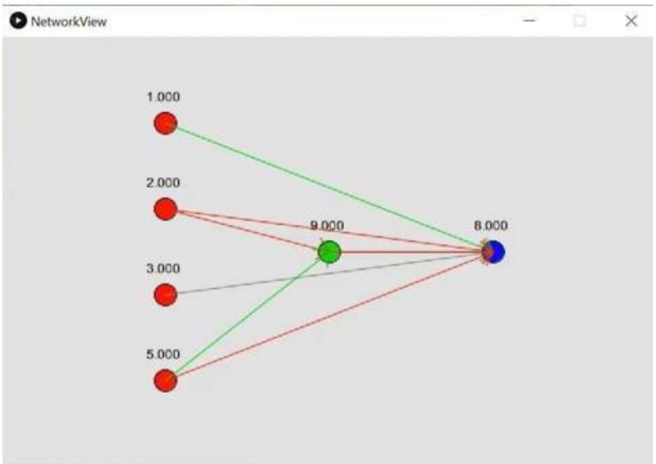

Fig 4 Network generated by the Algorithm after 40 Generations

The number of generations and their corresponding performance(fitness) has been plotted. We have also plotted the number of species present in each generation. By observing the Species vs Generation graph, we infer that as the number of generations increases, the species which havea higher average fitness are more likely to survive and in the process of creating new population, these species are more likely picked for reproduction. The offspring thus formed is also likely to be classified under the same species. So as the number of generations increase, the diversity in the population decreases.

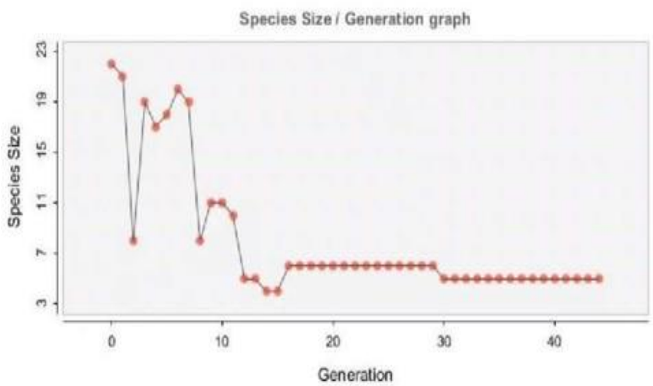

Fig 5 Species size / Generation graph

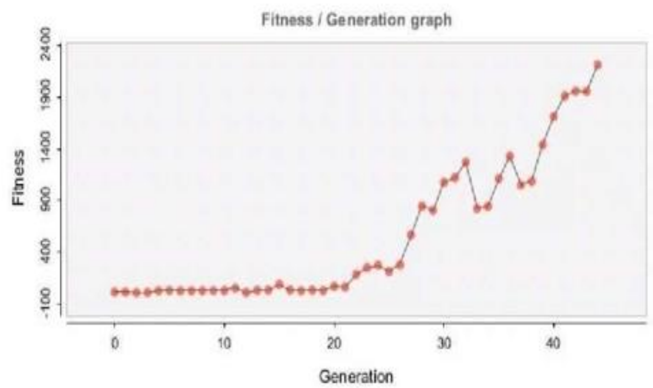

Fig 6 Fitness / Generation graph

VI. RESULT

Below is one of the highest performing networks generated by the algorithm, we observe that out of the total 7 inputs that we have initially provided, the network only used 4 inputs i.e., Bird's x position, Bird's y position, Bird's velocityand Top pipe's y position in order to successfully complete the game of Flappy Bird.

VII. FUTURE SCOPE

Genetic algorithm is based on natural selection and biological evolution. It is an excellent choice when it comes to finding an optimal solution for different kinds of problems. This algorithm has applications in fields of artificial intelligence and also in acoustics, data mining andmany more.

VIII. CONCLUSION

In this paper we have generated an optimal network to play the game of Flappy Bird but the scope of Genetic algorithms and NEAT is huge. The conclusion that we can draw is that several other complex tasks can be performed using NEAT algorithm

REFERENCES

1. Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning. *Machine Learning*, *3*(2), 95–99.
2. M. Wittkamp, L. Barone and P. Hingston, "Using NEAT for continuous adaptation and teamwork formation in Pacman," *2008 IEEE Symposium On Computational Intelligence and Games*, 2008, pp. 234-242, doi: 10.1109/CIG.2008.5035645.
3. L. Haldurai, T. Madhubala, R. Rajalakshmi, "A Study on Genetic Algorithm and its Applications," *International Journal of Computer Sciences and Engineering*, Vol.4, Issue.10, pp.139-143, 2016.
4. Kumar, Manoj and Husain, Mohammad and Upreti, Naveen and Gupta, Deepti, Genetic Algorithm: Review and Application (December 1, 2010).