UDC 004.942
DOI: 10.20535/SRIT.2308-8893.2018.4.11

SIMPLE MODEL FOR SEQUENCE PREDICTION BASED ON DENDRITIC SPATIOTEMPORAL INTEGRATION

V.M. OSAULENKO

Abstract. Recent experiments on dendritic spatiotemporal integration reveal the much bigger computational potential of a single neuron. An individual dendritic branch can work as a coincidence detector due to a dendritic spike initiated with locally spatially and temporally activated synapses. Here, we investigate a proposed idea that dendrites can perform temporal integration on behavior timescale ~1s, thus weakening simultaneous activation constraint. We construct the model of the recurrent neural network where each neuron activates not as a weighted summation of inputs, but due to their coincident activation both in space and time. We show that with using sparse distributed representation and tracking activity of the network in a certain time window it is possible to achieve a high capacity prediction system. We perform the theoretical analysis and estimate the capacity for the different parameters of the model where even the network with 100 neurons can store millions of sequences. Such a capacity results in a biologically unrealistic high number of synapses, much more than 100×100. However, this mechanism of tracking space-time coincidences in sparse activation can be realized in a limited biological neural network but still with a good sequence transition memory.

Keywords: sequence prediction, dendritic nonlinearity, association memory

INTRODUCTION

Observing sequential activation of neurons in response to temporally structured input lead to a recognition that temporal sequence learning is a fundamental computation performed by a brain [1]. There were a lot of models that tried to implement this computation, however, there is still no single working system that could do it as efficiently as the brain [2]. The main problem is that we do not fully understand all computational and biological details and how information should be represented and linked together. On high-level reasoning, it is clear that neural tissue somehow creates associations between events that are spread in time by connecting neural populations. Later, if initial events reappear the network can predict the next outcome and initiate suitable decisions. But, on the low detailed level, many unresolved questions arise, like how the events are encoded or how exactly associations in time are formed.

Here we investigate sequence prediction problem as one of the problems of sequence learning [3]. We take inspiration from the recent findings on a dendritic computation that each individual branch can work as a coincidence detector [4–7]. This is a form of spatial integration where the correct combination of simultaneously active neurons can activate other neurons. Also, temporal integration by dendrites was shown in [8], and later it was hypothesized that the time of integration can reach to behavioral time scale ~1s [9]. Thus, we weaken constrain of simultaneous activation and construct the model of the recurrent neural network where each neuron works like a multiple coincidences detector that learns

both spatial and temporal activation patterns. Events of the sequence represented as sparse binary vectors in discrete time steps. Each neuron learns a fingerprint of a sequence on last T time steps by storing labels of a small number of active neurons at different times into a dedicated dendritic branch or a cluster [10, 11]. If the incoming pattern has all active neurons that are stored in any of the clusters the neuron becomes active. From a single fingerprint, it is hard to deduce the whole sequence but is very easy with distributed representation where multiple fingerprints are stored across the population. This approach is similar to time delay neural networks, where the context for prediction is set by a recent sequence history [26], with the distinction of a neural model in the core.

We perform a theoretical analysis of the proposed model and show that it has a big capacity of sequence transitions thus it can reliably predict the next element. By feeding prediction as an input it is possible to predict the whole sequence or to generate the best guess. To make the model more biological plausible in sense of a number of synapses per neuron we reduced the possible number of connections for each cluster that serves as a fingerprint. For the network with size 1000 and for 3 synapses per cluster, the total capacity is 10^5 transitions with 4000 synapses per neuron. Notable, that the number of synapses is larger than the number of neurons since two neurons can connect with multiple synapses that belong to different clusters. Also, we discuss the future extension of the model to incorporate probabilities of events and the capability of generalization that comes from sparse distributed representation.

MODEL DESCRIPTION

Biological motivation

Beautiful experiment [8] showed that ordered input is spreading from the tip of a dendrite toward the soma elicit more activation than in the opposite direction. The authors showed that direction selectivity presents in the real neuron due to the nonlinear activation of NMDA receptors and higher impedance with higher distance from the soma. Therefore, the sequential activation of dendrite that starts further from the center depolarize the neuron larger. This experiment suggests that neuron can perform more complicated computations than it was though before, namely encode spatiotemporal sequences. Furthermore, according to theoretical calculations [9], dendrites can detect and differentiate sequences on a behavioral time-scale ≈ 1 second. This is in a good agreement with recent discoveries of long eligibility traces found in a cortex [12]. Activation of a dendritic branch span prolonged time and serves as the basis for further temporal integration.

Further evidence toward extending the time of temporal integration by a single neuron comes from recent experiment measured the receptive field of neurons in auditory cortex of ferrets [13]. On Fig. 1, *a* presented an example of one of the fields. Red dots represent excitatory and blue inhibitory weights. The most important information from the picture is that receptive field is spread in time, it is very localized to specific frequencies and it is sparse, that means that only small portion of frequencies determines neuron output. The authors showed that the similar receptive fields are formed in an artificial neural network optimized to predict the next elements. We take these results into account especially the sparseness of receptive field that spread in time.

Formulation of the task

The task for sequence prediction can be formulated as follows [3]:

$$\begin{cases} x_{j+1} = f(x_j, x_{j-1}, \ldots x_0, w), \\ w = g(x_j, x_{j-1}, \ldots x_0). \end{cases}$$

Where $x_j \equiv x(t_j),\ j = 0:T$ is a state of the network at the time t_j that encodes the input, and w are parameters of the model. We need to find the activation function $f(\cdot)$ and the learning rule $g(\cdot)$. In general, the next element can depend on all previous elements. This is analogous to sequence completion task.

The model

To solve this task, we constructed a recurrent neural network with lateral and feedforward connections. Feedforward determine the state of the network and encode at every discrete time step incoming input $I(t_j) \equiv I_j$ into $x(t_j) \equiv x_j$ that has a active neurons. So, the state of the network is described as the binary vector of size N, $x_j = E(I_j)$ where $E(\cdot)$ is an encoding function. We use as encoding a random projection $E(z) = kWTA(\sum_j w_j z_j)$ where $\{w_j\}$ — random binary weights and the function $kWTA(\cdot)$ returns $a << N$ the most active cells. The resulting encoding is a sparse binary vector, for example, $x_j = [0000001000000000001000000000000000000000000000100]$. Recently, it was shown that such encoding function is used in fruit fly and allows to preserve similarity like in local sensitive hashing [14].

Lateral connections are modulatory and in the absence of an input determine a prediction of the next input. To model lateral activation and learning we used ideas of dendritic spatiotemporal integration, so it is important how and where neurons are connected to a dendritic tree. Lateral activation is defined as follows:

$$x_s^j = \theta\left(\sum_m \prod_{(i,k)\in G_m^j} x_k^i\right).$$

Where $\theta(\cdot)$ is a step function and G_m^j m-th cluster of j neuron stores pairs of indices (i,k) — #neuron, time of activation.

In this model, clusters are parameters instead of usual weights. For a specific neuron to be active, it is necessary that all input neurons, that are stored in one of the clusters are active. Thus, the neuron learns features in the input stream and their sequential order. If a feature reappears, the neuron becomes active and predict the next state of the network. On Fig. 1, *b* presented a cartoon for a learning lateral connections. Filled circles represent active neurons at a specific time, empty circles show the prediction. Note that connections can be made to the same neurons, but track activation at a distinct time. For clarity, connections to only two neurons are presented. It depicts the case with $k = 2$ and $T = 3$, that means that history spans to three time steps and at each time step connection to two neurons are created.

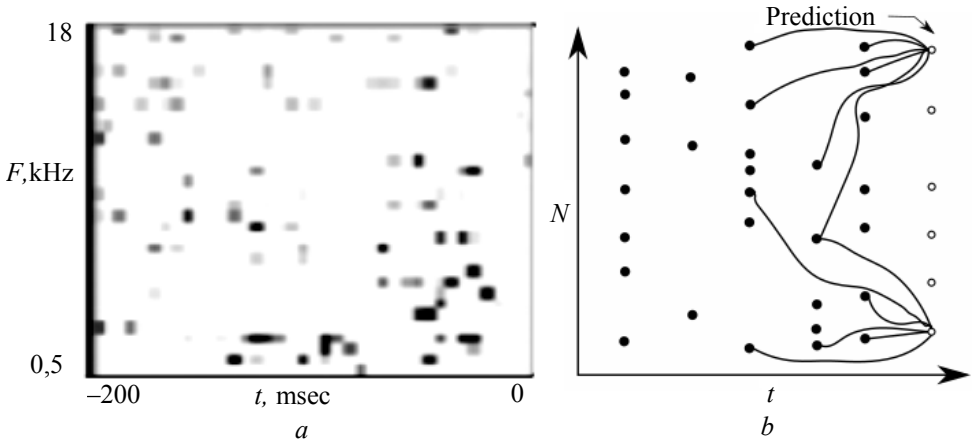


Fig. 1. Spatiotemporal receptive fields of ferrets A1, where individual neuron is very picky for specific frequencies and time (*a*); illustration of an idea of connection through time (*b*)

The presented model differs from the traditional that compute activation based on a previous state of the network since we track activation from more distant moments of time explicitly. As well, it is different from the standard recurrent neural network, since it uses binary neurons, non-differentiable activation function and unsupervised learning procedure of creating clusters. Furthermore, it is more biologically plausible since uses local learning rule. Similar ideas were presented earlier in [15]; however, we use different activation function and learning rule, that allowed us to gain much higher memory capacity.

RESULTS

Derivation of transition memory capacity

Here we present the theoretical calculation of a maximum number of predictions the network can reliably make. The size of the network is N, the number of active neurons that encode an event is a, sparsity $s = \frac{a}{N} << 1$, the number of synapses each neuron form for each event is k, and the number of previous time steps that influence activation of a neuron is T. This task is equivalent to forming an association with network size $N' = (C_N^k)^T$ and activation $a' = (C_a^k)^T$, where C_a^k is a binomial coefficient. To define capacity we create virtual weights w with size $[N, N']$ with learning as $w^i = x^i f(\vec{x})$ where $f(\vec{x})$ returns one randomly selected cell among a' in the network N'. Weights load for a neuron i is a relation of a number of nonzero weights to total size $s_w^i = \frac{|w^i|}{N'}$. Each learning event increases load for active neurons as $s_w = s_w + s_0$, where $s_0 = \frac{1}{N'}$. After the presentation of R events, the load is:

$$s_w = 1 - (1 - s_0)^{Rs}.$$

The probability of false prediction is given by:

$$p(\tilde{x}_m^i = 1 \mid x_m^i = 0) \equiv p = 1 - (1 - s_w)^{a'}(1 - s),$$

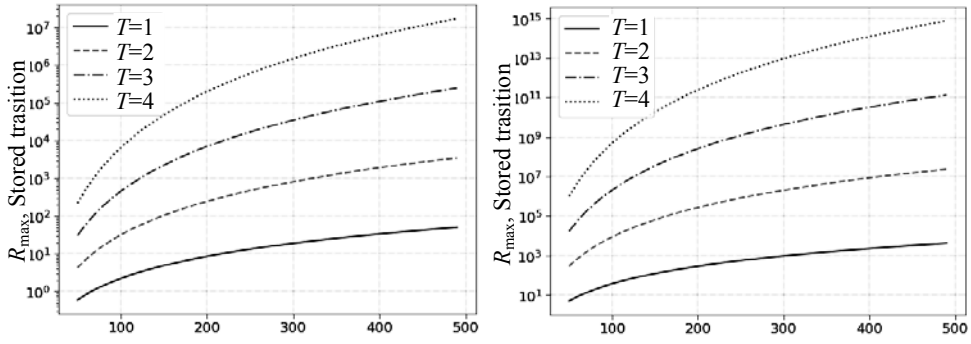
where $\tilde{x}_m^i$ prediction for neuron *i* for time *m*. We can set the fidelity parameter $\varepsilon = 0{,}01$ that determines how many mistakes can be tolerated so that less than 1% of cells could be falsely active. From this fidelity constraint, we can calculate $R_{\max}$. Taking the limit case $s_0 \to 0$ we can derive

$$R_{\max} = \frac{\varepsilon}{s(1-s)s'},$$

where $s' = \frac{a'}{N'}$. From this theoretical analysis, we have important conclusion that for the sparsely active network, detection of coincidence through time enlarges the neural dimension where it is easier to separate patterns. The maximum number of sequence transitions depend inversely to the sparsity of a network activation. Increasing dimension leads to increasing sparsity thus to increasing $R_{\max}$. The limit case of one active cell has the highest sparsity, but it uses the local code so that the network can have only *N* possible states. Lower bound on sparsity set the combinatorial term $C_N^a > R_{\max}$.

Investigating the capacity for different parameters

We investigated how the capacity of the model depends on parameters. On Fig 2 it is shown that the longer the history time *T* the neuron can access, the higher the capacity. Also, the more neurons from one moment of time connected into a cluster, the larger the capacity. This can be intuitively explained because the coincidence of even two neurons is much rarer than activation of a single one. Therefore, the coincident activation of several neurons serves as a fingerprint of an activity pattern. Interestingly, for some combination of parameters, the neural network can recognize and predict 10^{10} transitions. Such huge number requires a huge number of synapses and is practically and biologically unrealistic.



a *b*

Fig. 2. The capacity of sequence transition memory for different parameters. Capacity with $k = 1$, so that one synapse is created per event; capacity for larger networks reaches millions of transitions (*a*); capacity with $k = 2$. In this case, neuron much better recognizes events, thus it can remember the much larger number of transitions (*b*)

Model modifications

Proposed model presents a basic and straightforward way to track activation in time. To make it more biologically realistic we propose two modifications.

The first one treats the problem of sequences confusion. The model assumes that if two sequences are the same on interval T and differ at a current time step, then the model cannot correctly predict the next element. For example *ABCD* and *ABCR* with $T = 3$, the system will confuse D and R. This can be fixed by adding auto-associative connections to populations that encode D or R, that will track the strength of frequency of an event. Thus, the strength of interconnections of population encodes its probability $p(x_i)$. In this case, the task will be formulated in terms of probabilities of predicting the correct event $(p(x_i \mid x_{i-1}, x_{i-2} \ldots x_{i-T}))$. The correct prediction is selected as follows: $x_j = \underset{x_j}{\operatorname{argmax}}(p(x_j \mid x_{j-1}, x_{j-2} \ldots x_{i-T}))$. There is no complete understanding how the probabilities are represented in the biological neural network and experimentation with different algorithms is a good direction for the future.

The second modification proposes to remove a large portion of connections. It decreases the capacity of transitions, but because the connections are still dispersed in space and time the system still has a good capacity. On Fig 3, *b* presented the depiction of the idea, where each cell has a fixed number of connections with other neurons. By linking events at different times, predictive cells can represent different conditional probabilities, like $p(x_{10} \mid x_{9,} x_7)$ or $p(x_{10} \mid x_{9,} x_5)$. This allows reusing subpopulations to represent other sequences. In case of limited connections, the enlarged dimension is the following $N' = C_N^k C_{N(T-1)}^m$ with activation $a' = C_a^k C_{a(T-1)}^m$ where k is the number of connections to previous time step activation and m is the number of connections to other $T - 2$ steps of activation.

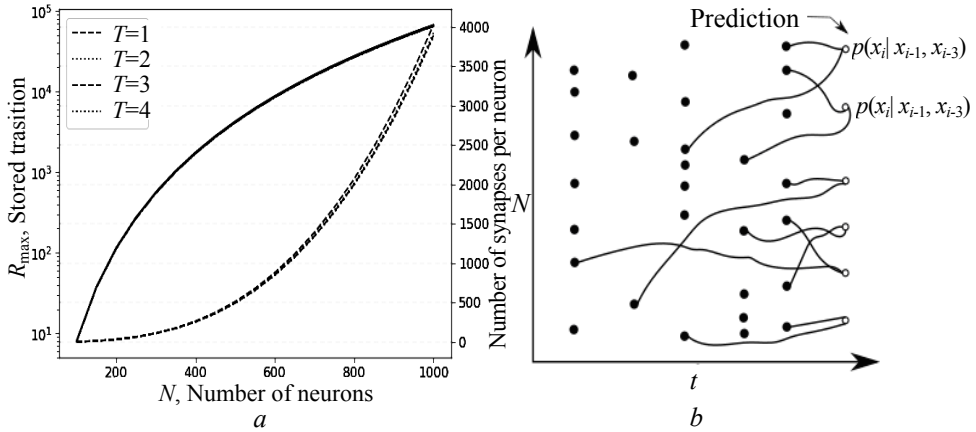


Fig. 3. A maximum number of sequence transitions for different network sizes and time windows, shown as an upper line (*a*); an illustration of an idea of fixed sparse number of connections into past activity and resulted in different joint distributions for subpopulations (*b*)

On the Fig. 3, *a* presented results for transition capacity for different network sizes for different time windows with $k = 1$, $m = 2$, and $a = 20$ — mean number

of active cells. On the right axis and with dotted curves presents the number of synapses per neuron. The capacity almost does not depend on time window and increases with network size. For $N = 100$ there are 4000 synapses per neuron which is biologically plausible, and the number of transitions is 100000. If to take sequences as words with an average number of symbols 7, there are near 14000 possible words encoded into the network.

Still, these modifications are in the early stage of investigations. Results, on real datasets and common benchmarks like TIMIT or Penn tree bank (PNB), should be obtained in the further research. However, it is not expected that these model will be able to compete with the state-of-the-art supervised systems since the main purpose of the model to propose the possible way how sequence prediction can be achieved in the brain and use unsupervised learning procedure.

DISCUSSION AND CONCLUSIONS

The importance of sparse distributed representation

Benefits of presented model rely on the sparse distributed representation of inputs [16, 17]. The sparsity of neural activation gives a high capacity of memory, possibility to express probabilities and generalization. Theoretical formula (1) on a maximum number of stored transition contains sparsity in the denominator, so the high sparsity results in higher memory capacity.

As it was noted by Barlow, the brain should somehow encode the probabilities of events [18], so that to be able to apply similar processing as a Bayesian inference [19, 20]. With this, it is possible to make many predictions and to select the most probable one to update current beliefs. Sparsity is crucial for representing probabilities since the dense representation has many active neurons and activation patterns have high overlap that blurs feature probabilities.

Another important topic is a generalization, that means an ability to make predictions not only for previously experienced sequences but for the new ones as well. With sparse activity, similar inputs are encoded with similar representation and their intersection encodes shared features. These common neurons at the intersection are activated more often and have higher chance to make connections and participate in prediction. For example, if the words are elements of a sequence, then sentences with similar worlds will be encoded with similar representations. The new sentence will be encoded with a similar pattern to similar sentences, and the network will try to make the correct prediction based on the previously learned sentence, in the context of the new one. Generalization comes from limited resources, otherwise, synapses could be connected not just to intersecting neurons, but to every pattern and population that encodes general features would be less significant. This idea is promising on a high-level consideration but needs to be implemented carefully with all the details elsewhere.

It worth to note that by assigning active neurons into clusters was made for simplicity and theoretical investigation. The real biological process should include structural plasticity of placing the synapses at specific dendritic locations. With this placement neuron stores, additional information and is able to recognize just the right combination of incoming inputs.

Previous experiments showed that spiking neural network with Hebbian plasticity rules, like STDP, is able to perform sequence prediction [21–23]. However, the achieved capacity relative to computational resources is too low for

practical implementation. Also, a similar model was proposed by Numenta team [24, 25]. They also use the neuron with many dendrites that act as a coincident detector and rely on sparse distributed representation. The main distinction is that presented model does not need the special columnar structure of the network and prediction depends not only on the current state of the network but on many previous. Most importantly, in our model similar sequences are encoded similarly that potentially enables to make basic unsupervised learning like clustering.

Overall conclusion

In this work, we presented a model of a recurrent neural network that makes sequence prediction. At learning phase neuron stores references to a small subset of active neurons at previous times into a dendritic cluster, that server as a fingerprint of a sequence. Activation occurs in case of matching the learned fingerprint with the incoming input. Importantly, a single cell does not store the full sequence, just some elements of it. This and sparse distributed representation of a sequence across the whole population enables to achieve the high capacity of sequence transition memory. Relatively small neural network with 1000 neurons can store millions of sequences and make a reliable prediction.

We captured only minimal biological details, namely multiple coincidence detections through time, but other significant elements are missing, for example, auto-associative connections that are thought to represent probabilities or realistic structural plasticity rules. The next natural extension of the model should be an adjusting for limited resources and encoding of probability distributions in the inner connectivity.

The idea that the single neuron can learn multiple spatiotemporal sequences on a behavioral time scale still needs more experimental verification. From our theoretical analysis, we can see that this yet hypothetical idea leads to a much greater computational power of a biological neuron and the network in general.

Overall, we showed that the model, inspired by recent experimental findings from dendritic computation, provides a high capacity of sequence memory and gives high accuracy for a sequence prediction.

REFERENCES

1. *Clegg B.A.* Sequence learning / B.A. Clegg, G.J. DiGirolamo, S.W. Keele // Trends Cogn. Sci. — 1998. — Vol. 2, N 8. — P. 275–281.
2. *Bhalla U.S.* Dendrites, Deep Learning, and Sequences in the Hippocampus / U.S. Bhalla // Hippocampus. — 2017. — Vol. 2014, N 6.
3. *Sun R.* Sequence learning: from recognition and prediction to sequential decision making / R. Sun, C.L. Giles // IEEE Intell. Syst. — 2001. — Vol. 16, N 4. — P. 67–70.
4. *London M.* Dendritic Computation / M. London, M. Häusser // Annu. Rev. Neurosci. — 2005. — Vol. 28, N 1. — P. 503–532.
5. *Branco T.* The single dendritic branch as a fundamental functional unit in the nervous system / T. Branco, M. Häusser // Curr. Opin. Neurobiol. — 2010. — Vol. 20, N 4. — P. 494–502.
6. *Sjöström P.J.* Dendritic Excitability and Synaptic Plasticity / P.J. Sjöström, A. Rancz, A. Roth et al. // Physiol. Rev. — 2008. — Vol. 88, N 2. — P. 769–840.

7. *Kastellakis G.* Synaptic clustering within dendrites: An emerging theory of memory formation / G. Kastellakis, D.J. Cai, S.C. Mednick et al. // Prog. Neurobiol. — 2015. — Vol. 126. — P. 19–35.
8. *Branco T.* Dendritic discrimination of temporal input sequences in cortical neurons / T. Branco, B.A. Clark, M. Häusser // Science. — 2010. — Vol. 329, N 5999. — P. 1671–1675.
9. *Bhalla U.S.* Synaptic input sequence discrimination on behavioral timescales mediated by reaction-diffusion chemistry in dendrites / U.S. Bhalla // Elife. — 2017. — Vol. 6. — P. 1–24.
10. *Kastellakis G.* Linking Memories across Time via Neuronal and Dendritic Overlaps in Model Neurons with Active Dendrites / G. Kastellakis, A.J. Silva, P. Poirazi // Cell Rep. — 2016. — Vol. 17, N 6. — P. 1491–1504.
11. *Frank A.C.* Hotspots of dendritic spine turnover facilitate clustered spine addition and learning and memory / A.C. Frank, S. Huang, M. Zhou et al. // Nat. Commun. — Vol. 9, N 1. — P. 422. — 2018/
12. *He K.* Distinct Eligibility Traces for LTP and LTD in Cortical Synapses / K. He, M. Huertas, S.Z. Hong et al. // Neuron. — 2015. — Vol. 88, N 3. — P. 528–538.
13. *Singer Y.* Sensory cortex is optimised for prediction of future input / Y. Singer, Y. Teramoto, B.D.B. Willmore et al. — 2017.
14. *Dasgupta S.* A neural algorithm for a fundamental computing problem / S. Dasgupta, C.F. Stevens, S. Navlakha // Science. — 2017. — Vol. 358, N 6364. — P. 793–796.
15. *Bose J.* An associative memory for the on-line recognition and prediction of temporal sequences / J. Bose, S.B. Furber, J.L. Shapiro // Proc. Int. Jt. Conf. Neural Networks. — 2005. — Vol. 2. — P. 1223–1228.
16. *Kanerva P.* Sparse Distributed Memory / P. Kanerva. — MIT Press, 1988.
17. *Kanerva P.* Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors / P. Kanerva // Cognit. Comput. — 2009. — Vol. 1, N 2. — P. 139–159.
18. *Barlow H.* Redundancy reduction revisited / H. Barlow // Netw. Comput. Neural Syst. — 2001. — Vol. 12, N 3. — P. 241–253.
19. *Knill D.C.* The Bayesian brain: The role of uncertainty in neural coding and computation / D.C. Knill, A. Pouget // Trends Neurosci. — 2004. — Vol. 27, N 12. — P. 712–719.
20. *Clark A.* Whatever next? Predictive brains, situated agents, and the future of cognitive science / A. Clark // Behav. Brain Sci. — 2013. — Vol. 36, N 3. — P. 181–204.
21. *Song S.* Competitive Hebbian learning through spike-timing-dependent synaptic plasticity / S. Song, K.D. Miller, L.F. Abbott // Nat. Neurosci. — 2000. — Vol. 3, N 9. — P. 919–926.
22. *Brea J.* Matching Recall and Storage in Sequence Learning with Spiking Neural Networks / J. Brea, W. Senn, J.-P. Pfister // J Neurosci. — 2013. — Vol. 33, N 23. — P. 9565–9575.
23. *Izhikevich E.M.* Polychronization: computation with spikes / E.M. Izhikevich // Neural Comput. — 2006. — Vol. 18, N 2. — P. 245–282.
24. *Ahmad S.* How do neurons operate on sparse distributed representations? A mathematical theory of sparsity, neurons and active dendrites / S. Ahmad, J. Hawkins // arXiv. — P. arXiv:1601.00720 [q-NC], Jan. 2016.
25. *Hawkins J.* Why Neurons Have Thousands of Synapses, a Theory of Sequence Memory in Neocortex / J. Hawkins, S. Ahmad // Front. Neural Circuits. — 2016. — Vol. 10, N March. — P. 1–20.
26. *Lang K.J.* The decelopment of the time delay neural network architecture for speech recognition / K.J. Lang, G.E. Hinton // Technical Report CMU-CS-88-152, Carnegie Mellon, 1988.

Reseived 01.10.2018

From the Editorial Board: the article corresponds completely to submitted manuscript.