WIKIPEDIA

Neuroevolution

Neuroevolution, or **neuro-evolution**, is a form of artificial intelligence that uses evolutionary algorithms to generate artificial neural networks (ANN), parameters, topology and rules.[1] It is most commonly applied in artificial life, general game playing[2] and evolutionary robotics. The main benefit is that neuroevolution can be applied more widely than supervised learning algorithms, which require a syllabus of correct input-output pairs. In contrast, neuroevolution requires only a measure of a network's performance at a task. For example, the outcome of a game (i.e. whether one player won or lost) can be easily measured without providing labeled examples of desired strategies. Neuroevolution can be contrasted with conventional deep learning techniques that use gradient descent on a neural network with a fixed topology.

Contents

Features

Comparison with gradient descent

Direct and indirect encoding

Taxonomy of embryogenic systems for indirect encoding

Examples

See also

References

External links

Features

Many neuroevolution algorithms have been defined. One common distinction is between algorithms that evolve only the strength of the connection weights for a fixed network topology (sometimes called conventional neuroevolution), as opposed to those that evolve both the topology of the network and its weights (called TWEANNs, for Topology and Weight Evolving Artificial Neural Network algorithms).

A separate distinction can be made between methods that evolve the structure of ANNs in parallel to its parameters (those applying standard evolutionary algorithms) and those that develop them separately (through memetic algorithms).[3]

Comparison with gradient descent

Most neural networks use gradient descent rather than neuroevolution. However, around 2017 researchers at Uber stated they had found that simple structural neuroevolution algorithms were competitive with sophisticated modern industry-standard gradient-descent deep learning algorithms, in part because neuroevolution was found to be less likely to get stuck in dead ends. In *Science*, journalist Matthew Hutson speculated that part of the reason neuroevolution is succeeding where it had failed before is due to the increased computational power available in the 2010s.[4]

Direct and indirect encoding

Evolutionary algorithms operate on a population of genotypes (also referred to as genomes). In neuroevolution, a genotype is mapped to a neural network phenotype that is evaluated on some task to derive its fitness.

In *direct* encoding schemes the genotype directly maps to the phenotype. That is, every neuron and connection in the neural network is specified directly and explicitly in the genotype. In contrast, in *indirect* encoding schemes the genotype specifies indirectly how that network should be generated.[5]

Indirect encodings are often used to achieve several aims:[5][6][7][8][9]

- modularity and other regularities;
- compression of phenotype to a smaller genotype, providing a smaller search space;
- mapping the search space (genome) to the problem domain.

Taxonomy of embryogenic systems for indirect encoding

Traditionally indirect encodings that employ artificial embryogeny (also known as artificial development) have been categorised along the lines of a *grammatical approach* versus a *cell chemistry approach*.[10] The former evolves sets of rules in the form of grammatical rewrite systems. The latter attempts to mimic how physical structures emerge in biology through gene expression. Indirect encoding systems often use aspects of both approaches.

Stanley and Miikkulainen[10] propose a taxonomy for embryogenic systems that is intended to reflect their underlying properties. The taxonomy identifies five continuous dimensions, along which any embryogenic system can be placed:

- **Cell (neuron) fate**: the final characteristics and role of the cell in the mature phenotype. This dimension counts the number of methods used for determining the fate of a cell.
- **Targeting**: the method by which connections are directed from source cells to target cells. This ranges from specific targeting (source and target are explicitly identified) to relative targeting (e.g. based on locations of cells relative to each other).
- **Heterochrony**: the timing and ordering of events during embryogeny. Counts the number of mechanisms for changing the timing of events.
- **Canalization**: how tolerant the genome is to mutations (brittleness). Ranges from requiring precise genotypic instructions to a high tolerance of imprecise mutation.
- **Complexification**: the ability of the system (including evolutionary algorithm and genotype to phenotype mapping) to allow complexification of the genome (and hence phenotype) over time. Ranges from allowing only fixed-size genomes to allowing highly variable length genomes.

Examples

Examples of neuroevolution methods (those with direct encodings are necessarily non-embryogenic):

Method	Encoding	Evolutionary algorithm	Aspects evolved
Neuro-genetic evolution by E. Ronald, 1994[11]	Direct	Genetic algorithm	Network Weights
Cellular Encoding (CE) by F. Gruau, 1994[7]	Indirect, embryogenic (grammar tree using S-expressions)	Genetic programming	Structure and parameters (simultaneous, complexification)
GNARL by Angeline et al., 1994[12]	Direct	Evolutionary programming	Structure and parameters (simultaneous, complexification)
EPNet by Yao and Liu, 1997[13]	Direct	Evolutionary programming (combined with backpropagation and simulated annealing)	Structure and parameters (mixed, complexification and simplification)
NeuroEvolution of Augmenting Topologies (NEAT) by Stanley and Miikkulainen, 2002[14][15]	Direct	Genetic algorithm. Tracks genes with historical markings to allow crossover between different topologies, protects innovation via speciation.	Structure and parameters
Hypercube-based NeuroEvolution of Augmenting Topologies (HyperNEAT) by Stanley, D'Ambrosio, Gauci, 2008[6]	Indirect, non-embryogenic (spatial patterns generated by a Compositional pattern-producing network (CPPN) within a hypercube are interpreted as connectivity patterns in a lower-dimensional space)	Genetic algorithm. The NEAT algorithm (above) is used to evolve the CPPN.	Parameters, structure fixed (functionally fully connected)
Evolvable Substrate Hypercube-based NeuroEvolution of Augmenting Topologies (ES-HyperNEAT) by Risi, Stanley 2012[9]	Indirect, non-embryogenic (spatial patterns generated by a Compositional pattern-producing network (CPPN) within a hypercube are interpreted as connectivity patterns in a lower-dimensional space)	Genetic algorithm. The NEAT algorithm (above) is used to evolve the CPPN.	Parameters and network structure
Evolutionary Acquisition of Neural Topologies (EANT/EANT2) by Kassahun and Sommer, 2005[16] / Siebel and Sommer, 2007[17]	Direct and indirect, potentially embryogenic (Common Genetic Encoding[5])	Evolutionary programming/Evolution strategies	Structure and parameters (separately, complexification)
Interactively Constrained Neuro-Evolution (ICONE) by	Direct, includes constraint masks to restrict the search to specific topology / parameter manifolds.	Evolutionary algorithm. Uses constraint masks to drastically reduce the search space through exploiting domain knowledge.	Structure and parameters (separately, complexification, interactive)

Rempis, 2012[18]			
Deus Ex Neural Network (DXNN) by Gene Sher, 2012[19]	Direct/Indirect, includes constraints, local tuning, and allows for evolution to integrate new sensors and actuators.	Memetic algorithm. Evolves network structure and parameters on different time-scales.	Structure and parameters (separately, complexification, interactive)
Spectrum-diverse Unified Neuroevolution Architecture (SUNA) by Danilo Vasconcellos Vargas, Junichi Murata[20] (Download code)	Direct, introduces the Unified Neural Representation (representation integrating most of the neural network features from the literature).	Genetic Algorithm with a diversity preserving mechanism called Spectrum-diversity that scales well with chromosome size, is problem independent and focus more on obtaining diversity of high level behaviours/approaches. To achieve this diversity the concept of chromosome Spectrum is introduced and used together with a Novelty Map Population	Structure and parameters (mixed, complexification and simplification)
Modular Agent-Based Evolver (MABE) by Clifford Bohm, Arend Hintze, and others.[21] (Download code)	Direct or indirect encoding of Markov Brains, Neural Networks, genetic programming, and other arbitrarily customizable controllers.	Provides evolutionary algorithms, genetic programming algorithms, and allows customized algorithms, along with specification of arbitrary constraints.	Evolvable aspects include the neural model and allows for the evolution of morphology and sexual selection among others.
Covariance Matrix Adaptation with Hypervolume Sorted Adaptive Grid Algorithm (CMA-HAGA) by Shahin Rostami, and others.,[22][23]	Direct, includes an atavism feature which enables traits to disappear and re-appear at different generations.	Multi-Objective Evolution Strategy with Preference Articulation	Structure, weights, and biases.

See also

- Automated machine learning (AutoML)
- Evolutionary computation
- Covariance Matrix Adaptation with Hypervolume Sorted Adaptive Grid Algorithm (CMA-HAGA)
- NeuroEvolution of Augmented Topologies (NEAT)
- Noogenesis
- HyperNEAT (A Generative version of NEAT)
- ES-HyperNEAT (A Generative version of NEAT that determines parameters and network structure)
- Evolutionary Acquisition of Neural Topologies (EANT/EANT2)
- Spectrum-diverse Unified Neuroevolution Architecture (SUNA)

References

1. Stanley, Kenneth O. (2017-07-13). "Neuroevolution: A different kind of deep learning" (https://www.oreilly.com/ideas/neuroevolution-a-different-kind-of-deep-learning). *O'Reilly Media*. Retrieved 2017-09-04.
2. Risi, Sebastian; Togelius, Julian (2017). "Neuroevolution in Games: State of the Art and Open Challenges" (https://arxiv.org/pdf/1410.7326v3.pdf) (PDF). *IEEE Transactions on Computational Intelligence and AI in Games*.
3. Togelius, Julian; Schaul, Tom; Schmidhuber, Jurgen; Gomez, Faustino (2008), "Countering poisonous inputs with memetic neuroevolution" (https://www.academia.edu/download/30945872/poison.pdf) (PDF), *Parallel Problem Solving from Nature*

4. "Artificial intelligence can 'evolve' to solve problems" (http://www.sciencemag.org/news/2018/01/artificial-intelligence-can-evolve-solve-problems). *Science | AAAS*. 10 January 2018. Retrieved 7 February 2018.
5. Kassahun, Yohannes; Sommer, Gerald; Edgington, Mark; Metzen, Jan Hendrik; Kirchner, Frank (2007), "Common genetic encoding for both direct and indirect encodings of networks" (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.159.705), *Genetic and Evolutionary Computation Conference*, ACM Press, pp. 1029–1036
6. Gauci, Stanley (2007), "Generating Large-Scale Neural Networks Through Discovering Geometric Regularities" (http://eplex.cs.ucf.edu/papers/gauci_gecco07.pdf) (PDF), *Genetic and Evolutionary Computation Conference*, New York, NY: ACM
7. Gruau, Frédéric; I, L'universite Claude Bernard-lyon; Doctorat, Of A. Diplome De; Demongeot, M. Jacques; Cosnard, Examinators M. Michel; Mazoyer, M. Jacques; Peretto, M. Pierre; Whitley, M. Darell (1994). *Neural Network Synthesis Using Cellular Encoding And The Genetic Algorithm* (http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.29.5939).
8. Clune, J.; Stanley, Kenneth O.; Pennock, R. T.; Ofria, C. (June 2011). "On the Performance of Indirect Encoding Across the Continuum of Regularity" (http://ieeexplore.ieee.org:80/document/5910671/?reload=true). *IEEE Transactions on Evolutionary Computation*. **15** (3): 346–367. doi:10.1109/TEVC.2010.2104157 (https://doi.org/10.1109%2FTEVC.2010.2104157). ISSN 1089-778X (https://www.worldcat.org/issn/1089-778X).
9. Risi, Sebastian; Stanley, Kenneth O. (2012). "An Enhanced Hypercube-Based Encoding for Evolving the Placement, Density and Connectivity of Neurons" (http://eplex.cs.ucf.edu/papers/risi_alife12.pdf) (PDF). *Artificial Life journal*. Cambridge, Massachusetts: MIT Press.
10. Stanley, Kenneth O.; Miikkulainen, Risto (2003). "A Taxonomy for Artificial Embryogeny" (http://nn.cs.utexas.edu/downloads/papers/stanley.alife03.pdf) (PDF). *The MIT Press Journals*.
11. Ronald, Edmund; Schoenauer, March, "Genetic Lander: An experiment in accurate neuro-genetic control" (http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.56.3139), *PPSN III 1994 Parallel Programming Solving from Nature*
12. Angeline, Peter J.; Saunders, Gregory M.; Pollack, Jordan B. (1994). "An evolutionary algorithm that constructs recurrent neural networks" (http://demo.cs.brandeis.edu/papers/ieeenn.pdf) (PDF). *IEEE Transactions on Neural Networks*. (5): 54–65.
13. Yao, Xin; Liu, Yong (May 1997). "A new evolutionary system for evolving artificial neural networks" (http://www.cs.bham.ac.uk/~axk/evoNN2.pdf) (PDF). *IEEE Transactions on Neural Networks*. **8** (3): 694–713.
14. Stanley, Kenneth O.; Bryant, Bobby D.; Miikkulainen, Risto (December 2005). "Real-Time Neuroevolution in the NERO Video Game" (http://nn.cs.utexas.edu/downloads/papers/stanley.ieeetec05.pdf) (PDF).
15. Stanley, Kenneth O.; Miikkulainen, Risto (2002). "Evolving Neural Networks through Augmenting Topologies" (http://nn.cs.utexas.edu/downloads/papers/stanley.ec02.pdf) (PDF). *Evolutionary Computation*. MIT Press. **10** (2): : 99–127.
16. Kassahun, Yohannes; Sommer, Gerald (April 2005), "Efficient reinforcement learning through evolutionary acquisition of neural topologies" (http://www.ks.informatik.uni-kiel.de/~yk/ESANN2005EANT.pdf) (PDF), *13th European Symposium on Artificial Neural Networks*, Bruges, Belgium, pp. 259–266
17. Siebel, Nils T.; Sommer, Gerald (October 2007). "Evolutionary reinforcement learning of artificial neural networks" (https://web.archive.org/web/20080905102111/http://www.ks.informatik.uni-kiel.de/~vision/doc/Publications/nts/SiebelSommer-IJHIS2007.pdf) (PDF). *International Journal of Hybrid Intelligent Systems*. **4** (3): 171–183. Archived from the original (http://www.ks.informatik.uni-kiel.de/~vision/doc/Publications/nts/SiebelSommer-IJHIS2007.pdf) (PDF) on 2008-09-05.
18. Rempis,, Christian W. (October 2012). "Evolving Complex Neuro-Controllers with Interactively Constrained Neuro-Evolution" (http://repositorium.uni-osnabrueck.de/handle/urn:nbn:de:gbv:700-2012101710370) (PhD thesis). Osnabrück University. urn:nbn:de:gbv:700-2012101710370
19. Sher, Gene I. (November 2012). *Handbook of Neuroevolution Through Erlang* (https://www.springer.com/computer/swe/book/978-1-4614-4462-6). Springer Verlag.
20. Vargas, Danilo Vasconcellos; Murata, Junichi. "Spectrum-Diverse Neuroevolution With Unified Neural Models" (https://dx.doi.org/10.1109/TNNLS.2016.2551748). *IEEE Transactions on Neural Networks and Learning Systems*.
21. Edlund, Jeffrey; Chaumont, Nicolas; Hintze, Arend; Koch, Christof; Tononi, Giulio; Adami, Christoph. "Integrated Information Increases with Fitness in the Evolution of Animats" (https://doi.org/10.1371/journal.pcbi.1002236). *PLOS Computational Biology*.

22. Rostami, Shahin; Neri, Ferrante (2017-06-01). "A fast hypervolume driven selection mechanism for many-objective optimisation problems" (http://www.sciencedirect.com/science/article/pii/S2210650216301328). *Swarm and Evolutionary Computation* **34** (Supplement C): 50–67. doi:10.1016/j.swevo.2016.12.002 (https://doi.org/10.1016%2Fj.swevo.2016.12.002).
23. "Multi-objective evolution of artificial neural networks in multi-class medical diagnosis problems with class imbalance - IEEE Conference Publication" (http://ieeexplore.ieee.org/abstract/document/8058553). *ieeexplore.ieee.org*. Retrieved 2017-11-28.

External links

- "Evolution 101: Neuroevolution | BEACON". *beacon-center.org*. Retrieved 2018-01-14.
- "NNRG Areas - Neuroevolution". *nn.cs.utexas.edu*. University of Texas. Retrieved 2018-01-14.</ref> (has downloadable papers on NEAT and applications)
- "SharpNEAT Neuroevolution Framework". *sharpneat.sourceforge.net*. Retrieved 2018-01-14. mature Open Source neuroevolution project implemented in C#/.Net.
- ANNEvolve is an Open Source AI Research Project (Downloadable source code in C and Python with a tutorial & miscellaneous writings and illustrations
- "Nils T Siebel - EANT2 - Evolutionary Reinforcement Learning of Neural Networks". *www.siebel-research.de*. Retrieved 2018-01-14.</ref> Web page on evolutionary learning with EANT/EANT2] (information and articles on EANT/EANT2 with applications to robot learning)
- NERD Toolkit. The Neurodynamics and Evolutionary Robotics Development Toolkit. A free, open source software collection for various experiments on neurocontrol and neuroevolution. Includes a scriptable simulator, several neuro-evolution algorithms (e.g. ICONE), cluster support, visual network design and analysis tools.
- "CorticalComputer (Gene)". *GitHub*. Retrieved 2018-01-14. Source code for the DXNN Neuroevolutionary system.
- "ES-HyperNEAT Users Page". *eplex.cs.ucf.edu*. Retrieved 2018-01-14.