Evolution of Artificial Neural Networks

Stefano Nolfi Domenico Parisi
Institute of Psychology
National Research Council
15, Viale Marx, 00137 - Rome
e-mail:nolfi@ip.rm.cnr.it parisi@ip.rm.cnr.it

Introduction

Artificial neural networks are computational models of nervous systems. Natural organisms, however, do not possess only nervous systems but also genetic information stored in the nucleus of their cells (genotype). The nervous system is part of the phenotype which is derived from this genotype through a process called development. The information specified in the genotype determines aspects of the nervous system which are expressed as innate behavioral tendencies and predispositions to learn. When neural networks are viewed in the broader biological context of Artificial Life they tend to be accompanied by genotypes and to become members of evolving populations of networks in which genotypes are inherited from parents to offspring (Parisi, 1997).

Artificial neural networks can be evolved by using evolutionary algorithms (Holland, 1975; Schwefel, 1995; Koza, 1992). An initial population of different artificial genotype, each encoding the free parameters (e.g. the connection strengths and/or the architecture of the network and/or the learning rules) of a corresponding neural network, are created randomly. The population of networks is evaluated in order to determine the performance (fitness) of each individual network. The fittest networks are allowed to reproduce (sexually or a-sexually) by generating copies of their genotypes with the addition of changes introduced by some genetic operators (e.g., mutations, crossover, duplication). This process is repeated for a number of generations until a network that satisfies the performance criterion (fitness function) set by the experimenter is obtained (for a review of methodological issue see Yao, 1993).

The genotype might encode all the free parameters of the corresponding neural network or only the initial value of the parameters and/or other parameters that affects learning. In the former case of the network is entirely innate and there is no learning. In the latter networks changes both philogenetically (i.e. through out generations) and ontogenetically (i.e. during the period of time in which they are evaluated).

Evolution and development

A cornerstone of biology is the distinction between inherited genetic code (genotype) and the corresponding organism (phenotype). What is inherited from the parents is the genotype. The phenotype is the complete individual that is formed according to the instructions specified in the genotype.

Evolution is critically dependent on the distinction between genotype and phenotype, and on their relation, i.e. the genotype-to-phenotype mapping. The fitness of an individual, that affect selective reproduction, is based on the phenotype, but what is inherited is the genotype, not the phenotype. Furthermore, while the genotype of an individual is one single entity, the organism can be considered as a succession of different phenotypes taking form during the genotype-to-phenotype mapping process, each derived from the previous one under genetic and environmental influences.

When the genotype-to-phenotype mapping process takes place during individuals' lifetime we can talk of development. In this case, each successive phenotype, corresponding to a given stage of development, has a distinct fitness. The total fitness of a developing individual is a complex function of these developmental phases. Evolution must ensure that all these successive forms are viable and, at the same time, that they make a well-formed sequence where each form leads to the next one until a mostly stable (adult) form is reached. This puts various constraints on evolution but it also offers new means for exploring novelty. Small changes in the developmental rates of different components of the phenotype, for example, can have huge effects on the resulting phenotype. Indeed it has been hypothesized that in natural evolution changes affecting regulatory genes that control the rates of development played a more important role than other forms of change such as point mutations (Gould, 1977).

Although the role of the genotype-to-phenotype mapping and of development has been ignored in most of the experiments involving artificial evolution, there is now an increasing awareness of its importance. Wagner & Altenberg (1996) write: "In evolutionary computer science it was found that the Darwinian process of mutation, recombination and selection is not universally effective in improving complex systems like computer programs or chip designs. For adaptation to occur, these systems must possess *evolvability*, i.e. the ability of random variations to sometimes produce improvement. It was found that evolvability critically depends on the way genetic variation maps onto phenotypic variation, an issue known as the representation problem." (p. 967).

Genetic Encoding

To evolve neural networks one should decide how to encode the network in the genotype in a manner suitable for the application of genetic operators. In most cases, all phenotypical characteristics are coded in an uniform manner so that the description of an individual at the level of the genotype assumes the form of a string of identical elements (such as binary or floating point numbers). The transformation of the genotype into the phenotypical network is called genotype-to-phenotype mapping.

In direct encoding schemes there is a one-to-one correspondence between genes and the phenotypical characters subjected to the evolutionary process (e.g. Miller et al., 1989). Aside from being biological implausible, simple one-to-one mappings has several drawbacks. One problem, for example, is scalability. Since the length of the genotype is proportional to the complexity of the corresponding phenotype, the space to be searched by the evolutionary process increases exponentially with the size of the network (Kitano, 1990).

Another problem of direct encoding schemes is the impossibility to encode repeated structures (such as network composed of several sub-networks with similar local connectivity) in a compact way. In one-to-one mappings, in fact, elements that are repeated at the level of the phenotype must be repeated at the level of the genotype as well. This does not only affect the length of the genotype and the corresponding search space, but also the evolvability of individuals. A full genetic specification of a phenotype with repeated structures, in fact, implies that adaptive changes affecting repeated structures should be independently rediscovered through changes introduced by the genetic operators.

Growing methods

The genotype-to-phenotype process in nature is not only an abstract mapping of information from genotype to phenotype but it is also a process of physical growth (growth in size and in physical structure). By taking inspiration from biology therefore, one may decided to encode in the genotype growing instructions. The phenotype is progressively built by executing the inherited growing instructions.

Nolfi et al. (1994) used a growing encoding scheme to evolve the architecture and the connection strenghts of neural networks that controlled a small mobile robot (for a similar method see Husband et al., 1994). These controllers were composed of a collection of artificial neurons distributed over a 2-dimensional space with growing and branching axons (Figure 1, top). Inherited genetic material specified instructions that controlled the axonal growth and the branching process of neurons. During the growth process, when a growing axonal branch of a particular neuron reached another neuron a connection between the two neurons is established. On the bottom of Figure 1 you can see the network resulting from the growth process displayed in the top of the Figure after the elimination of isolated and non-functional neurons. Axons grew and brunched only if the activation variability of the corresponding neurons was larger than a genetically-specified threshold. This simple mechanism is based on the idea that sensory information coming from the environment has a critical role in the maturation of the connectivity of the biological nervous system and, more specifically, that the maturation process is sensitive to the activity of single neurons (see Purves, 1994). Therefore the developmental process was influenced both by genetic and environmental factors (i.e. the actual sequence of sensory states experienced by the network influenced the process of neural growth).

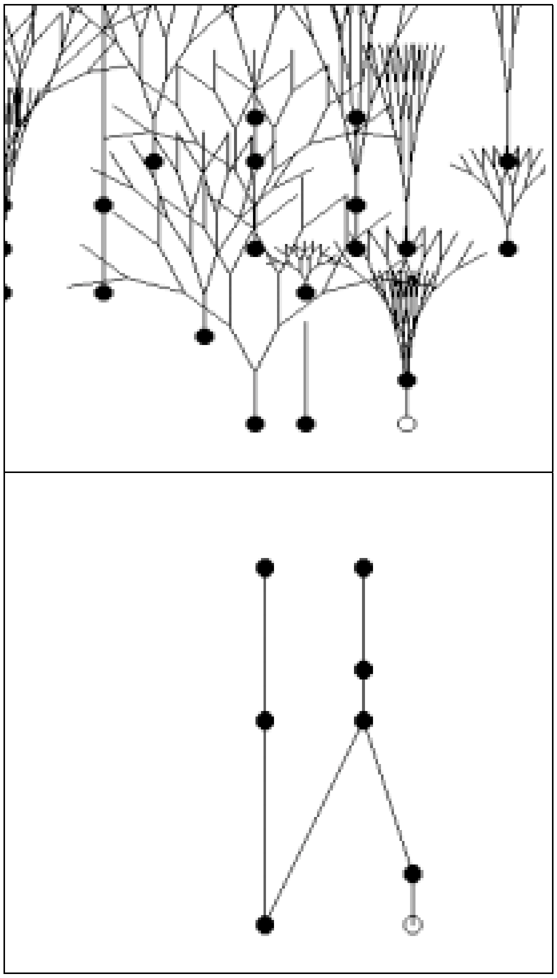

Figure 1. Development of an evolved neural network. Top: The growing and branching process of the axons. Bottom: the resulting neural network after removal of nonconnecting branches and the elimination of isolated neurons and groups of interconnected neurons.

This method allows the evolutionary process to select neural network topologies that are suited to the task chosen. Moreover, the developmental process, by being sensitive to the environmental conditions, might display a form of plasticity. Indeed, as shown by the authors, if some aspects of the task are allowed to vary during the evolutionary process, evolved genotypes display an ability to develop into different final phenotypical structures that are adapted to the current conditions.

Cellular Encodings

In natural organisms the development of the nervous system begins with a folding in of the ectodermic tissue which forms the neural crest. This structure gives origin to the mature nervous system through three phases: the genesis and proliferation of different classes of neurons by cellular duplication and differentiation, the migration of the neurons toward their final destination, and the growth of neurites (axons, dendrites). The growing process described in the previous section therefore characterizes very roughly only the last of these three phases. A number of attempts have been made to include other aspects of this process in artificial evolutionary experiments.

Cangelosi et al. (1994), for example, extended the model described in the previous section by adding a cell division and migration stage to the already existing stage of axonal growth. The genotype, in this case, is a collection of rules governing the process of cell division (a single cell is replaced by two "daughter" cells) and migration (the new cells can move in the 2D space). The genotype-to-phenotype process therefore starts with a single cell which, by undergoing a number of duplication and migration processes, produces a collection of neurons arranged in a 2D space. These neurons grow their axons and establish connection until a neural controller is formed (for a related approaches see Dellaert and Beer, 1994).

Gruau (1994) proposed a genetic encoding scheme for neural networks based on a cellular duplication and differentiation process. The genotype-to-phenotype mapping starts with a single cell that undergoes a number of duplication and transformation processes ending up in a complete neural network. In this scheme the genotype is a collection of rules governing the process of cell divisions (a single cell is replaced by two "daughter" cells) and transformations (new connections can be added and the strengths of the connections departing from a cell can be modified). In this model, therefore, connection links are established during the cellular duplication process.

The instructions contained in the genotype are represented as a binary-tree structure as in genetic programming (Koza, 1992). During the genotype-to-phenotype mapping process, the genotype tree is scanned starting from the top node of the tree and then following each ramification. The top node represents the initial cell that, by undergoing a set of duplication processes, produces the final neural network. Each node of the genotype tree encodes the operations that should be applied to the corresponding cell and the two sub-trees of a node specify the operations that should be applied to the two daughter cells. The neural network is progressively built by following the tree and applying the corresponding duplication instructions. Terminal nodes of the tree (i.e. nodes that do not have sub-trees) represents terminal cells that will not undergo further duplications. Gruau also considered the case of genotypes formed by many trees where the terminal nodes of a tree may point to other trees. This mechanism allows the genotype-to-phenotype process to produce repeated phenotypical structures (e.g. repeated neural sub-networks) by re-using the same genetic informations. Trees that are pointed to more than once, in fact, will be executed more times. This encoding method has two advantages: (a) compact genotypes can produce complex phenotypical networks, and (b) evolution may exploit phenotypes where repeated sub-structures are encoded in a single part of the genotype. Since the identification of sub-structures that are read more than once is an emergent result of the evolutionary process, Gruau defined this method Automatic Definition of Neural Subnetworks (ADNS) (Gruau, 1994).

Discussion

Artificial evolution can be seen as a learning algorithm for training artificial neural networks. From this point of view, one distinctive feature is the limited amount of feedback required. Supervised learning algorithms require immediate and detailed desired answers as a feedback. Reinforcement learning algorithms require less - only a judgement of right or wrong which should not be necessarily immediate. Viewed as a learning algorithm, artificial evolution requires still less - only an overall evaluation of the performance of the network over the entire evaluation period. A second distinctive feature is that any parameter of the neural network (e.g. the connection strengths, the network topology, the learning rules, the transfer function of the neurons) can be subjected to the evolutionary process.

Although systematic comparison between artificial evolution and other algorithms are not been done yet, it is reasonable to claim that artificial evolution tend to produce better results when detailed feedback is not available. This is the case, for example, of a neural networks that should control mobile robots (Nolfi and Floreano, 2000). In this case in fact, although the experimenter can provide a general evaluation of how much the behavior of a robot approximates the desired behavior, he or she cannot usually indicate what the robot should do each time step to produce such a desired behavior. Moreover artificial evolution might result more effective in those cases in which certain features of the network (such as the network topology or the transfer functions) that cannot be properly set by hand are crucial. Artificial evolution, in fact, provide a way to co-adapt different type of parameters.

The analogy with natural evolution however can also be considered more strictly. In this case the evolutionary process is not seen as an abstract training algorithm but as a process that mimics some of the key aspects of the evolutionary process in nature. From this point of view neural networks tend to be viewed as a part of a population of artificial organisms that adapt autonomously by interacting with the external environment.

References

Cangelosi, A., S. Nolfi, and D. Parisi, 1994. Cell division and migration in a 'genotype' for neural networks, Network- Computation in Neural Systems, 5:497-515.

Dellaert F. and R.D. Beer (1994). Toward an evolvable model of development for autonomous agent synthesis, in Proceedings of the Forth Conference on Artificial Life, (R. Brooks, and P. Maes, eds.), Cambridge, MA: MIT Press.

Gould, S.J., 1977. Ontogeny and Phylogeny. Cambridge, MA: Harward University Press.

Gruau F., 1994. Automatic definition of modular neural networks, Adaptive Behavior, 3:151-183.

Holland, J.J., 1975, Adaptation in natural and artificial systems. Ann Arbor, MI: University of Michigan Press.

Husbands, P., I. Harvey, D. Cliff, and G. Miller, 1994. The use of genetic algorithms for the development of sensorimotor control systems, in From Perception to Action, (P. Gaussier, and J-D. Nicoud, eds.), Los Alamitos CA: IEEE Press.

Kitano, H., 1990. Designing neural networks using genetic algorithms with graph generation system, Complex Systems, 4:461-476.

Koza, J.R., 1992. Genetic Programming: On the Programming of Computers by Means of Natural Selection, Cambridge, MA: MIT Press.

Miller, G.F., P.M. Todd, and S.U. Hedge (1989). Designing neural networks using genetic algorithms, in Proceedings Third International Conference on Genetic Algorithms, (L. Nadel, and D. Stein, eds.), San Mateo, CA: Morgan Kaufmann, pp. 379-384.

Nolfi, S., and D. Floreano, 2000. Evolutionary Robotics: The Biology, Intelligence, and Technology of Self-Organizing Machines, Cambridge, MA: MIT Press/Bradford Books.

Nolfi, S., O. Miglino, and D. Parisi, 1994. Phenotypic Plasticity in Evolving Neural Networks, in Proceedings of the Intl. Conf. From Perception to

Action, (D. P. Gaussier and J-D. Nicoud eds.), Los Alamitos, CA: IEEE Press, pp. 146-157.

Parisi, D., 1997. Artificial life and higher level cognition, Brain and Cognition, 34:160-184.

Purves, D., 1994. Neural activity and the growth of the brain. Cambridge: Cambridge University Press.

Schwefel, H.P., 1995. Evolution and Optimum Seeking, New York: Wiley Press.

Yao, X., 1993. A review of evolutionary artificial neural networks, International Journal of Intelligent Systems, 4:203-222.

Wagner, G.P., and L. Altenberg, 1996. Complex adaptations and the evolution of evolvability, Evolution, 50:967-976.