An Attractor Network-Based Model with Darwinian Dynamics

Harold P. de Vladar
Harold.Vladar@parmenides-foundation.org

Anna Fedor
fedoranna@gmail.com

András Szilágyi [*]
and.szilagyi@gmail.com

István Zachar [†]
istvan.zachar80@gmail.com

Eörs Szathmáry [*†]
szathmary.eors@gmail.com

Parmenides Centre for the Conceptual Foundations of Science
Kirchplatz 1. 82094 Pullach/Munich, Germany

ABSTRACT

The human brain can generate new ideas, hypotheses and candidate solutions to difficult tasks with surprising ease. We argue that this process has evolutionary dynamics, with multiplication, inheritance and variability all implemented in neural matter. This inspires our model, whose main component is a population of recurrent attractor networks with palimpsest memory that can store correlated patterns. The candidate solutions are represented as output patterns of the attractor networks and they are maintained in implicit working memory until they are evaluated by selection. The best patterns are then multiplied and fed back to attractor networks as a noisy version of these patterns (inheritance with variability), thus generating a new generation of candidate hypotheses. These components implement a truly Darwinian process which is more efficient than both natural selection on genetic inheritance or learning, on their own. We argue that this type of evolutionary search with learning can be the basis of high-level cognitive processes, such as problem solving or language.

Keywords

Attractor network; autoassociative neural network; learning; evolutionary search; Darwinian dynamics; neurodynamics.

1. INTRODUCTION

Previous work in neuroscience has proposed the hypothesis that selection acts on a group of neurons [3, 2, 6]. This is suggestive of a synergy between learning and evolution. However, this synergy has not been exploited theoretically, partly because, although invoking selection in an analogous way as in evolution, these neuroscientific ideas regard the process as a "one shot", lacking the iterative (generational) component, and, importantly means to further variation at each round [9].

In order to properly explore the effects of learning on evolution, or, equivalently, an evolutionary implementation of learnable systems, it is necessary to consider three aspects: multiplication, inheritance and variability [12]. In evolutionary processes, either algorithms or biological, variability comes from mutations and recombination of heritable units. Furthermore, if these hereditary units affect survival, then they constitute units of evolution. Importantly, this definition does not restrict the nature of the units, in the sense that they might well be physical genes, organisms, linguistic constructs or, in the case of this work, neuronal networks.

We introduce a minimal model of evolution where the individuals are attractor networks. This is intended to be a proof of concept of possible mechanisms that occur in the brain. However, we point out that there are also algorithmic advantages, because, as shown before, the evolution with neurodynamics can be more powerful than selection or learning alone [8, 10, 4, 5].

2. METHODS

2.1 Recurrent attractor networks

The individuals in our population are recurrent attractor networks. These are networks that can learn several patterns; each learnt pattern is called a *prototype*. Prototypes become attractors: given the same input, the prototype, or a highly correlated variant, is returned (Fig. 1, steps 1-2). Also, there is certain set of inputs that will always return the same prototype. This defines as a basin of attraction towards the prototype, hence the name.

We consider binary neurons $\zeta \in \{-1, +1\}$ representing inactive and active states, respectively. In our model, a neuron i fires ($\zeta_i = +1$) if the total sum of incoming collaterals, h_i,

[*]Also at: Department of Plant Systematics, Ecology and Theoretical Biology, Research Group of Ecology and Theoretical Biology, Eötvös University and The Hungarian Academy of Sciences, Budapest, Hungary.

[†]Also at: Department of Plant Systematics, Ecology and Theoretical Biology, Institute of Biology, Eötvös University, Budapest, Hungary.

GECCO'16 Companion, July 20-24, 2016, Denver, CO, USA

 ISBN 978-1-4503-4323-7/16/07…$15.00

DOI: http://dx.doi.org/10.1145/2908961.2931672

is greater than 0, where

$$h_{ij}^m = \sum_{\substack{k=1 \\ k\neq i,j}}^{N} w_{ik}^{m-1}\zeta_k^m. \quad (1)$$

and w_{ik} are the incoming associative weights and m takes values in the patterns used for training.

We employ a learning scheme introduced by Storkey [13], in which the associative weights are updated according to

$$w_{ij}^m = \begin{cases} w_{ij}^{m-1} + \frac{1}{N}\left(\zeta_i^m\zeta_j^m - \zeta_i^m h_{ji}^m - h_{ij}^m\zeta_j^m\right) & \text{if } i \neq j \\ 0 & \text{if } i = j \end{cases} \quad (2)$$

This update scheme has a larger learning capacity $C = \kappa N$ ($\kappa = 0.25$) than the simpler and popular Hebb's rule ($\kappa = 0.14$) [13]. (Capacity is the number of different prototypes that a network can store.) More important than the capacity itself is the palimpsest property. Under Hebb's scheme if the capacity is surpassed, catastrophic forgetting ensues and the network will be unable to retrieve any of the previously stored patterns, forgetting all it has learnt. This does not occur under the scheme of Storkey, in which older prototypes are forgotten and replaced by new ones.

Our experiments used $\aleph = 20$ structurally identical attractor networks, each consisting of $N = 200$ neurons, implementing Storkey's palimpsest learning rule, and initially trained with random patterns.

2.2 Selection

We use a scheme of elitist selection, where fitness of a pattern is defined as follows (Fig.1, steps 3-4): After provocation of each network $i = 1, \ldots, \aleph$ we first evaluate Pearson's product-moment correlation r_i between the output pattern of the i$^{\text{th}}$ network, ζ_i^{out}, and an externally set target pattern T (e.g. a cognitive problem to solve). Then, the pattern with the highest r is selected to breed true: $\Omega \equiv \zeta_b^{\text{out}}$, where $b = \arg\max\{r_1, r_2, \ldots r_\aleph\}$. We implement this selection scheme according to the following algorithm:

```
1: function SELECT.BEST(n, P, T)
2:     for i ← 1, ℵ do
3:         o_i ← n_i(P_i)
4:         r_i ←Pearson.Corr(o_i, T)
5:     end for
6:     b ← arg max_i(r)
7:     return o_b
8: end function
```

Note that in this scheme, attractor networks act simply as a decoding function because they output a value that only depends only on their previous training. Thus, at this stage, although selection is implemented in accordance to evolutionary notions, it acts on non-heritable particles. Therefore, if the networks are provoked again with Ω, they will simply return Ω.

3. RETRAINING

Selection on variation is necessary, but if there is no heritability, there is no evolution. The analogous process to heritability in our system is implemented by retraining some of the networks in the population (Fig 1, step 5). This is achieved by training a subset of $\aleph_R$ networks (different in each generation) with Ω', a mutated version (with rate $\mu_R = 0.01$) of Ω. This step is crucial since it constitutes the basis for the Darwinian evolutionary search over attractor networks, as it effectively implements heritable variation though retraining.

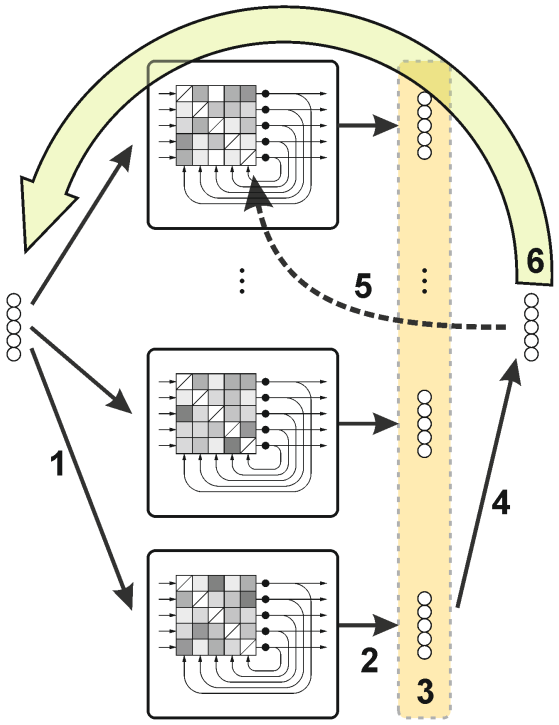

Figure 1: Evolutionary algorithm coupled with learning. 1: provocation. 2: Output of prototype variant. 3: evaluation of population output. 4: Selection of best pattern Ω. 5: Retraining of networks with mutated Ω'. 6: Provocation of population with mutated Ω''.

3.1 Mutation and sources of variability

As indicated in the introduction, we need to add sources of variability. First note that attractor networks are noisy in the sense that they return a pattern that has some minor variations to the prototype of the provoked attractor, and in that sense these are "mutated" versions of their provoked prototype. Occasionally, these mutated patterns fall into alternative different attractors of other networks in the population and consequently provoke prototypes that were not provoked before.

A second natural source of variation is the appearance of spurious patterns. In some cases, the network converges to a pattern different from any one learned previously. These spurious patterns are a linear combination of an odd number of stored patterns:

$$\zeta_i^{\text{spur}} = \pm\,\text{sgn}(\pm\zeta_i^{m_1} \pm \zeta_i^{m_2} \ldots \pm \zeta_i^{m_S}) \quad (3)$$

where S is the number of the stored patterns [11].

However, these two sources of variability act before selection, which means that in an iterative scheme all networks in the following round would be provoked with the same pattern, which eventually leads to a halt (results not shown). Therefore, after selection, we introduce further mutations

to the optimal pattern. The selected pattern Ω is mutated with rate $\mu_L = 0.005$ before redistribution to each of the $\aleph$ networks (Fig. 1, step 6).

3.2 Emulated-network model

An additional model for input-output was employed to emulate attractor networks without the need to explicitly employ neural networks. In this emulated model, each individual stores C_f patterns (C_f can be arbitrarily tuned, but for comparison we kept it close to the actual capacity C of the networks we employ). When an individual is fed with an input pattern it outputs the pattern that has the closest Hamming distance to the input, with additional noise ($\mu_e = 0.001$). This procedure emulates the almost-perfect recall property of the attractor networks, including their noisy behavior.

3.3 Evolutionary algorithm

By algorithmically combining the elements above, we are able to formally implement an evolutionary model that includes learning as heritable component.

Given the metaparameters $\aleph, N, T, \mu_L, \aleph_T$, and μ_T, the complete evolutionary algorithm is described in the following pseudocode:

procedure NEURODYNAMICS
  Declare $\aleph$ networks n each with N neurons
  for $i \leftarrow 1, \aleph$ **do** ▷ Train networks
    $\tau_i \leftarrow$ random binary vector of length N
    Train(n_i) with τ_i
  end for
  $\rho_0 \leftarrow$ random binary vector of length N
  $P_i \leftarrow \rho_0, i = 1, \ldots, N.$ ▷ Initial provocation vector
  while $\Omega \neq T$ **do**
    $\Omega \leftarrow$ Select.Best(n, P, T)
    for $i \leftarrow 1, \aleph$ **do** ▷ New provocation vector
      $P_i \leftarrow$ Mutate(Ω) with rate μ_L
    end for
    $R \leftarrow$ Random vector of length $\aleph_R$ ▷ Retraining
    for all $i \in R$ **do**
      $\tau_i \leftarrow$ Mutate(T) with rate μ_R
      Train(n_i) with τ_i
    end for
  end while
end procedure

4. RESULTS

Learning of new patterns into the attractors allows the networks to adapt to a solution by performing an evolutionary search. The results of the evolutionary experiments clearly prove that an appropriate architecture of attractor networks (Fig. 1) can implement evolutionary search.

We first focus on the evolutionary search on a single peaked landscape. Networks can build their own stepping stones (Fig. 2) in order to reach the global optimum. This is achieved though the learning of the actual best pattern of the previous generation.

In this scenario, neither the global optimum nor a route toward it is assumed to pre-exist in the system. We found that the system can converge to the global optimum, and this convergence is robust against a wide range of mutation rates. The speed of convergence to the optimum increases with the number of retrained networks $\aleph_R$ (Fig. 2).

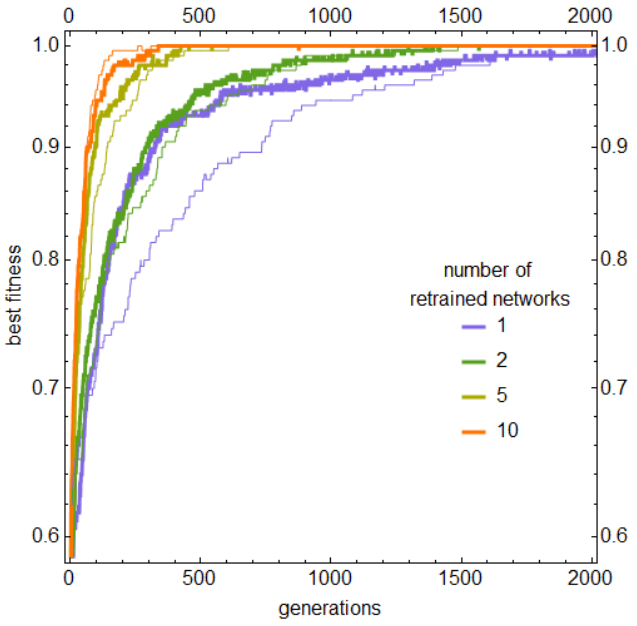

Figure 2: Evolutionary outcome of the learning-search for the optimum for different number of retrained networks. Thick lines: network model; thin lines: emulated morel. (See text for details.)

Note that the emulated-network model produces outputs comparable to the neuronal population. This suggests that non-genetic modes of inheritance have comparable evolvability to genetic inheritance, which is relevant to understand cultural evolution.

We now turn to a more challenging scenario, where the environment is periodically switching between two optima. For this we simply alternate the target pattern between T_1 and T_2 every $g_{env}(= 1000)$ generations. (This number was chosen large enough as to allow learning of the landscape.) After $g_{learn}(= 8000)$ we switch off network retraining. We chose $T_1 = (1, 1, \ldots, 1)$ and $T_2 = (-1, -1, \ldots, -1)$, both of dimension N.

Figure 3 shows that the network population can quickly adapt to changing environments. Different environments have fixed global optima that are revisited from time to time. After selectively finding and learning the optima of each of the two environments separately for a couple of periods, further learning is suppressed. The fact that networks are nevertheless able to recall the optimum fitness right after the environmental change proves that they use previously stored memories for this instead of repeatedly optimizing. For this recall, however, a partially correlated input cue is necessary to trigger the appropriate memory, otherwise the input would not fall in the basin of the required stored pattern. Note that the presence of an appropriate stored memory for the given environment makes the response to the environmental change almost instantaneous.

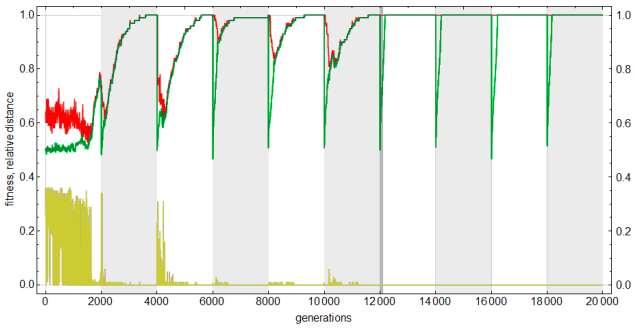

Figure 3: Fitness and recall accuracy over periodically alternating environments. Red: average fitness $\langle r \rangle$; green: best fitness (r_{o_b}); ochre: distance of the best output of the population from the closest one stored in memory.

5. DISCUSION

Evolution by means of selection is rationalised and described as a search on a fitness landscape, where a population across generations climbs to the peaks.

Attractor networks have a somewhat different mode than biological evolution. The reason is that networks, unlike organisms, just map inputs to outputs, sieving out the variation generated by mutation, in turn counteracting selection on the output values. In this sense, attractor networks work against variability, impeding hill-climbing. Therefore, it becomes justifiable and utterly necessary to include "heredity" through network retraining, which restores evolvability.

Our choice to use attractor networks for a proof of concept of Darwinian neurodynamics is motivated by the following factors. First, attractor networks employ the same architecture for generating, testing and storing novel patterns. Second, they can retrieve previously learnt patterns which facilitate evolutionary search by relying on past experience. Third, our choice of Storkey's learning model is significant because it naturally provides means to generate novelty through the superposition of attractors. We interpret the target T as the solution to an externally-posed problem and the hill-climbing elitist fitness landscape represents cognitive advance towards T.

However, note that there is still a relatively narrow window of action for evolution to happen. On the one hand, information transmission has to be accurate enough as to allow fitness increase. But on the other hand, if information transmission is too noisy, heritability is loss [7].

The equivalent idea was discussed by Adams [1] in his "Hebb and Darwin" paper. There, he discusses that synaptic replication and synaptic mutation are important factors for the brain to implement a Darwinian system.

Synaptic replication refers to the strengthening of existing synapses, in turn reflected in neural networks by the increase in synaptic weights. He realised that copying is not enough, but rather, inexact copies were necessary in order to test new variants against the other functional alternatives.

6. CONCLUSIONS

Our work evidences that the synergy between neuronal and evolutionary dynamics can implement an instance of natural evolution that can be more powerful than natural selection in the wild. This is because constrains which we find in genetics, such as limited mutation, are not limiting. This is thanks to the different nature of the network architectures relative to organisms that develop from genotypes.

Thus whilst network populations can on the one hand neutralise heritability if untrained, when endowed with training rounds during evolution can respond to selection more efficiently than most genetic systems [5].

Attractor networks thus offer a proof of concept of holistic replication of neuronal patterns that can have unlimited hereditary potential.

7. ACKNOWLEDGMENTS

Our research has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no 308943 (INSIGHT project).

8. REFERENCES

[1] P. Adams. Hebb and Darwin. *J. Thero. Biol.*, 195(4):419–438, 1998.

[2] J. P. Changeux. *Neuronal man: The biology of mind.* (Princeton University Press, Princeton, NJ, 1985.

[3] J. P. Changeux, P. Courrége, and A. Danchin. A theory of the epigenesis of neuronal networks by selective stabilization of synapses. *Proc. Nat. Acad. Sci.*, 70(10):2974–2978, 1973.

[4] S. Churchill, A W amd Sigtia and C. Fernando. Learning to generate genotypes with neural networks. *Evol. Comput.*, 2015.

[5] H. P. de Vladar and E. Szathmáry. Neuronal boost to evolutionary dynamics. *Interface Focus*, 5(6):20150074, 2015.

[6] G. M. Edelman. *Neural Darwinism. The theory of neuronal group selection.* Basic Books, New York, NY, 1987.

[7] M. Eigen. Self-organization of matter and the evolution of biological macromolecules. *Naturwissenschaften*, 58(10):465–523, 1971.

[8] C. Fernando, R. Goldstein, and E. Szathmáry. The neuronal replicator hypothesis. *Neural Comput.*, 22:2809–2857, 2010.

[9] C. Fernando, E. Szathmáry, and P. Husbands. Selectionist and evolutionary approaches to brain function: a critical appraisal. *Front. Comp. Neurosci.*, 6(24), 2012.

[10] C. Fernando, V. Vasas, E. Szathmáry, and P. Husbands. Evolvable neuronal paths: a novel basis for information and search in the brain. *PLoS ONE*, 6(8):e23534, 2011.

[11] J. Hertz, R. G. Palmer, and A. S. Krogh. *Introduction to the theory of neural computation.* Perseus Publishing, New York, NY, 1st edition, 1991.

[12] J. Maynard Smith. *The problems of biology.* Oxford University Press, Oxford, 1986.

[13] A. J. Storkey. *Efficient Covariance Matrix Methods for Bayesian Gaussian Processes and Hopfield Neural Networks.* PhD thesis, Neural System Group, Department of Electrical Engineering, Imperial College, University of London, London, UK, 1999.