

NEAT algorithm on 2D games

Name: Hitesh Chandravadiya

Dept. name: DEPSTAR(6CE1)

Org Name: Charusat University

Location: Upleta Gujarat

Email: 19dce016@charusat.edu.in

hiteshchandravadiya@gmail.com

Name: Jeel Devani

Dept. name: DEPSTAR(6CE1)

Org Name: Charusat University

Location: Surat, Gujarat

Email: 19dce025@charusat.edu.in

devanijeel3@gmail.com

Name: Daksh Jain

Dept. Name: DEPSTAR(6CE1)

Org Name: Charusat University

Location: Idar, Gujarat

Email: 19dce041@charusat.edu.in

Abstract

Neuroevolution (NE) is that the artificial evolution of neural networks victimisation genetic algorithms. In this project, we tend to explore the chances of the Neuroevolution of augmenting topologies (NEAT) algorithmic rule by victimisation it during a computer game. The game is concerning 2 robots during a 2nd plane learning to shoot one another and escape attacks at identical time. Experiments were performed for over fifty generations of Neural Evolution, the results of that shows sensible learning by the robots within the given setting.

Keywords: neuro evolution, neat, genes, genome

Introduction

The main good thing about the Neuro Evolution of Augmenting Topologies (NEAT) is that it decreases the magnitude of the search house of connecting weights.

The structure during this algorithmic rule is evolved such the topologies square measure reduced and fully grown incrementally which ends up in important gains in learning speed.

It is claimed in [1] that the hyperbolic potency is because of

employing a principled method of crossover of different topologies

protecting structural innovation using speciation.

incrementally growing from minimal structure.

The performance obtained and also the variation in fitness and structure parameters therefore shaped once evolving is being studied.

Previous Work

NEAT algorithmic rule isn't the primary algorithmic rule that evolve each neural network topologies and weight. TWEANNs is one such algorithmic rule involving.

They moon-faced many issues like competency prescript downside that was taken care of by NEAT victimisation the historical info concerning the genes.

Other issues like survival of the innovation once optimisation takes place over an extended variety

of generations is additionally taken care of by victimisation the conception of parallel evolution.

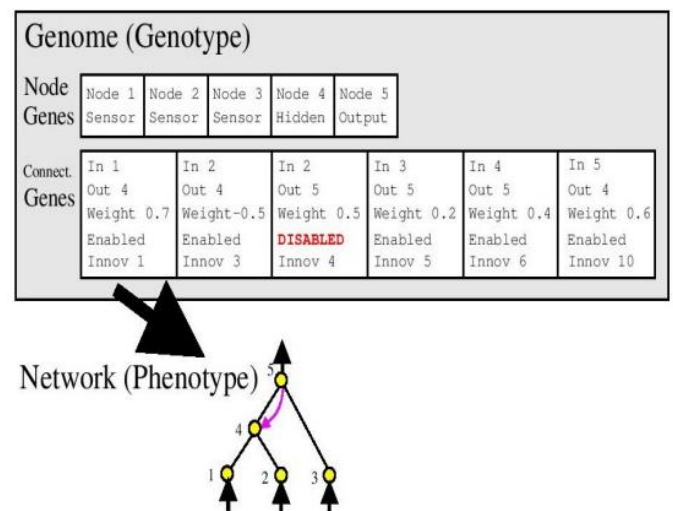
Methodology

overcome the drawbacks in TWEANNs as mentioned earlier, the NEAT algorithmic rule uses many innovative modifications. These square measure delineated very well within the following subsections as mentioned in [2].

Genetic Encoding:

Genomes square measure linear representations of network property. every genomic sequence includes an inventory of association genes, every of that refers to 2 node genes being connected.

Node genes give an inventory of inputs, hidden node, and outputs that may be connected. every association sequence specifies the in-node, the outnode, the load of the association, whether or not or not the association sequence is expressed (an alter bit), Associate in Nursing innovation variety, that permits finding corresponding genes.



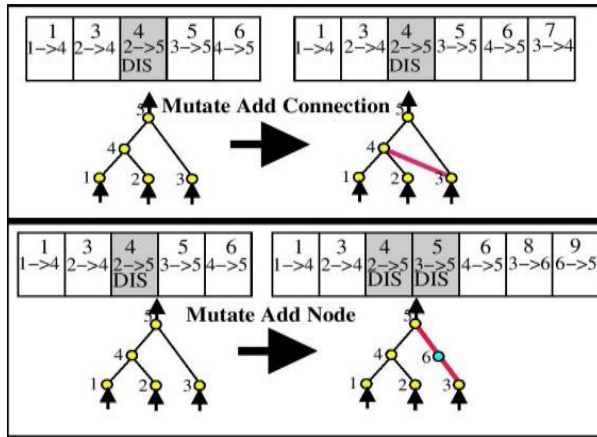
(Image from NEAT algorithm's official documentation)

The on top of figure demonstrates 2 reasonably structural mutation that takes place, that square measure add association and add node.

- In Associate in Nursing add association permutation, one new association sequence is intercalary connecting 2 recent unconnected nodes.
- Whereas in Associate in Nursing add node mutation, Associate in Nursing existing association is split and also the

new node placed wherever the recent association accustomed be.

Tracking Genes through Historical Markings:



(Image from NEAT algorithm's official documentation)

Origin of every sequence is caterpillar-tracked employing a creation variety to every sequence shaped, that represents the timeline of the arrival of the sequence within the system.

Solves the matter of competency conventions.

Protecting Innovation through Speciation:

Speciating the people could be a answer to 1 of the issues of the TWEANNs mentioned on top of.

It permits the entity to challenge between their own niches instead of the entire population at giant.

In this manner, population creation gets longer to optimize in their own niches and therefore innovations square measure protected victimisation biological evolution.

Dismissing range through gradual Growth from Minimal Structure:

TWEANNs usually begin with Associate in Nursing early population of random topologies to introduce diversity from the start whereas, NEAT biases the search towards smallest dimensional areas.

This is done by beginning out with a homogenous population of networks with zero hidden nodes and structural modifications square measure done bit by

bit of that solely those survive that square measure found helpful through fitness assessment.

When the population starts minimum, the spatial property of the search house is reduced, and NEAT is often rummaging through fewer dimensions than different TWEANNs and fixed-topology NE systems.

Datasets and algorithmic rule Used

Datasets and Algorithm Used

A 2nd region was shaped wherever each the birds(flappy bird) were allowed to maneuver around and withstand the pipes in any potential manner and obtain feedback from the Fitness perform thenceforth.

This feedback was used for the evolution of ordering for over fifty generation.

The inputs were the X and Y coordinate information of each the birds(flappy bird), and 2 outputs were taken that were fed to the fitness perform.

Code Used

The code is implemented in python library name is neat python.

Neat-Python examples:

<https://github.com/CodeReclaimers/neat-python>

Results

The results obtained square measure within the kind of videos, every golem activity some basic tasks(the main task was divided into many subtasks that helped in achieving the most goal).

Statistical information involves variation in Fitness worth and species size over generations.

Fitness analysis

The average and also the best fitness was evaluated over many generations as shown within the figure.

Where the typical fitness is found to be nearly at, the most effective fitness is found to possess short peaks.

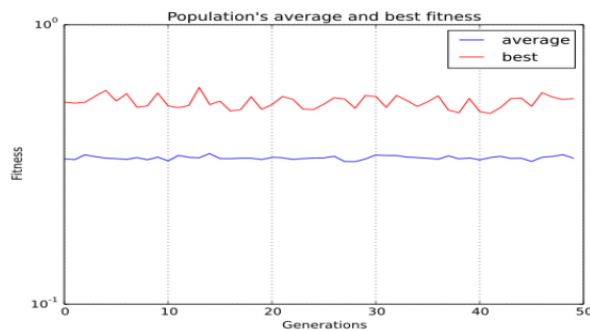
The seemingly reason being the actual fact that the minimum of the 2 fitnesses is taken because the overall fitness.

Fitness evaluation

The average and the best fitness was evaluated over several generations as shown in the figure.

Where the average fitness is found to be almost at, the best fitness is found to have short peaks.

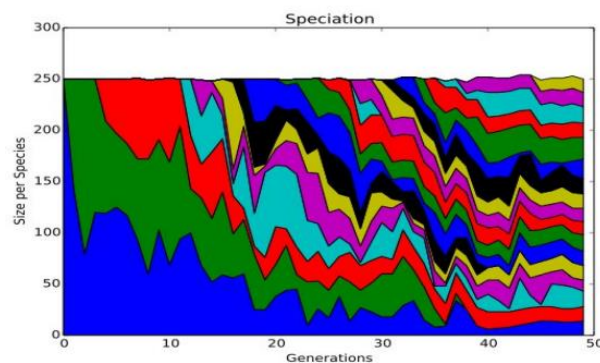
The likely reason being the fact that the minimum of the two fitnesses is taken as the overall fitness.



(Image from NEAT algorithm's official documentation)

Speciation

it started with one single species and whenever any mutation resulted in any structure that had a distance larger than some threshold from all the opposite current species, a replacement species is made. additional concerning such structure and threshold is delineated in [2].



(Image from NEAT algorithm's official documentation)

References

- [1] C. W. Anderson. Learning to control an inverted pendulum using neural networks. *IEEE Control Systems Magazine*, 9: 31–37, 1989.
- [2] P. J. Angeline, G. M. Saunders, and J. B. Pollack. An evolutionary algorithm that constructs recurrent neural networks. *IEEE Transactions on Neural Networks*, 5:54–65, 1993.
- [3] A. G. Barto, R. S. Sutton, and C. W. Anderson. Neuronlike adaptive elements that can solve difficult learning control problems. *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-13:834–846, 1983.
- [4] D. E. Goldberg and J. Richardson. Genetic algorithms with sharing for multimodal function optimization. In J. J. Grefenstette, editor, *Proceedings of the Second International Conference on Genetic Algorithms*, pages 148–154. San Francisco, CA: Morgan Kaufmann, 1987.
- [5] F. Gomez and R. Miikkulainen. Incremental evolution of complex general behavior. *Adaptive Behavior*, 5:317–342, 1997.
- [6] F. Gomez and R. Miikkulainen. Solving non-Markovian control tasks with neuroevolution. In *Proceedings of the 16th International Joint Conference on Artificial Intelligence*, Denver, CO, 1999. Morgan Kaufmann.
- [7] F. Gruau, D. Whitley, and L. Pyeatt. A comparison between cellular encoding and direct encoding for genetic neural networks. In John R. Koza, David E. Goldberg, David B. Fogel, and Rick L. Riolo, editors, *Genetic Programming 1996: Proceedings of the First Annual Conference*, pages 81–89, Cambridge, MA, 1996. MIT Press.
- [8] S. W. Mahfoud. *Niching Methods for Genetic Algorithms*. PhD thesis, U. of Illinois at Urbana-Champaign, May 1995.
- [9] D. Michie and R. A. Chambers. BOXES: An experiment in adaptive control. In E. Dale and D. Michie, editors, *Machine Intelligence*. Oliver and Boyd, Edinburgh, UK, 1968.
- [10] D. E. Moriarty. *Symbiotic Evolution of Neural Networks in Sequential Decision Tasks*. PhD thesis, Department of Computer Sciences, The University of Texas at Austin, 1997. Technical Report UT-AI97-257.
- [11] D. E. Moriarty and Risto Miikkulainen. Efficient reinforcement learning through symbiotic evolution. *Machine Learning*, 22:11–32, 1996.
- [12] M. A. Potter and K. A. De Jong. Evolving neural networks with collaborative species. In *Proceedings of the 1995 Summer Computer Simulation Conference*, 1995.
- [13] J. C. F. Pujol and R. Poli. Evolving the topology and the weights of neural networks using a dual representation. *Special Issue on*

Evolutionary Learning of the Applied Intelligence Journal, 8(1):73–84, January 1998.

[14] N. J Radcliffe. Genetic set recombination and its application to neural network topology optimisation. Neural computing and applications, 1(1):67–90, 1993.

[15] N. Saravanan and D. B. Fogel. Evolving neural control systems. IEEE Expert, pages 23–27, June 1995.

[16] K. Stanley and R. Miikkulainen. Evolving neural networks through augmenting topologies. Tech. Report AI2001-290, Dept. of Computer Sciences, The U. of Texas at Austin, 2001.

[17] C. J. C. H. Watkins and P. Dayan. Q-learning. Machine Learning, 8(3):279–292, 1992.

[18] D. Whitley, S. Dominic, R. Das, and C. W. Anderson. Genetic reinforcement learning for neurocontrol problems. Machine Learning, 13:259–284, 1993.

[19] A. P. Wieland. Evolving neural network controllers for unstable systems. In Proceedings of the International Joint Conference on Neural Networks (Seattle, WA), pages 667–673. Piscataway, NJ: IEEE, 1991.

[20] A. P. Wieland. Evolving controls for unstable systems. In David S. Touretzky, Jeffrey L. Elman, Terrence J. Sejnowski, and Geoffrey E. Hinton, editors, Connectionist Models: Proceedings of the 1990 Summer School, pages 91–102. San Francisco, CA: Morgan Kaufmann, 1990.

[21] X. Yao. Evolving artificial neural networks. Proceedings of the IEEE, 87(9):1423–1447, 1999.

[22] B. Zhang and H. Muhlenbein. Evolving optimal neural networks using genetic algorithms with Occam’s razor. Complex Systems, 7:199–220, 1993.