Memetic Computing: NICSO 2010 Special Issue manuscript No.
(will be inserted by the editor)

Where is the Brain inside the Brain?

On Why Artificial Neural Networks should be Developmental.

Julian F. Miller · Gul Muhammad Khan

Received: date / Accepted: date

Abstract Biological brains are capable of general learning without supervision. This is learning across mutiple domains without interference. Unlike artificial neural networks, in real brains, learned information is not purely encoded in real-valued weights but instead it is resides in many neural aspects. Such aspects include, dendritic and axonal morphology, number and location of synapses, synaptic strengths and the internal state of neural components. Natural evolution has come up with extraordinary 'programs' for neurons that allow them to build learning systems through group activity. The neuron is the 'brain within the brain'. We argue that evolving neural developmental programs which when executed continuously build, shape and adjust neural networks is a promising direction for future research. We discuss aspects of neuroscience that are important, and examine a model that incorporates many of these features that has been applied to a number of problems: wumpus world, checkers and maze solving.

Keywords First keyword · Second keyword · More

Julian F. Miller
Department of Electronics, University of York, Heslington, York, UK, YO10 5DD
Tel.: +44 (0)1904 32 2383
Fax: +44 (0)1904 32 2335
E-mail: jfm7@ohm.york.ac.uk

Gul Muhammad Khan
Electrical Engineering Department, University of Engineering and Technology, NWFP, Pakistan
Tel.: +92 91 92167968
Fax: +92 91 9216663
E-mail: gk502@nwfpuet.edu.pk

1 Introduction

Artificial neural networks (ANNs) were first proposed over sixty years ago [36]. Despite decades of research few would argue that they even approach the learning capabilities of relatively simple organisms. The relative lack of progress is all the more puzzling since, our understanding of neuroscience has increased enormously over the same period [24]. In addition, computer systems have undergone a sustained period of exponential improvements in speed. We argue that the one of the weaknesses of most ANNs models is that they encode learned knowledge in the form of connection strenths (i.e. weights). We refer to this as "the synaptic dogma" (SD). It has been known for some time that the SD causes problems, ever since French identified the fundamental problem of "catastrophic forgetting" (CF) [15]. This is apparent since when a trained ANN is re-trained on a new problem it forgets how to solve the original problem. This is not at all surprising since the learned information is only encoded in the weights and it is precisely these that are changed when the network is trained on any particular problem. In addition, it has been shown that learning the weights in a neural network is an NP-complete problem, and therefore computationally intractable [23]. However, Baum showed that if one allows the addition of neurons and weighted connections, ANNs "can solve in polynomial time any learning problem that can be solved in polynomial time by any algorithm whatever" [2]. These findings gave rise to a type of ANN called a "constructive ANN" [14][12]. Using supervised learning, these algorithms add neurons and connections in a measured way, adjusting weights incrementally until training errors are reduced. Crick made a general criticism concerning the validity from the point of view of neuroscience about such training algorithms, to quote: "Moreover, the theorists working on the subject are so remote from actual neurons that they have been cavalier in omitting one type of unit altogether. Obviously there should be a unit to compare the output of each output neuron with the signals from the teacher, in order to calculate the error (in back-prop nets this is done by the computer)" [7]. Quartz argued in 1999 "the evidence suggests that cortical development involves the progressive elaboration of neural circuits in which experience-dependent neural growth mechanisms act alongside intrinsic developmental processes to construct the representations underlying mature skills" [38]. Quinlan reviewed what he termed "dynamic networks" these are "any artificial neural network that automatically changes its structure through exposure to input stimuli" [40] [41]. He also noted that "Although there has been some work on the removal of weighted connections, these schemes have not explored a general framework where the number of connections is both increased and decreased independently of the number of processing units. According to the present view a more fruitful strategy for future work is to examine more closely the changes made to connections between units and not continue to concentrate on systems that add and remove whole units" [40]. Despite the advances made in constructive or dynamic neural networks, the algorithms manipulate neural networks as a whole and are not plausible in terms of developmental biology.

There have been few attempts to incorporate mechanisms that are closer to known developmental biology (we review these, in section 3. Some of these have taken a neuron-centric view, and tried to obtain some form of program that represents a neuron. However, very often it is an impoverished view of a neuron. The aim of this position paper is to identify and discuss properties of neural systems that could be incorporated into neuron-centric ANNs. We call these networks, developmental artificial neural networks (DANNs) since they utilize aspects that mimic developmental biology. We discuss a recent model that is based on many of these aspects. The long term aim is to arrive at a model of DANNs that will allow a single network to develop over multiple problem instances without losing a previously acquired ability on an earlier encountered problem.

The plan of the paper is as follows. In section 2 we begin with an examination of memory in brains and compare and contrast this with memory in ANNs. In section 3 we examine development in biological brains and discuss work on developmental artificial neural networks (DANNs). Section 4 briefly examines a number of important features on biological neural systems and their counterparts in ANNs and DANNs. Section 5 discusses a model that incorporates many of these features. The paper ends with conclusions and further work.

2 Memory and learning in biological brains versus memory and learning in ANNs

One of the most fundamental discoveries in neuroscience concerning memory is Long Term Potentiation (LTP) [3]. LTP is now considered to happen when there is a facilitation of chemical transmission between two neurons as a result of near coincident activity of pre- and post-synaptic elements [6]. This phenomenon was previously predicted by Hebb [19] (known as Hebbian learning). In Spiking Neural Networks, Hebbian learning is often used to train the networks [18]. However, even apart from the problem of catastrophic forgetting, there is much research that calls into question that memory in brains is solely, or even principally related, to synaptic strengths. Firstly, it is now known that most synapses are not fixed structures but are constantly pruned away and replaced by new synapses and learning is related strongly to this process [47]. Secondly, synaptic plasticity is not just a matter of increasing and decreasing the number of synapses, but rather the exact location of the synapses on the dendritic tree and the actual geometry of the dendritic branches is important! Indeed, Quartz and Sejnowski suggest that local dendritic segments may well constitute the brain's fundamental computational units [39]. Thirdly, there is a large body of research indicating that learning and environmental interaction are strongly related to structural changes in neurons. For forty years it has been known that dendrites are constantly growing in response to the external environment! Mice reared in the dark and then placed in the light develop new dendrites in the visual cortex withing days [56]. It is known that the animals

reared in complex environments where active learning is taking place have an increased density of dendrites and synapses [29][30]. Even more classic are the well known studies of songbirds in the breeding season, where it is known that there is an increase in the number, size and spacing of neurons [55] and the study of hippocampi of London taxi drivers, which showed that the posterior hippocampi of taxi drivers were significantly larger relative to those of control subjects [35]. Although direct evidence is still uncertain, it is now thought that small protusions on dendrites known as spines are linked with learning and long-term memory formation [45]. Dendritic spines are where excitatory synapses form on dendrites [54]. Rose argues that after a few hours of learning the brain is permanently altered "if only by shifting the number and position of a few dendritic spines on a few neurons in particular brain regions" [43]. Rose also points to the fundamental nature of *recall* in the brain. In brains the act of remembering alters and augments the original thing remembered. Another, almost obvious, aspect supporting the view that structural changes in the brain are strongly associated with learning, is simply that the most significant period of learning in animals happens in infancy, when the brain is developing [10]. The consensus of opinion at present is that many different brain regions are involved in different aspects and forms of memory (see articles in [13]).

3 Development in biological brains and ANNs

In the human body there are approximately 10^{14} cells while in the brain there are approximately, 10^{11}. Evolution has not had to evolve these cells independently as they all arise from the genome in the fertilized ovuum. The process whereby a single cell can produce a complete organism is called biological development. Indeed, the learning capability of brains is a consequence of a development. To make brains, evolution had to create the instructions in a single neuron, so that when it replicated and differentiated it could form a working brain. Evolutionary algorithms are often used to train ANNs, however there is an obvious flaw in this since, the size of the artificial genotype would have to grow with the network size. This potentially could cause severe scalability problems as very large genotypes may be very hard to optimise. Developmental evolutionary algorithms (DEAs) have been devised to alleviate this problem. In DEAs, the genotype-phenotype mapping is highly indirect and non-linear. Genes act like instructions and development is the process of executing those instructions and dealing with the highly parallel interactions between them and the structure they create [33]. As in biology, the genome has no direct interaction with the environment. The developing *phenotype* responds to and is shaped by interaction with the external environment. The genes interact with each other at various hierarchical levels and at different times within the phenotype. Quartz and Sejnowski argue that "learning is a dynamic interaction between a changing, structured environment and neural mechanisms. The neural machinery is extensively shaped by activity stem-

ming from the environment, while its intrinsic properties also constrain this modulation and play an indispensable role in shaping the resulting structures".

A number of ANN researchers have considered ways of incorporating development to help construct ANNs, for reviews of various approaches see [33][52]. A variety of genotype representations have been devised. Cangelosi et al. defined genotypes which were a mixture of variables, parameters, and rules (e.g. cell type, axon length and cell division instructions) [5]. Dellaert and Beer represented a cell program using Boolean networks [8]. Gruau adapted a form of Genetic Programming [32] to develop a neural network. Kitano encoded matrix re-writing rules to develop graphs [28]. Rust et al constructed a genotype consisting of developmental parameters (encoded in binary) that controlled the times at which dendrites could branch and how the growing tips would interact with patterns of attractants placed in an environment [44]. Jacobi created an impressive artificial genome regulatory network (GRN), where the genotype is a string of 64 letters (a to d). The GRN produced and consumed and produced simulated proteins that define various cell actions (protein diffusion movement, differentiation, division, threshold). After a cellular network had developed it was interpreted as a neural network [22]. Astor and Adami also encoded a form of GRN, however cells were predfined to exist in a hexagonal grid. Genes encoded conditions involving concentrations of simulated chemicals which determine the level of activation of cellular actions (e.g. grow axon or dendrite, increase or decrease weight, produce chemical) [1]. Federici encoded a cell program as a simple recursive neural network that allowed cells (on a two-dimensional grid) to replicate, release chemicals or die. The type and metabolic concentrations of a cell are used to specify the internal dynamics and synaptic properties of its corresponding neuron. The position of the cell within the organism is used to produce the topological properties of neuron: its connections to inputs, outputs and other neurons. Some researchers have studied the potential of Lindenmeyer systems [34] for developing artificial neural networks and generative design. Boers and Kuiper adapted L-systems to develop the architecture of artificial neural networks [4]. They evolved the rules of an L-system that generated feed-forward neural networks. They found that this method produced more modular neural networks that performed better than networks with a predefined structure. Hornby and Pollack evolved L-systems to construct complex robot morphologies and neural controllers [21, 20]. Downing favours a higher abstraction level and avoids the complexities of axonal and dendritic growth, while maintaining key aspects of cell signaling, competition and cooperation of neural topologies in nature [11].

Stanley introduced the idea of using evolutionary algorithms to build neural networks constructively (called NEAT).The NEAT approach begins with a simple structure, with no hidden neurons. It consists of a feed-forward network of input and output neurons, representing the input and output signals. As evolution progresses, the topology of the network is augmented by adding a neuron along an existing connection, or by adding a new connection between previously unconnected neurons [51]. However, the mechanism for producing more complex networks is through the use of evolutionary operators. This is

potentially very slow, since evolutionary operators use random processes. In addition, this approach has no biological plausibility since natural evolution does not operate on aspects of the brain directly. Recently, Stanley has introduced a promising extension to the NEAT approach called HyperNEAT [50] which uses an evolved generative encoding called a Compositional Pattern Producing Network (CPPN) [49]. The CPPN takes coordinates of pairs of neurons and outputs a number which is interpreted as the weight of that connection. The advantage this brings is that ANNs can be evolved with complex patterns where collections of neurons have similar behaviour depending on their spatial location. It also means that one evolved function (the CPPN) can determine the strengths of connections of many neurons. It is a form of non-temporal development, where geometrical relationships are translated into weights. However, it should be noted that although neurons in a developing brain are spatially sensitive, the development of the brain is very responsive to changing conditions both the internal and external (sensory) environments.

4 A selection of important neural features and their presence in ANNs and DANNs

This section compares and constrasts many features of biological neural systems with artificial neural networks, and neural developmental approaches. It is not possible to argue that this is a final or definitive list, it is simply a list of properties that the authors feel are very important to consider in creating new developmental models of neural networks. The brain is an extremely complex system in which a vast number of processes are interwined and nobody yet knows the relative importance of these processes to producing intelligent behaviour. We have devised a model that incorporates almost all of these properties, this is described in section 5. It models a neuron as a set of seven evolved computer programs.

Neuron structure: Biological neurons have dendrites and axons with branches. Each neuron has a single axon with a variable number of axon branches. In addition, it has a number of dendrites and dendrite branches. Some published developmental neural models have these features. ANNs have only nodes and connections, but there is no concept of branches.

Interaction of branches: In biological dendrites the signals from different branches interact with each other, whereas in ANNs and published neural development there is no concept of interaction between connections [53]. Although ANNs and neural development sometimes regard connections between neurons as dendritic, they are far from biological dendrites in the types of morphology that can exist.

Neural function: Biological neurons are decision making entities, they integrate the signal and fire. However, the potential that initiates firing is not a linear sum of weighted signals. Koch suggests that polynomial sums would be more realistic, where connections interact with each other via product terms [31] (page 347). Using a form of GP that can encode a large range of possible

mathematical relationships looks like a promising idea to try (see the next section where we describe a model that does this).

Resistance: Branches in biological neurons have the property of electrical resistance. Resistance not only affect the strength of the signal but is also related to the length of the branch. There is no concept of branch resistance in the ANN literature. Although some of the neural developmental models adopted this property for branching axons and dendrites they did not consider the affect of increases and decreases in resistance as consequence of variation in length of these branches.

Health: Biological neurons have property which one can loosely consider to be 'Health'. Since biological neurons that are inactive for a long time tend to dwindle away. There is no concept of health of neurons or branches in ANNs or neural development models.

Neural activity: This refers to the degree of activeness of neurons and neurites. In biology, not all neurons and neurites actively participate in every function that brain performs. In ANNs and neural development all neurons have to be processed before the function of the network can be assessed. So all the neurons are always active in ANNs.

Synaptic communication: Electrical signals are transferred from one neuron to another through synapses. Synapses are not just the point of contact, as considered in many published models of ANNs and neural development, they modulate the signal and provide a complex mechanism for signal transfer [46].

Arrangement of neurons: The overall architecture of both ANNs and many neural developmental systems is fixed once developed, whereas biological neurons exist in space, interact with each other and move their branches from one place to another. This means that the morphology of the network is time dependent and able to change during problem solving.

Spiking (information processing): Signals are passed from neuron to neuron via spikes (nerve impulses) in the biological brain, some of the ANN models (Spiking Neural Networks) and some neural development models used a spiking mechanism for signal processing. The dependence of neural processes on the levels of electrical potential (see activity dependent morphology below) only makes sense if neurons emit and receive spikes.

Synaptic plasticity: Plasticity means the ability to change or reform. This occurs at several levels in synapses [9]. It can mean that changes in the the probability of generating action potential in response to a fixed stimulus, depends on the previous pattern of input [17]. The numbers of receptors (sites of neurotransmitter action) on the membrane can also be altered by synaptic activity [16]. These processes can interact resulting in positive feedback effects, where some cells never fire and others may saturate at some maximal firing rate. Synaptic plasticity in SNN models is primarily based on the work of Hebb [19]. A recent development of this idea is the use of spike time dependent plasticity (STDP) rules for updating weights in SNN networks [42], [48].

Developmental Plasticity: Neurons in biological systems are in constant state of change, their internal processes and morphology change all the time based on the environmental signals [53].

[58]. This eliminates weaker synaptic contacts, but preserves and strengthens stronger connections. More common experiences, which generate similar sensory inputs, determine which connections to keep and which to prune. More frequently activated connections are preserved. Neuronal death occurs through the process of apoptosis, in which inactive neurons become damaged and die.

Activity Dependent Morphology: There are few proposed models in which changes in levels of activity (in potentials or signals) between neurons leads to changes in neural morphology. This is an extremely important aspect of real brains [57].

5 The brain within the brain is the neuron

Koch notes, "Contrary to received opinion, nerve cells are considerably more complex than suggested by the neural network community. Like morons, they are reduced to computing nothing but a thresholded sum of their inputs" [31]. Crick argued "Why not look at the brain, both to get new ideas and to test existing ones? The usual answer given by the psychologists is that the details of the brain are so horrendously complicated" [7]. Quartz and Sejnowski note "Human development accordingly consists of two processes, first a prolonged period of representation construction in which neural structures respond to the informational structure of the environment, and, second, rapid learning, made possible by the first." [39]. We feel that the former has been relatively unexplored. In developing brains, massive changes occur in response to sensory signals received from the environment. Moreover these changes occur in unsupervised learning situations.

One of the difficulties in attempting to create a dynamic computational model inspired by neuroscience is that, the internal dynamics of biological neurons are extremely complicated and many of these processes might be unnecessary in a machine learning technique. However, the biology of neurons (i.e. their gross morphology and connectivity) *is* sufficiently well understood [24], [46] to allow us to identify essential sub-systems that we could attempt to evolve in order to achieve a computational equivalent. Conventional models of neural networks do not consider the genetics of neurons and the unsupervised development of a network during learning.

In order to represent the computational processes of neurons, we argue that Genetic Programming (GP) [32] offers (at least in principle) the capability of representing neural programs and the transfer of genetic changes from generation to generation. It is important to note that in this approach neural networks are not evolved directly, instead, programs that continously change and shape neural structures are evolved. The aim is to evolve the *capability* for learning rather than using evolution to directly encode learned information.

To illustrate this approach we discuss briefly a recently proposed model that incorporates many of the processes discussed in section 4 [27][25]. It uses a graph-based form of genetic programming called Cartesian Genetic Programming [37]. The technique is called Cartesian Genetic Programming De-

velopmental Network (CGPDN). The model has idealized the behaviour of a neuron in terms of seven main processes.

1. Local interaction among neighbouring branches of the same dendrite.
2. Processing of signals received from dendrites at the soma and deciding whether to fire an action potential.
3. Synaptic connections which transfer potential through axon branches to the neighbouring dendrite branches.
4. Dendrite branch growth and shrinkage. Production of new dendrite branches, removal of old branches.
5. Axon branch growth and shrinkage. Production of new axon branches, removal of old branches.
6. Creation or destruction of neurons.
7. Updating the synaptic weights (and consequently the capability to make synaptic connections) between axon branches and neighbouring dendrite branches.

Each aspect is represented with a separate chromosome (CGP program). The advantage of having a compartmentalized model is that different aspects of the model can be examined separately. Their utility to the whole can be assessed and if necessary the different compartments of the model can be refined.

In the CGPDN neurons are placed randomly in a two dimensional grid so that they are only aware of their spatial neighbours (see figure 1). Each neuron is initially allocated a random number of dendrites, dendrite branches, one axon and a random number of axon branches. An integer variable that mimics electrical potential is used for internal computation in neurons and communication between neurons. Neurons receive information through dendrite branches this is processed by the evolved dendrite program (D)and transferred to the evolved soma program (S). S determines the final potential in the soma, which is compared to a threshold to determine whether it should fire or not. Axon branches transfer information only to dendrite branches in their proximity by passing the signals from all the neighbouring branches through a CGP program(AS), acting as an electro-chemical synapse, which in turn updates the values of potential only in neighbouring branches. The weight processing chromosome, W, adjusts the weights of potential connections to the synapse. The signal is transferred to the postsynaptic neuron having the largest weight. External inputs and outputs are also converted into potentials before being applied to the network.

The number of programs, that are run to transfer the potential from all active neurons to other active neurons is dependent on the number of active neural electrical components. Developmental programs determine the morphology of the neural network (i.e. how many dendrite branches, axosynapes and somae). The number of dendrites on each neuron is fixed, however the number of dendrite branches on each dendrite is variable and is determined by whether the developmental dendrite branch programs (DBL) in the past decided to grow ore eliminate new branches. Every neuron is invested with a

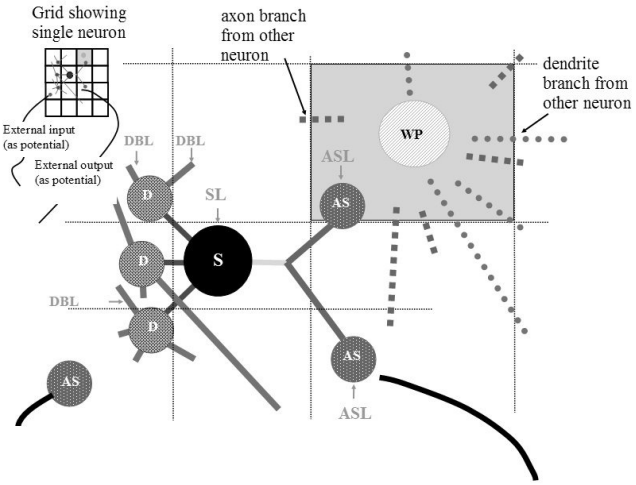

Fig. 1 On the top left a grid is shown containing a single neuron. The rest of the figure is an exploded view of the neuron. The neuron consists of seven evolved computational functions. Three are electrical and process a simulated potential in the dendrite (D), soma (S) and axo-synapse branch (AS). Three more are developmental in nature and are responsible for the life-cycle of neural components (shown in grey). They decide whether dendrite branches (DBL), soma (SL) and axo-synaptic branches (ASL) should die, change or replicate. The remaining evolved computational function (WP) adjusts synaptic and dendritic weights and is used to decide the transfer of potential from a firing neuron to a neighbouring neuron

single axon, however, the number of axosynapses attached to each axon is determined by whether axosynaptic branch program (ASL) in the past decided to grow or eliminate new axosynapses. The number of neurons (it starts with an initial number) is determined over time by whether soma developmental programs (SL) decided to replicate neurons or not. Whatever, the number of programs that are run in the developing neural network, the size of the genotype is fixed and depends only on the sizes of the seven chromosomes that give rise to a network. This is one of the advantages of the developmental approach. A relatively simple collection of evolved programs can define an entire network of arbitrary complexity. The CGPDN model has been evaluated on a number of problems in artificial intelligence. Wumpus world [27], Checkers [25] and maze solving [26]. Results show that the CGPDN produces networks that learn with experience (without further evolution). With structurally *different* networks they can recognize situations that have occured before and cause the same actions. For instance, we observed that in a series of games of checkers, CGPDN players make appropriate, and often the same move, when a new game starts even though the neural network is different from the network that existed at the start of a previous game.

6 Conclusions and future outlook

General learning refers to an ability to learn in multiple task domains without interference occuring. We have argued that evolving programs that continously build and adjust networks autonomously offers a favourable approach. In principle such an approach should allow the same evolved programs (genotypes) to modify networks (phenotypes) so that learning can take place across multiple domains. There remains much to be done. Determining the best way to represent suitable neural components as genotypes requires more research. A fundamental problem in creating general learning systems is the *encoding problem*. This is where the data that is fed into the neural networks has to be specifically encoded for each problem. Biological brains avoid this by using sensors to acquire information about the world and actuators to change it. We suggest that such universal representations will be required in order for developmental artificial neural networks to show general learning.

References

1. Astor, J.C., Adami, C.: A development model for the evolution of artificial neural networks. Artificial Life **6**, 189–218 (2000)
2. Baum, E.B.: A proposal for more powerful learning algorithms. Neural Computation **1**, 201–207 (1989)
3. Bliss, T.V.P., Lomo, T.: Long-lasting potentiation of synaptic transmission in the dentate area of the anesthetized rabbit following stimulation of the perforant path. Journal of Physiology **232**, 331–356 (1973)
4. Boers, E.J.W., Kuiper, H.: Biological metaphors and the design of modular neural networks. Masters thesis, Department of Computer Science and Department of Experimental and Theoretical Psychology, Leiden University (1992)
5. Cangelosi, A., Nolfi, S., Parisi, D.: Cell division and migration in a 'genotype' for neural networks. Network Computation in Neural Systems **5**, 497–515 (1994)
6. Cooke, S.F., Bliss, T.V.P.: Plasticity in the human central nervous system. Brain **129**, 1659–1673 (2006)
7. Crick, F.: The recent excitement about neural networks. Nature **337**, 129–132 (1989)
8. Dalaert, F., Beer, R.: Towards an evolvable model of development for autonomous agent synthesis. In: Brooks, R. and Maes, P. eds. Proceedings of the Fourth Conference on Artificial Life. MIT Press (1994)
9. Debanne, D., Daoudal, G., Sourdet, V., Russier, M.: Brain plasticity and ion channels. Journal of Physiology-Paris **97(4-6)**, 403–414 (2003)
10. Dekaban, A.S., Sadowsky, D.: Changes in brain weights during the span of human life. Ann. Neurol. **4**, 345–356 (1978)
11. Downing, K.L.: Supplementing evolutionary developmental systems with abstract models of neurogenesis. In: Proc. Conference on Genetic and Evolutionary Computation, pp. 990–996. ACM (2007)
12. Fahlman, S.E., Lebiere, C.: Advances in neural information processing systems, chap. The cascade-correlation architecture. Morgan Kaufmann (1990)
13. Federico Bermúdez-Rattoni, F. (ed.): Neural Plasticity and Memory From Genes to Brain Imaging. CRC Press (2007)
14. Frean, M.: The upstart algorithm: A method for constructing and training feedforward neural networks. Neural Computation **2**, 198–209 (1990)
15. French, R.M.: Catastrophic forgetting in connectionist networks: Causes, consequences and solutions. Trends in Cognitive Sciences **3(4)**, 128–135 (1999)
16. Frey, U., Morris, R.: Synaptic tagging and long-term potentiation. Nature **385**, 533–536 (1997)

17. Gaiarsa, J., Caillard, O., Ben-Ari, Y.: Long-term plasticity at gabaergic and glycinergic synapses: mechanisms and functional significance. Trends in Neurosciences **25(11)**, 564–570 (2002)
18. Gerstner, W., Kistler, W.: Spiking Neuron Models. Cambridge University Press (2002)
19. Hebb, D.: The organization of behavior. Wiley, New York (1949)
20. Hornby, G., Lipson, H., Pollack, J.B.: Generative representations for the automated design of modular physical robots. IEEE Transactions on Robotics and Automation **19**, 703–719 (2003)
21. Hornby, G., Pollack, J.B.: Body-brain co-evolution using L-systems as a generative encoding. In: Proc. Conference on Genetic and Evolutionary Computation, pp. 868–875 (2001)
22. Jakobi, N.: Harnessing Morphogenesis, Cognitive Science Research Paper 423, COGS. University of Sussex (1995)
23. Judd, S.: On the complexity of loading shallow neural networks. J. Complexity **4**, 177–192 (1988)
24. Kandel, E.R., Schwartz, J.H., Jessell, T.: Principles of Neural Science, 4th Edition. McGraw-Hill (2000)
25. Khan, G., Miller, J.: Evolution of cartesian genetic programs capable of learning. In: Proc. Conf. on Genetic and Evolutionary computation (GECCO), pp. 707–714 (2009)
26. Khan, G., Miller, J.: Solving mazes using an artificial developmental neuron. In: Proc. Conf. on Artificial Life (ALIFE) XII, pp. 241–248. MIT Press (2010)
27. Khan, G., Miller, J., Halliday, D.: Coevolution of intelligent agents using cartesian genetic programming. In: Proc. Conf. on Genetic and Evolutionary computation (GECCO), pp. 269 – 276 (2007)
28. Kitano, H.: Designing neural networks using genetic algorithms with graph generation system. Complex Systems **4**, 461–476 (1990)
29. Kleim, J.A., Lussnig, E., Schwartz, E.R., Comery, T.A., Greenough, W.T.: Synaptogenesis and fos expression in the motor cortex of the adult rat after motor skill learning. J. Neurosci **16**, 4529–4535 (1996)
30. Kleim, J.A., Vij, K., Ballard, D.H., Greenough, W.T.: Learning-dependent synaptic modifications in the cerebellar cortex of the adult rat persist for at least four weeks. J. Neurosci **17**, 717–721 (1997)
31. Koch, C.: Biophysics of Computation: Information Processing in Single Neurons. Oxford, U.K.: Oxford Univ. Press (1999)
32. Koza, J.: Genetic Programming: On the Programming of Computers by Means of Natural selection. MIT Press (1992)
33. Kumar S. (Editor), B.J.: On Growth, Form and Computers. Academic Press (2003)
34. Lindenmeyer, A.: Mathematical models for cellular interaction in development, parts i and ii. Journal of Theoretical Biology **18**, 280–315 (1968)
35. Maguire, E.A., Gadian, D.G., Johnsrude, I.S., Good, C.D., Ashburner, J., Frackowiak, R.S.J., Frith, C.D.: Navigation-related structural change in the hippocampi of taxi drivers. PNAS **97**, 4398–4403 (2000)
36. McCulloch, Pitts, W.: A logical calculus of the ideas immanent in nervous activity. The Bulletin of Mathematical Biophysics **5**, 115–133 (1943)
37. Miller, J.F., Thomson, P.: Cartesian genetic programming. In: Proc. of the 3rd European Conf. on Genetic Programming, vol. LNCS 1802, pp. 121–132. Springer (2000)
38. Quartz, S.: The constructivist brain. Trends in Cog. Sci **3**, 48–57 (1999)
39. Quartz, S., Sejnowski, T.: The neural basis of cognitive development: A constructivist manifesto. Behav. Brain. Sci **20**, 537–556 (1997)
40. Quinlan, P.T.: Structural change and development in real and artificial networks. Neural Networks **11**, 577–599 (1998)
41. Quinlan, P.T. (ed.): Connectionist Models of Development. Psychology Press (2003)
42. Roberts, P., Bell, C.: Spike-timing dependent synaptic plasticity in biological systems. Biological Cybernetics **87**, 392–403 (2002)
43. Rose, S.: The Making of Memory: From Molecules to Mind. Vintage (2003)
44. Rust, A., Adams, R., H., B.: Evolutionary neural topiary: Growing and sculpting artificial neurons to order. In: Proc. of the 7th Int. Conf. on the Simulation and synthesis of Living Systems (ALife VII), pp. 146–150. MIT Press (2000)

45. Segal, M.: Dendritic spines and long-term plasticity. Nature. Rev. Neurosci. **6**, 277–284 (2005)
46. Shepherd, G.: The synaptic organization of the brain. OUP (1990)
47. Smythies, J.: The Dynamic Neuron. MIT Press (2002)
48. Song, S., Miller, K.D., Abbott, L.F.: Competitive hebbian learning through spike-timing-dependent synaptic plasticity. Nat. Neurosci **3**, 919–926 (2000)
49. Stanley, K.O.: Compositional pattern producing networks: A novel abstraction of development. Genetic Programming and Evolvable Machines **8**, 131–162 (2007)
50. Stanley, K.O., D'Ambrosio, D.B., Gauci, J.: A hypercube-based encoding for evolving large-scale neural networks. Artificial Life **15**, 185–212 (2009)
51. Stanley, K.O., Miikkulainen, R.: Evolving neural network through augmenting topologies. Evolutionary Computation **10**, 99–127 (2002)
52. Stanley, K.O., Miikkulainen, R.: A taxonomy for artificial embryogeny. Artificial Life **9(2)**, 93–130 (2003)
53. Stuart, G., Spruston, N., Häusser, M. (eds.): Dendrites. OUP (2008)
54. Tada, T., Sheng, M.: Molecular mechanisms of dendritic spine morphogenesis. Curr. Opin. Neurobio. **16**, 95–101 (2006)
55. Tramontin, A.D., Brenowitz, E.: Seasonal plasticity in the adult brain. Trends in Neuroscience **23**, 251–258 (2000)
56. Valverde, F.: Rate and extent of recovery from dark rearing in the visual cortex of the mouse. Brain Res. **33**, 1–11 (1971)
57. Van Ooyen, A. (ed.): Modeling Neural Development. MIT Press (2003)
58. Van Ooyen, A., Pelt, J.: Activity-dependent outgrowth of neurons and overshoot phenomena in developing neural networks. Journal of Theoretical Biology **167**, 27–43 (1994)