



HOMEOSTATIC LEARNING RULE FOR ARTIFICIAL NEURAL NETWORKS

*M. Ruzek**

Abstract: This article presents an improvement of learning algorithm for an artificial neural network that makes the learning process more similar to a biological neuron, but still simple enough to be easily programmed. This idea is based on autonomous artificial neurons that are working together and at same time competing for resources; every neuron is trying to be better than the others, but also needs the feed back from other neurons. The proposed artificial neuron has similar forward signal processing as the standard perceptron; the main difference is the learning phase. The learning process is based on observing the weights of other neurons, but only in biologically plausible way, no back propagation of error or ‘teacher’ is allowed. The neuron is sending the signal in a forward direction into the higher layer, while the information about its function is being propagated in the opposite direction. This information does not have the form of energy, it is the observation of how the neuron’s output is accepted by the others. The neurons are trying to find such setting of their internal parameters that are optimal for the whole network. For this algorithm, it is necessary that the neurons are organized in layers. The tests proved the viability of this concept – the learning process is slower; but has other advantages, such as resistance against catastrophic interference or higher generalization.

Key words: *artificial neural network, learning rule, biological neuron*

Received: January 12, 2018

DOI: 10.14311/NNW.2018.28.011

Revised and accepted: April 17, 2018

1. Introduction

Although the original inspiration for ANN was the biological neuron [1], currently most of the artificial neural networks are very different from biological paradigm. A typical example is the most common learning algorithm, back propagation, where the error back-propagation is used for training, an idea that is not realistic in biological network.

The presented research is aimed to the design of algorithm that is using principles similar to biological neuron.

The idea of bringing ANN back to nature is not new and in fact many updates of NN algorithms are inspired by natural processes. For example, in [2] it is presented

*Martin Ruzek; Czech Institute of Informatics, Robotics and Cybernetics, E-mail: martin.ruzek@cvut.cz

a brain based automata that is suggested for the development of hardware for recognition systems. The main contribution is a study of selective system in neural networks.

In [3] it is shown that the recognition capability can be simulated by large-scale networks of spiking neurons. This simulation demonstrated how large networks can respond to input patterns by generating autonomous neural activity.

There are several other examples of research that are applying the inspiration by brain structures into the field of artificial intelligence, such as grey wolf optimization [4], genetic feature selector [5], associative memory [6] or fuzzy neural network [7]. In many applications, the biological motivation for the paradigm of neural networks is an advantage because the modeled processes are also of biological origin, such as automated analysis of medical or physiological data [8, 9] or mental processes modeling [10–12].

In [13] a process of simultaneous discharge of various neurons in a spiking neural network is presented. This process is similar to what happens in biological neural networks. A simple model consisting of only 2 equations of spiking neurons can be found in [14]. With this simple model, it is possible to observe similar behavior in terms of spiking, bursting and adaptation as in the biological mammalian network.

A neuronal network inspired by the cerebral cortex composed of 100 000 interconnected neurons presented in [15, 16]. The model led to identification of receptive and projective fields that are similar to those of vivo. This confirms the idea that with relatively uncomplicated model it is possible to induce very complex processes.

Some projects go even further, such as those that are targeted on isomorphic model of the whole biological neural network, such as the Blue Brain Project (BBP) which attempts to create a detailed model of the human brain with the resolution of particular molecules [17, 18]. The disadvantage of this approach is the lack of knowledge about the brain organization and the fact that even with an exact copy of the biological network, the desired processes such as self-learning, creativity or problem solving may not emerge.

The motivation of this research is focused more on the theoretical possibilities of the biologically plausible learning algorithms. The scope is not to try to copy the learning process of the biological network, but to use the biological template to design new learning rule that can be later used in many different areas of artificial intelligence.

2. Methodology

2.1 Concept of learning

The basic requirement for the proposed algorithm is that the learning process should be similar to a biological neuron. This goal is hard to achieve because of the high complexity of the biological neuron; however, it is possible to simplify the biological neuron by reducing its information channels (e. g. by leaving out chemical transmitter and focusing only on action potentials).

The proposed neuron shall not use any information that is not present in the biological network. The neuron is receiving the signal from other neurons and observing what happens with the signal that it sends, but only within the neurons

to which it is connected. It can not 'see' behind the neurons in the upper layer. No higher structure, error propagation etc. are allowed.

The topology of the neuron is based on a standard paradigm, which means every neuron is connected to neighboring neurons by dendrites, synapses and axons. The dendrites conduct the input to the neuron, the axon transmits its output towards other neurons and the synapse is the interface between axon and dendrite. Every neuron has only one axon, but the number of dendrites and synapses is not limited.

In further text, the term 'output neuron' will be used for any neuron that is receiving signal from the reference neuron and the term 'input neuron' for any neuron that is reading the signal of the reference neuron.

Analogically, the term input weight (w^i) will be used for the weights that are leading the signal into the reference neuron and output weights (w^o) for weights that are transmitting the signal from the reference neuron.

In this model, the input weight of an artificial neuron corresponds to the dendrites and the output weight to an axon. The weights are shared - the output weight of one neuron is input weight of the neuron in a higher layer. The neuron can read both its input and output weight, but can modify only its input weight.

The biological neuron is significantly more complex and uses different types of communication; however, even though this model is much more simple, it keeps the most important principles of electrical pulses.

2.2 Forward phase of signal processing

In the proposed model, the forward phase is composed of two steps, as in the perceptron model. In the first step, the neuron computes the weighted sum of its inputs:

$$s = \sum_{j=1}^n x_j \cdot w_j^i,$$

where x_j is the j -th input and w_j^i is the j -th input weight. In the second step, this sum is used as an input to the transfer function. In our case, the sigmoid function was always used

$$y = \frac{1}{1 + e^{-\alpha s}}.$$

The weights are limited to interval $(-1; 1)$; this interval was chosen for the sake of biological plausibility, therefore signal can not be gained, but can be reversed as in the case of inhibitory neurons.

In the described realization, only the weights w^i are subject to changes, the slope α is considered constant.

Reason for this decision is the intention to focus on the learning algorithm by simplifying the number of variables.

On the other hand, if the slope is also updated during the learning process, the learning may be faster, which might be the goal of future research.

2.3 Learning phase

In biological neural network there does not exist any dedicated channel for error back propagation, therefore this is also not used in the proposed model. Instead

of the error propagation, the neurons are gathering information about their functionality from other neurons only by observing their weights.

From the point of view of the reference neuron, the signal is being propagated in only one direction, from the neurons in the lower (input) layer to neurons in the higher (output) layer. The reference neuron is aware of how well its output is accepted by the other neurons because it knows the weights of the neurons in the higher layer. In other words, the reference neuron is setting the input weights based on the output weights (opposite direction than in the forward phase).

2.4 Utility

The reference neuron needs to convert the output weights into one number that describes how well its output is accepted by other neurons. This number, here referred to as *utility*, is an additional value which helps the neuron to find the optimal setting. It quantifies how well is the output of the neuron is accepted by other neurons. Utility is calculated from the output weights. There are several ways how to calculate the utility are possible, such as:

$$u = \sum_{i=1}^n w^o, \quad (1)$$

$$u = \max(w^o), \quad (2)$$

$$u = \text{med}(w^o), \quad (3)$$

$$u = Q_\alpha(w^o). \quad (4)$$

The first option (utility is the sum of all output weights) gives the same importance to all output neurons. This option is the most biologically plausible because the neuron is aware of which part of its output energy was accepted by other neurons, without distinguishing particular output neurons. The disadvantage is that in the case of highly connected networks (axon of biological neuron can be connected to 10 000 synapses) the learning process will be very slow.

The second option uses the maximum of output weights as the value of utility. In this situation, the reference neuron is 'working' only for one neuron in the higher layer. In the first step it finds the neuron with highest weight and in the next steps it is trying to maximize this weight. The main disadvantage is that this concept is not reflecting the real situation of biological neuron.

Between these two options are many compromise solutions that take into account a certain part of the output neurons, such as median (working for the half of neurons with greatest absolute weights), quintile (working for a certain part of the output neurons) or any weighted sum.

2.5 Updating the weights based on utility

The neuron uses one of the equations (1)–(4) to improve its weights. In two consecutive steps the utility should have an increasing trend, otherwise the last change had a negative effect and should be revoked.

The comparison of utility between two or more following steps means that the neuron must be equipped with memory to store the previous values of utility and the respective weights.

3. Realization

3.1 Problem of the highest layer

The described learning algorithm can be used for all neurons except for the highest layer. The reason is that the output weights over the highest layer are not part of the neural network, therefore there is no control about their values.

This problem is solvable in biological networks where all weights are part of a closed loop and the outputs are connected back to the input.

It is possible to design network that is part of closed loop; however, this solution will significantly increase the complexity of the network and will raise new problems with stability. As the scope of this work is to study the behavior of a particular neuron, the solution with a closed-loop network was rejected and the problem of the highest layer was solved by applying a different learning algorithm - back propagation - to this layer. This is a deviation from the original idea; however it makes possible to observe in detail the learning process of the neurons in the hidden layer.

3.2 Learning algorithm

The Fig. 1 represents the proposed learning rule that takes into account the problem of the highest layer (step D), for which the learning rule is not applicable. In case of closed loop, this algorithm would be used for all neurons.

3.3 Used data

The learning concept was tested on both artificial and real data. The artificially generated data were calculated as the absolute value of sinusoidal function of four input variables from interval $\langle 0;1 \rangle$:

$$s = \sin |x_1 + x_2 + x_3 + x_4|.$$

Real data was used from the iris flower dataset, which contains 4 characteristics of different species of iris flower. This dataset is often used for tests of machine learning.

The datasets were enhanced by 2 columns with dummy values. These inputs don't carry any useful information (no relation between these inputs and the desired output). This was done to test if the network is able to recognize that these inputs are noise and not to read them.

The dataset was divided into 2 subsets: the first contained 50 % of the data and was used to measure the learning speed (how many iterations are needed to achieve the desired MSE). The second part included also the remaining 50 % (is equal to the whole dataset) and was used to measure the generalization and learning capacity.

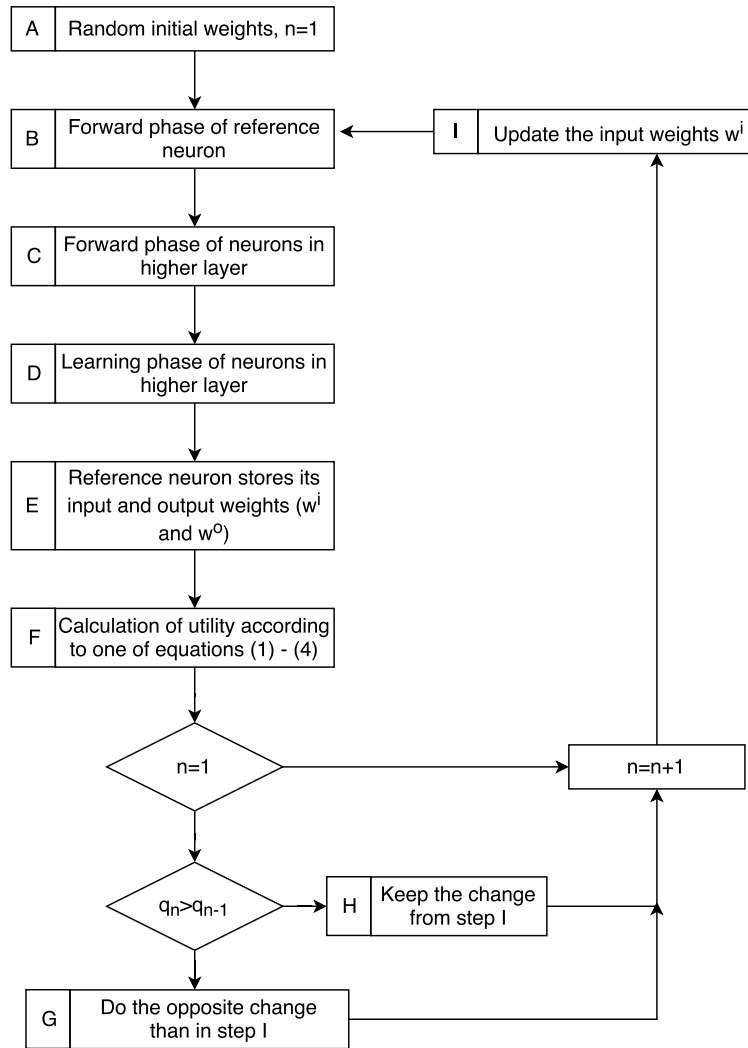


Fig. 1 Simplified algorithm of the proposed learning rule. The steps I and G are in fact more complicated than simple confirmation or rejection of the last change, but it based on the difference between q_n and q_{n-1} the new weights are calculated.

3.4 Topology of the network

The neurons of the proposed network are organized into layers, which corresponds to the structure of the neocortex. No lateral (in the same layer) or circular (pair of neurons connected by 2 weights in opposite direction) connections are allowed.

The topology is not identical to the biological network where the presence of lateral or circular weights can be observed, but these are not too frequent.

There is no standard way how to set the topology of the neural network, therefore equation from [21] was used to estimate the optimal number of hidden nodes:

$$h = 2 \cdot n + 1,$$

where n is the number of inputs. The topology [6 13 1] was used because the test dataset contained 6 inputs (Fig. 2).

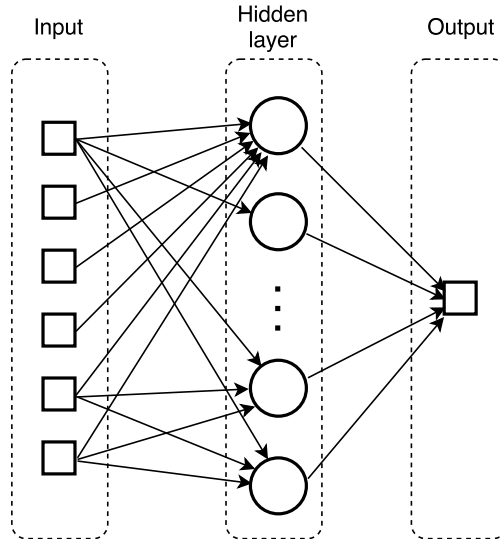


Fig. 2 Topology of the network used for testing.

3.5 Criteria for evaluation

The purpose of the testing is twofold: to prove the viability of this idea and to compare it with some reference method. As a reference method, the back propagation algorithm in batch mode was chosen. The topology of the tested and reference network was always the same.

The evaluation was based on 3 criteria: speed, generalization and capacity of learning.

The *speed of learning* is calculated as the number of learning epochs that are needed to achieve the requested MSE (in this case 0.01).

The *generalization* is the resistance to over fitting, which is a typical problem of artificial neural networks. The network can learn to model the training dataset very accurately, but fails with new data. The reason is that the network was trained to the concrete data with noise rather than to the real process.

Practically, the generalization was evaluated by feeding the network with testing dataset immediately after the network was trained with the training dataset. The

relation of MSE with testing and training dataset then describes the generalization abilities – the closer these numbers are, the better is the generalization ability.

The *continuity of learning* is an important property of neural networks which ensures resistance to phenomena of catastrophic interference [22,23]. The biological neural networks are able to continue learning even when the environment changes. A simple test was done to evaluate the continuity of learning.

The network was trained with training dataset until the satisfactory MSE was reached. Then testing dataset was presented. After that learning continued until the satisfactory MSE was again reached with the testing dataset. The capacity was then evaluated as the number of learning epochs with the testing dataset.

To avoid the influence of random error, every measurement was repeated 10 times.

These characteristics are illustrated on Fig. 3.

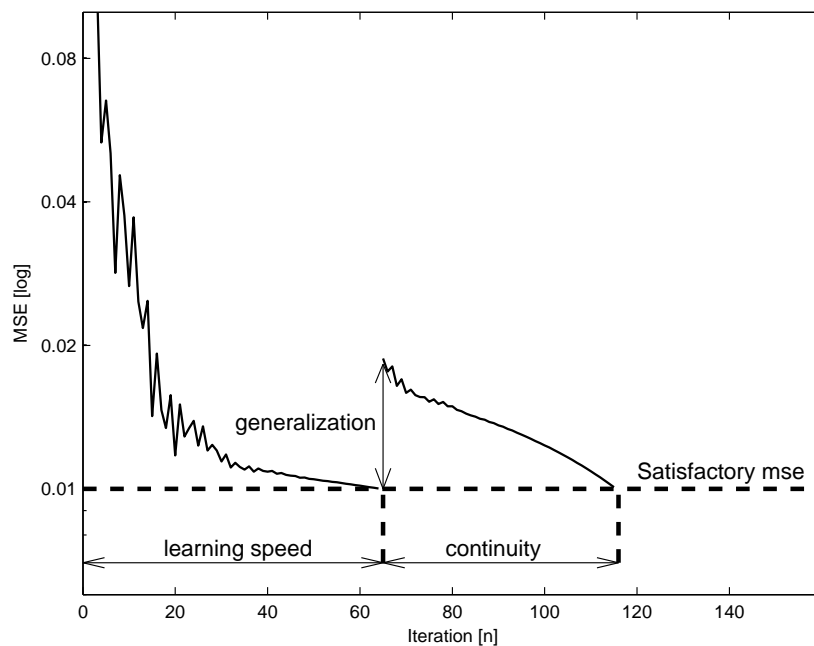


Fig. 3 Example of learning process with artificial data. The criteria used for evaluation are the learning speed, generalization and continuity of learning.

4. Results

The calculations were performed in MATLAB 2013.

In terms of speed, the back propagation algorithm performed better than any of the tested types of homeostatic learning. This applies for both training and testing dataset, so that in the categories *learning speed* and *continuity*, the back propagation algorithm was always the winner.

With respect to the generalization, all the variants of the homeostatic network provided better results in case of artificial data, but not in the case of the iris flower dataset.

If we take into account only the networks with homeostatic learning process, the best results were achieved when the maximum utility rule according to Eq. (2) was used. With this rule, the learning was significantly faster for both training and testing dataset. The generalization was similar for all equations. The probable reason is the small size of the tested network.

One interesting finding is that the learning process with the homeostatic network is more stable, the results with the back propagation algorithm had much higher variability.

The results are shown in Tabs. I and II.

It is important to keep in mind that these results are valid only for the given dataset and network topology.

The most important conclusion is however the confirmation of the learning ability of the proposed algorithm.

Network Type	Utility rule	Speed [epochs]	Generalization [%]	Capacity [epochs]
HN	$\sum(w^o)$	109	62	295
HN	$\text{med}(w^o)$	100	62	297
HN	$\text{max}(w^o)$	74	64	16
MLP	n.a.	50	72	15

Tab. I Comparison of back propagation and homeostatic learning algorithm with artificial data.

Network Type	Utility rule	Speed [epochs]	Generalization [%]	Capacity [epochs]
HN	$\sum(w^o)$	99	55	183
HN	$\text{med}(w^o)$	174	90	184
HN	$\text{max}(w^o)$	158	60	152
MLP	n.a.	64	48	77

Tab. II Comparison of back propagation and homeostatic learning algorithm with iris flower dataset.

5. Conclusions

The test results proved the viability of the described learning rule in terms of generalization and learning capacity. The speed of learning was slower than in case of back-propagation, but this criterion is not always the most important.

The proposed network is working in a biologically plausible way. The neuron is working only with the inputs that are present in a biological neural network. In many tasks the biological neural networks perform better than the artificial networks; decision making or creativity are two of many examples. Therefore, adapting the principles of biological neural networks into an artificial environment is a good way to increase the possibilities of neural networks.

The learning idea was tested on both artificial and real world data. In both cases, the network was converging. For the purpose of test, a part of the network was trained with the back propagation algorithm. This is a deviation from the original idea; however it makes it possible to observe the homeostatic learning in the rest of the network.

The speed of learning is slower than in the case of the reference algorithm. This is expected result, because the proposed algorithm has the disadvantage that it is not using all the inputs that the standard artificial neural networks.

The main advantage of the proposed method is the generalization. This is an important conclusion, because problems like falling into local minima are a significant limitation of the gradient algorithm.

The continuation of this research will be the amplification of the test data to find the suitable tasks for this type of network.

References

- [1] MCCULLOCH W., PITTS W. A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematical Biophysics*, 1943/5, pp. 115–133.
- [2] REEKE G.N., EDELMAN G.M. Selective Neural Networks and Their Implications for Recognition Automata, *International journal of supercomputer applications and high performance computing*, 1987/1, pp. 44–69.
- [3] MCKINSTRY J.L., EDELMAN G.M.: Temporal sequence learning in winner-take-all networks of spiking neurons demonstrated in a brain-based device, *Frontiers in neurobotics*, 2013/7, doi: [10.3389/fnbot.2013.0010](https://doi.org/10.3389/fnbot.2013.0010)
- [4] MOSAVI M.R., KHISHE M., GHAMGOSAR A. Classification of sonar data set using neural network trained by Gray Wolf Optimization, *Neural Network World*, 2016, 26(4), pp. 393–415, doi: [10.14311/NNW.2016.26.023](https://doi.org/10.14311/NNW.2016.26.023).
- [5] SASIKALA S., APPAVU S., S. GEETHA S. Improving detection performance of artificial neural network by Shapley value embedded genetic feature selector, *Neural Network World*, 2016, 16(2), pp. 175–201, doi: [10.14311/NNW.2016.26.010](https://doi.org/10.14311/NNW.2016.26.010).
- [6] FENG N., YAO Y. No rounding reverse fuzzy morphological associative memories, *Neural Network World*, 2016, 26(6), pp. 571–587, doi: [10.14311/NNW.2016.26.033](https://doi.org/10.14311/NNW.2016.26.033)
- [7] HANG Y., ZHENG C.-D. Novel stochastic stability conditions of fuzzy neural networks with Markovian jumping parameter under impulsive perturbations, *Neural Network World*, 2016, 26(6), pp. 543–557, doi: [10.14311/NNW.2016.26.001](https://doi.org/10.14311/NNW.2016.26.001).
- [8] SEENIVASAGAM V., CHITRA R. Myocardial infarction detection using intelligent algorithms, *Neural Network World*, 2016, 26(1), pp. 91–110, doi: [10.14311/NNW.2016.26.005](https://doi.org/10.14311/NNW.2016.26.005).
- [9] HLAVICA J., PRAUZEK M., PETEREK T., MUSILEK P. Assessment of Parkinson's disease progression using neural networks and ANFIS models, *Neural Network World*, 2016, 26(2), pp. 111–128, doi: [10.14311/NNW.2016.26.006](https://doi.org/10.14311/NNW.2016.26.006).
- [10] KIM S., YU Z., LEE M. Understanding human intention by connecting perception and action learning in artificial agents, *Neural Networks*, 2017, 92, pp. 29–38.
- [11] LEE S., LEE C., KWAK D., HA J., ZHANG B. Dual-memory neural networks for modeling cognitive activities of humans via wearable sensors, *Neural Networks*, 2017, 92, pp. 17–28.

- [12] SILVER D. et al: Mastering the game of Go with deep neural networks and tree search, *Nature*, 2016, 529, pp. 484–489, doi: [10.1038/nature16961](https://doi.org/10.1038/nature16961).
- [13] SUSI G., CRISTINI A., SALERMO M.: Path multimodality in a Feedforward SNN module, using LIF with Latency model, *Neural Network World*, 2016, 26(4), pp. 363–376, doi: [10.14311/NNW.2016.26.021](https://doi.org/10.14311/NNW.2016.26.021)
- [14] IZHIKEVICH E.M.: Simple model of spiking neurons, *IEEE TRANSACTIONS ON NEURAL NETWORKS*, 2003, 6, pp. 1569–1572.
- [15] IZHIKEVICH E.M., GALLY J.A., EDELMAN G.M.: Spike-timing dynamics of neuronal groups, *CEREBRAL CORTEX*, 2004, 8, pp. 933–944.
- [16] FROEMKE R.C.: Spike-timing-dependent synaptic modification induced by natural spike trains, *NATURE*, 2002, 416, pp. 433–438, doi: [10.1038/416433a](https://doi.org/10.1038/416433a)
- [17] PECK Ch. et al: Network-Related Challenges and Insights from Neuroscience, Bio-Inspired Computing and Communication, *BIOWIRE -First Workshop on Bio-Inspired Design of Networks*, 2007, Cambridge, pp. 67–78.
- [18] MARKRAM H. et al: Introducing the Human Brain Project, *The European Future Technologies Conference and Exhibition*, 2011, 7, pp. 39–42.
- [19] YOLCU U., BAS E., EGRIOGLU E., ALADAG C.H.: A New Multilayer Feedforward Network Based on Trimmed Mean Neuron Model, *Neural Network World*, 2015, 25(6), pp. 587–602, doi: [10.14311/NNW.2015.25.029](https://doi.org/10.14311/NNW.2015.25.029).
- [20] BORGES F.S., PROTACHEVICZ P.R., LAMEU E.L., BONETTI R.C., BATISTA A.M.: Synchronised firing patterns in a random network of adaptive exponential integrate-and-fire neuron model, *Neural Networks*, 2017, 90, pp. 1–7.
- [21] MIRJALILI S., LEWIS A.: S-shaped versus V-shaped transfer functions for binary Particle Swarm Optimization, *Swarm and Evolutionary Computation*, 9/2013, pp. 1-14, doi: [10.1016/j.swevo.2012.09.002](https://doi.org/10.1016/j.swevo.2012.09.002).
- [22] MUSCA S.C., ROUSSET S., ANS B. Artificial neural network whispering to the brain: Nonlinear system attractors induce familiarity with never seen items, *Connection Science*, 2009, 21(4), pp. 359–377.
- [23] NIKOLIC D. Practopoiesis: Or how life fosters a mind, *Journal of Theoretical Biology*, 2015, 373, pp. 40–61.