

RESEARCH ARTICLE | NOVEMBER 03 2016

# On the applicability of STDP-based learning mechanisms to spiking neuron network models

A. Sboev; D. Vlasov; A. Serenko; R. Rybka; I. Moloshnikov



AIP Advances 6, 111305 (2016)  
<https://doi.org/10.1063/1.4967353>



**APL Energy**  
**Latest Articles Online!**  
**Read Now**



## On the applicability of STDP-based learning mechanisms to spiking neuron network models

A. Sboev,<sup>1,a</sup> D. Vlasov,<sup>2</sup> A. Serenko,<sup>3</sup> R. Rybka,<sup>4</sup> and I. Moloshnikov<sup>4</sup>

<sup>1</sup>National Research Centre Kurchatov Institute, MEPhI National Research Nuclear University and Plekhanov Russian University of Economics, Moscow, Russia

<sup>2</sup>MEPhI National Research Nuclear University, Moscow, Russia

<sup>3</sup>National Research Centre Kurchatov Institute, Moscow, Russia and Moscow Institute of Physics and Technology, Moscow, Russia

<sup>4</sup>National Research Centre Kurchatov Institute, Moscow, Russia

(Received 31 July 2016; accepted 21 October 2016; published online 3 November 2016)

The ways to creating practically effective method for spiking neuron networks learning, that would be appropriate for implementing in neuromorphic hardware and at the same time based on the biologically plausible plasticity rules, namely, on STDP, are discussed. The influence of the amount of correlation between input and output spike trains on the learnability by different STDP rules is evaluated. A usability of alternative combined learning schemes, involving artificial and spiking neuron models is demonstrated on the iris benchmark task and on the practical task of gender recognition. © 2016 Author(s). All article content, except where otherwise noted, is licensed under a Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>). [<http://dx.doi.org/10.1063/1.4967353>]

### I. INTRODUCTION

Applying spiking neural networks to the classification task is currently relevant from two points of view. Firstly, a practical supervised learning algorithm of spiking neural networks gives the way for implementing in autonomous neuromorphic hardware with ultra-low power consumption. Secondly, the creation of such algorithm based on biologically plausible plasticity rules may also explain the role of long-term plasticity in the brain.

Addressing the question of creating practically effective supervised learning methods for spiking neuron networks, there is a number of works devoted to it (Gütig and Sompolinsky, 2006; Mitra *et al.*, 2009 and Franosch *et al.*, 2013), but still no such method has been created based only on the current knowledge of biological neural systems operating rules, namely, on spike-timing-dependent plasticity (STDP), a biologically inspired long-term plasticity model. In Section II we discuss, under what conditions the weights with which a neuron performs the desired input-output transformation can in principle be stably reached as the result of STDP. In Section II B 2 we demonstrate that the steady value of a weight is determined by the amount of correlation between the output spike train and the corresponding input spike train. Based on this fact, in Section II C we propose a supervised learning algorithm and show its capability to solve a linear classification task.

There is also a straightforward approach, to map the trained formal network onto the spiking one. In (Eliasmith, 2013) each formal neuron is replaced with several spiking ones, that, along with the encoding and decoding machinery, reproduce its activation function. Furthermore, one can simply transfer synaptic weights obtained by training a formal network to a spiking network of same topology (Diehl *et al.*, 2015). In Section III B we show on the Fisher's Iris benchmark that the spiking network can give an increase in classification accuracy compared to the formal one. In Section III C we apply this approach to the task of recognizing gender of a text author. This task is of great importance in safety and guard systems, and social network analysis, as a component of authorship profiling, i. e.

<sup>a</sup>Electronic mail: [Sboev\\_AG@nrcki.ru](mailto:Sboev_AG@nrcki.ru)

extraction of information about the unknown author of text (demographics, psychological traits, etc.), based on the analysis of linguistic parameters.

## II. STDP-BASED APPROACH

### A. Materials and methods

In the Spike-Timing-Dependent Plasticity model (Morrison et al., 2008), the strength of a synapse is characterized by weight  $0 \leq w \leq w_{\max}$ , whose change depends on the difference between presynaptic  $t_{\text{pre}}$  and postsynaptic  $t_{\text{post}}$  spike moments:

$$\Delta w = \begin{cases} -W_- \cdot \left(\frac{w}{w_{\max}}\right)^{\mu_-} \cdot \exp\left(-\frac{t_{\text{pre}} - t_{\text{post}}}{\tau_-}\right), & \text{if } t_{\text{pre}} - t_{\text{post}} > 0; \\ W_+ \cdot \left(1 - \frac{w}{w_{\max}}\right)^{\mu_+} \cdot \exp\left(-\frac{t_{\text{post}} - t_{\text{pre}}}{\tau_+}\right), & \text{if } t_{\text{pre}} - t_{\text{post}} < 0. \end{cases} \quad (1)$$

where  $W_+ = 0.03$ ,  $W_- = 1.035 \cdot W_+$ ,  $\tau_+ = \tau_- = \tau = 20$  ms. The rule with  $\mu_+ = \mu_- = 0$  is called additive STDP, with  $\mu_+ = \mu_- = 1$  – multiplicative, intermediate values  $0 \leq \mu \leq 1$  are also possible.

In case of additive STDP the auxiliary clauses are added to prevent the weight from falling below zero or exceeding the maximum value  $w_{\max}$ : if  $w + \Delta w > w_{\max}$ ,  $w \rightarrow w_{\max}$ ; if  $w + \Delta w < 0$ ,  $w \rightarrow 0$ .

An important part of STDP rule is the scheme of pairing pre- and postsynaptic spikes when evaluating weight change according to the rule (1). Besides the all-to-all scheme, there exist several nearest-neighbour ones (Morrison et al., 2008). We used the restricted symmetric scheme (fig. 1), in which a presynaptic spike is paired with the last preceding postsynaptic, if it has not yet been accounted in a pre-after-post pair, and vice versa: a postsynaptic spike is paired with the nearest preceding presynaptic if the latter has not yet participated in any other post-after-pre pair.

As the neuron model we used Leaky Integrate-and-Fire, in which the membrane potential  $V$  obeys

$$\frac{dV}{dt} = \frac{-(V(t) - V_{\text{resting}})}{\tau_m} + \frac{I_{\text{syn}}(t)}{C_m} + \frac{I_{\text{ext}}}{C_m};$$

when  $V \geq V_{\text{th}}$ ,  $V \rightarrow V_{\text{reset}}$ , and during the refractory period  $\tau_{\text{ref}} = 3$  ms the neuron is insensitive to the synaptic input. Membrane capacity  $C_m = 300$  pF, membrane time constant  $\tau_m = 10$  ms. The postsynaptic current is of exponential form: a presynaptic spike from synapse  $i$  at time  $t_{\text{sp}}$  adds  $w_i(t_{\text{sp}}) \frac{q_{\text{syn}}}{\tau_{\text{syn}}} e^{-\frac{t-t_{\text{sp}}}{\tau_{\text{syn}}}} \theta(t-t_{\text{sp}})$  to  $I_{\text{syn}}$ , where  $q_{\text{syn}} = 0.03$  pC,  $\tau_{\text{syn}} = 5$  ms,  $w_i$  is the synaptic weight and  $\Theta(t)$  is the Heaviside step function.

### B. The principal possibility of weight to converge to the target on base of input-output correlation

Provided that the desired synaptic weights are known in advance, the question is whether such weights can emerge as the result of applying STDP to input spike trains and some output spike train. Following (Legenstein et al., 2005), we investigated the ability of weights to converge to the target under the following protocol:

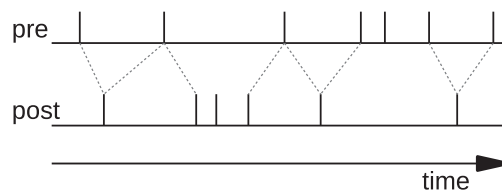


FIG. 1. The restricted symmetric spike pairing scheme. Tics denote spikes, and a dashed line means taking that pair of spikes into account in the STDP weight change rule, potentiation in pre-before-post case and depression in post-before-pre case.

1. The output train of the neuron with target weights and without STDP is recorded. It is then considered as the desired output.
2. STDP is turned on, and the neuron, receiving the same input trains, is forced to fire spikes in desired moments. This is expected to make the weights converge to the target, starting from arbitrary (but low) initial ones.

From our preliminary works the following is known: The weights can converge to the target not with all spike pairing schemes. The restricted symmetric scheme showed the best convergence, so it is the one we use.

The existence of short-term plasticity in the synapse model does not affect the weights convergence. The neuron model also does not matter: we checked Leaky Integrate-and-Fire, [Hodgkin and Huxley, 1952](#) and [Izhikevich, 2003](#) models and a static adder, in which an incoming spike just adds its synapse' weight to the membrane potential, and when the accumulated value reaches threshold, it is dropped to zero and the neuron fires a spike.

The convergence persists if mean frequencies of input spike trains are changed during the simulation, but slightly declines in case different inputs receive trains of different mean frequencies. ([Sboev et al., 2016a](#)).

Finally, the considered protocol of forcing the output leads to weights convergence to target in case a set of binary vectors is used as input, and weights with which the neuron divides the set into two classes are used as target ([Sboev et al., 2016b](#)). Though this result does not provide the learning mechanism to obtain weights needed for a classification task, it shows that such weights can be stably reached as the result of STDP.

### 1. Weights convergence in case of non-additive STDP

In case of additive STDP only 0 and  $w_{\max}$  are stable points of weight. Using non-additive STDP allows more wide range of weight distributions to be reached with the protocol under consideration. Fig. 2 shows two examples of target weight distributions that can be reached with the parameters that we found,  $\mu_+ = 0.06$  and  $\mu_- = 0.01$ .

### 2. Correlative nature of STDP

We now demonstrate that the steady value of a weight is determined by the amount of correlation between input and output spike trains.

a. *The correlation measure.* The normed cross-correlation function is defined as

$$\Gamma(\Delta t) = \frac{1}{\sum_k S_{\text{pre}}(kt_{\text{bin}}) \cdot \sum_k S_{\text{post}}(kt_{\text{bin}})} \cdot \sum_k S_{\text{pre}}(k \cdot t_{\text{bin}}) S_{\text{post}}(k \cdot t_{\text{bin}} + \Delta t), \quad (2)$$

where  $S_{\text{pre/post}}(t)$  indicates a pre/postsynaptic spike respectively at time  $t$ , and  $t_{\text{bin}}$  is the simulation step,

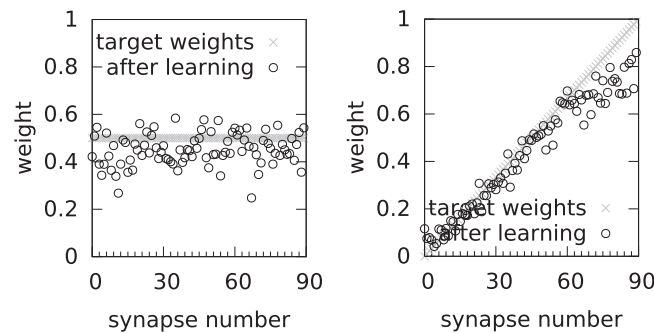


FIG. 2. Target synaptic weights and weights reached after applying the force-output protocol described in Section II B. In the left plot target weights are all equal to 0.5, and in the right plot target weights are distributed randomly between 0 and 1.

$$I = \sum_{\Delta t=0}^{\tau} \Gamma(\Delta t)$$

can be used as a rough correlation indicator, where  $\tau$  is the STDP time window constant. A similar estimate is often used in analytical studies such as (Rossum et al., 2000).

*b. Results.* We here artificially generated, based on the technique from (Gütig et al., 2003), input and output trains with different values of correlation. Applying STDP to these trains leads to weights convergence, and the resulting weight value depends monotonously on the input-output correlation (“artificial output” points in Fig. 3). The established weights, in their turn, reproduce the signal with the same level of correlation as the initial artificial signal (“neuron output” points in Fig. 3). STDP here was non-additive, with  $\mu_+ = 0.06$  and  $\mu_- = 0.01$ .

So, any desired weight value can be reached by making the neuron generate output with the proper amount of correlation with the corresponding input. Based on this fact, we suggest the following protocol of supervised learning.

### C. A learning algorithm based on controlling input-output correlation

Our model now consists of a single neuron with 100 incoming synapses, all excitatory. As the input data we use 10-dimensional binary vectors, having half components of 0 and the other half of 1. Each vector component of 1 is encoded by 10 synapses of the neuron receiving independent Poisson trains with mean frequency of 20 Hz, a component of 0 – by 10 independent 2-Hz trains. Let each vector belong to one of two classes,  $C_+$  and  $C_-$ , and let the task be that the neuron should produce high mean firing rate in response to vectors from  $C_+$  and the lowest possible mean firing rate in response to vectors from  $C_-$ .

STDP weight change constants are chosen to be  $W_+ = 0.01$ ,  $W_- = 1.035W_+$ . Initially all weights are set to 0.4.

*The learning protocol* is the following. Input vectors are presented to the neuron in an alternating manner: a vector from  $C_+$  during 5 s, then a vector from  $C_-$  for 1.5 s. During the presentation of a vector from  $C_-$  the neuron is stimulated with constant current, high enough to make the mean output rate close to the highest possible due to refractoriness,  $1/\tau_{\text{ref}}$ .

#### 1. Results

While the neuron is receiving an input vector from  $C_+$  class, a synapse receiving high-frequency input contributes more to the neuron’s output, therefore its weight is more rewarded by STDP. As a result, weights of synapses receiving vector components of 1 increase in 66% cases, and weights of synapses receiving components of 0 decrease in 66% cases with the parameters we have chosen. When a vector from  $C_-$  class is presented, the neuron output is caused by the stimulating current and

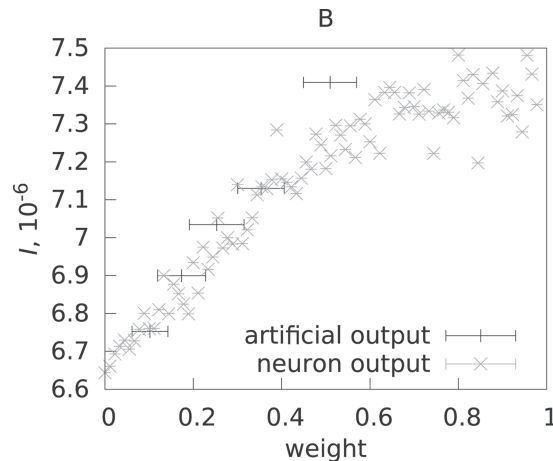


FIG. 3. The correlation indicator  $I$  in dependence of the weight that established by applying STDP to inputs and output with that correlation (“artificial output”), and  $I$  of the output that the neuron produces with the established weight (“neuron output”).

is poorly correlated with input. So, all weights decrease (for them not to fall to zero the duration of a vector from  $C_-$  is 1.5 s in contrast to 5 s of a vector from  $C_+$ ), but weights of high-frequency inputs decrease more due to higher number of post-before-pre events.

To assess the ability of the algorithm to solve a classification task, we took six binary vectors:

$$\begin{aligned} S_1 &= (1\ 1\ 1\ 1\ 0\ 0\ 0\ 0\ 1\ 0), \\ S_2 &= (0\ 1\ 0\ 0\ 1\ 1\ 1\ 0\ 1\ 0), \\ S_3 &= (1\ 0\ 1\ 0\ 1\ 1\ 1\ 0\ 0\ 1), \\ S_4 &= (1\ 1\ 0\ 0\ 0\ 1\ 0\ 1\ 0\ 1), \\ S_5 &= (0\ 1\ 1\ 0\ 1\ 0\ 0\ 1\ 0\ 1), \\ S_6 &= (0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 0); \end{aligned}$$

three of which are linearly separable from the other three. The desired weights which separate them are known (each digit below corresponds to 10 synapses having the same target weight):

$$\vec{w}_{\text{target}} = (1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0),$$

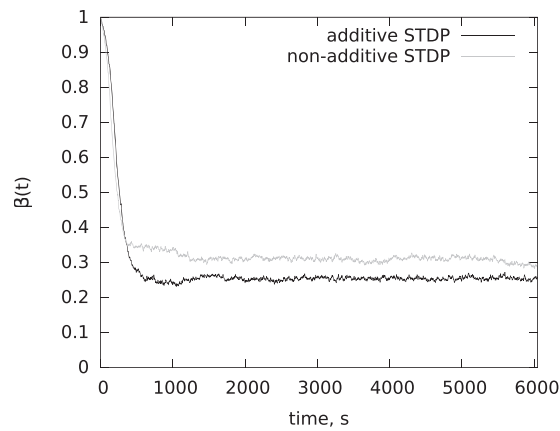


FIG. 4. Deviation  $\beta$  between actual and target weights during learning with additive STDP and with non-additive ( $\mu_+ = 0.06$ ,  $\mu_- = 0.01$ ).

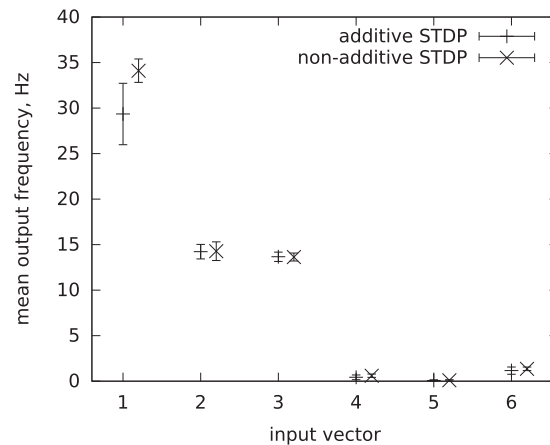


FIG. 5. Mean firing rate of the neuron in response to the input vectors after learning with additive and non-additive STDP. The first three vectors belong to  $C_+$  class, and the second three to the  $C_-$  class. Firing rate was averaged over 5 tries, each having independent 30-s input spike trains.

so learning performance can be characterized by the deviation

$$\beta(t) = \frac{\sum_{i=1}^{100} |w^i(t) - w_{\text{target}}^i|}{\sum_{i=1}^{100} w_{\text{target}}^i}$$

between actual and target weights during learning (Fig. 4). After 6,045 s of learning (310 cycles of presenting the whole set of vectors) weights converge to bimodal stationary distribution, i. e. each weight tends to either 0 or 1. Not all weights converge to target due to probabilistic nature of input spike trains. However, this effect is averaged out thanks to excessive number of synapses, 10 per each input vector component, and after learning the neuron clearly distinguishes the classes by its mean firing rate, as shown in Fig. 5. Note that weights convergence and classification distinctness are nearly equal for additive STDP and non-additive (with  $\mu_+ = 0.06$  and  $\mu_- = 0.01$ ).

### III. ANN TO SNN MAPPING APPROACH

#### A. Network parameters and learning algorithm

We here used, following (Diehl et al., 2015), the combined learning algorithm, involving artificial (ANN) and spiking neural networks (SNN). It consists of the following steps (fig. 6):

1. *Training the artificial neural network by backpropagation.* The ANN neurons' activation function was ReLU for hidden layers and Softmax for the output layer. Neuron biases were set to zero. Input data were normalized so that the Euclidian norm of each input vector equaled 1.
2. *Mapping the synaptic weights to the spiking neural network.* In the SNN we here used non-leaky Integrate-and-fire neuron without refractoriness, in which a dimensionless membrane potential obeys  $\frac{dV}{dt} = 1/\tau \sum_i \sum_{s \in S_i} w_i \delta(t - s)$ , where  $S_i$  is spike train on  $i$ -th input synapse,  $w_i$  is synaptic weight, and  $\tau = 0.01$  ms. Reaching the threshold  $\Theta$ , the potential is reset to zero.
3. *Encoding input data to spike trains.* An input vector component  $x$  was encoded by a Poisson spike train with mean frequency  $x \cdot \nu_{\text{max}}$ .
4. *Optimizing the spiking network parameters.* Besides  $\nu_{\text{max}}$  and  $\Theta$ , simulation time  $T$  and simulation step  $\Delta t$  were adjusted. According to (Diehl et al., 2015), there are two necessary conditions to eliminate accuracy losses after transfer:
  - the simulation time should be long enough to exclude probabilistic influence of spike trains;
  - the neuron should not have to fire several spikes in one simulation step. So, total input a neuron receives during one simulation step must not exceed the threshold:

$$\nu_{\text{max}} \cdot \Delta t \cdot \sum_i w_i \leq \Theta. \quad (3)$$

To fulfill (3) all spiking neural network weights are divided by the normalization factor  $M$ , same for all neurons in a layer but unique for each layer,

$$M = \frac{1}{\Theta} \max_j \left( \sum_j w_{ij} \right), \quad (4)$$

where  $w_{ij}$  is  $i$ -th synapse weight of  $j$ -th neuron in current layer. Note that this assumes that no more than one spike can arrive from one input in one timestep.

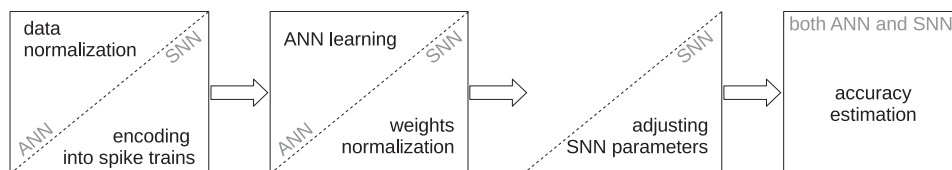


FIG. 6. The correspondence between artificial network (ANN) designing steps to the ones for spiking (SNN).

The conditions above are necessary but not sufficient, so achieving maximal classification accuracy still requires adjusting  $v_{\max}$  and  $\Theta$ .

## B. Fisher's iris classification

To test the algorithm described above the popular toy task of Fisher's iris classification was solved. The network had 4 neurons in the input layer, 4 neurons in the single hidden layer, 3 neurons in the output layer. Spiking network weights were normalized according to (4). Each input vector was presented during 1 s. The simulation step was chosen to be 0.1 ms (decreasing it to 0.01 ms did not affect the results). Classification result was determined according to the output neuron that had fired the most of spikes during the simulation.

### 1. Results

The ReLU network gave the classification error (ratio of wrongly classified input samples to the total number of samples) of  $0.04 \pm 0.01$  on the training sets and  $0.06 \pm 0.04$  on the testing sets, averaged over 10 different divisions of input data into training and testing sets. Mean classification error on test set of spiking network with different  $\Theta$  and  $v_{\max}$  is shown in Fig. 7. The highest classification accuracy on spiking network, error of  $0.04 \pm 0.01$ , was obtained at high input frequencies and thresholds. Increase in both frequency and threshold, keeping (3), increases the amount of input spikes a neuron has to integrate before it fires a spike itself, and therefore increases classification accuracy. However, increasing  $v_{\max}$  over  $1/\delta t$  worsens the accuracy because breaks the condition of no more than one input spike to a synapse in one timestep.

## C. Gender prediction

We now apply the approach under consideration to the task of recognizing gender of a text author. For this purpose we took a subset of RusPersonality (Zagorovskaya et al., 2012), the first corpus of Russian-language texts labeled with information on their authors: gender, age, psychological testing data, etc. This free-to-use corpus contains over 1,850 documents, 230 words per document in average, from 1,145 respondents. The texts were written by university students, who were given a few themes to describe. The themes were same for male and female participants, so that one can focus on the peculiarities caused by authors' gender rather than by their individual styles.

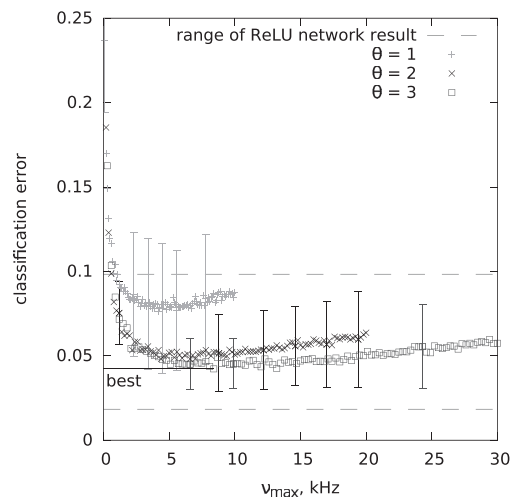


FIG. 7. The dependence of Fisher's iris classification error on maximum input frequency for different neuron thresholds. Each point is averaged over 5 independent realizations of input spike trains, and then over 10 different divisions of data to training and testing sets (deviations shown not for all points for readability).



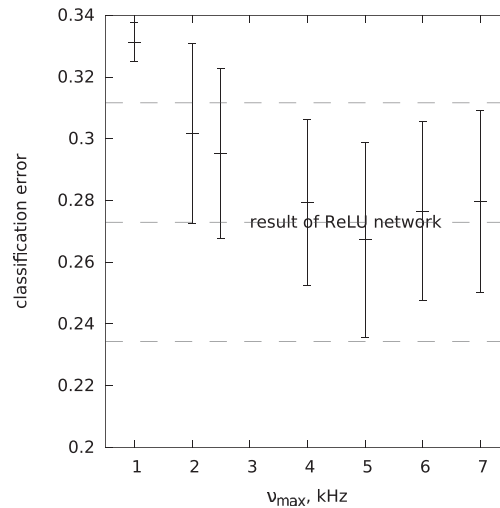


FIG. 8. The gender recognition error in dependence on the maximum input frequency of the spiking network. The neuron threshold equaled 1, the duration of presenting one input vector was  $T = 10$  s. Dashed lines restrict the classification error range of the ReLU network (obtained by splitting into training and testing sets 10 times).

As the input data for the network, each text was described by 141 features:

- numbers of different parts of speech: nouns, numerals, adjectives, prepositions, verbs, pronouns, interjections of adverbs, articles, conjunctions, participles, infinitives, and the number of finite verbs (13 total);
- numbers of syntactic relations defined in the Russian National Corpus (<http://www.ruscorpora.ru/en/>), 60 total;
- different ratios of number of one part of speech to another according to Sboev et al., 2015, 27 total;
- numbers of exclamatory marks, question marks, dots, and emoticons (4 total);
- numbers of words expressing a particular emotion according to “Emotions and feelings in lexicographical parameters: Dictionary of emotive vocabulary of the Russian language” (<http://lexrus.ru/default.aspx?p=2876> [page in Russian]), 37 emotions total.

The training set contained 364 texts, the testing one 187. Network topology was feedforward: 141 input neurons, 81 neurons in the first hidden layer, 19 neurons in the second hidden layer and 2 neurons in the output layer. Weights were normalized according to (4).

## 1. Results

The classification error of ReLU neural network was  $0.01 \pm 0.02$  on the training set and  $0.28 \pm 0.04$  on the testing set. Mean classification error on test set of spiking neural network with  $\Theta = 1$  and different  $\nu_{\max}$  is shown in Fig. 8. Other thresholds are not shown as they provide significantly worse classification accuracy. The lowest classification error is  $0.28 \pm 0.03$ , which equals to the one of ReLU network.

Hereby, the approach under consideration can be applied to the practical machine learning task. However, achieving classification accuracy optimum requires additional adjustment of the spiking network parameters.

## IV. CONCLUSION

Supervised learning can be performed on base of bare spike-timing-dependent plasticity, without any modifications, along with keeping all its parameters in biologically plausible ranges. A solution for the implementation of the teacher is that the neuron is provided with the information on the classes in the form of controlling the correlation between the neuron’s output and the inputs.

There is also a straightforward way to obtain the spiking network for a classification task by training a well-studied artificial network and then using the ready weights in the spiking network. Such a transfer not only allows the network to be implemented in a low-energy-consuming hardware, but also may increase the classification accuracy due to the presence of additional adjustable parameters in a spiking network compared to a formal one.

## ACKNOWLEDGMENTS

This work was supported by RSF project 16-18-10050 “Identifying the Gender and Age of Online Chatters Using Formal Parameters of their Texts”. Simulations were carried out using the NEST Simulator (Gewaltig and Diesmann, 2007) and high-performance computing resources of federal center for collective usage at NRC “Kurchatov Institute”, <http://computing.kiae.ru>.

- Diehl, P. U., Neil, D., Binas, J., Cook, M., Liu, S.-C., and Pfeiffer, M., “Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing,” in IEEE International Joint Conference on Neural Networks (IJCNN), 2015.
- Eliasmith, C., *How to build a brain: A neural architecture for biological cognition* (Oxford University Press, 2013).
- Franosch, J.-M. P., Urban, S., and van Hemmen, J. L., “Supervised spike-timing-dependent plasticity: A spatiotemporal neuronal learning rule for function approximation and decisions,” *Neural computation* **25**, 3113–3130 (2013).
- Gewaltig, M.-O. and Diesmann, M., “Nest (neural simulation tool),” *Scholarpedia* **2**, 1430 (2007).
- Gütig, R., Aharonov, R., Rotter, S., and Sompolinsky, H., “Learning input correlations through nonlinear temporally asymmetric hebbian plasticity,” *The Journal of neuroscience* **23**, 3697–3714 (2003).
- Gütig, R. and Sompolinsky, H., “The tempotron: A neuron that learns spike timing-based decisions,” *Nat. Neurosci.* **9**, 420–428 (2006).
- Hodgkin, A. L. and Huxley, A. F., “A quantitative description of membrane current and its application to conduction and excitation in nerve,” *J. Physiol.* **117**, 500–544 (1952).
- Izhikevich, E. M., “Simple model of spiking neurons,” *IEEE Transactions on Neural Networks* **14** (2003).
- Legenstein, R., Naeger, C., and Maass, W., “What can a neuron learn with spike-timing-dependent plasticity,” *Neural Computation* **17**, 2337–2382 (2005).
- Mitra, S., Fusi, S., and Indiveri, G., “Real-time classification of complex patterns using spike-based learning in neuromorphic vlsi,” *IEEE Transactions on biomedical circuits and systems* **3** (2009).
- Morrison, A., Diesmann, M., and Gerstner, W., “Phenomenological models of synaptic plasticity based on spike timing,” *Biol. Cybern.* **98**, 459–478 (2008).
- Sboev, A., Gudovskikh, D., Rybka, R., and Moloshnikov, I., “A quantitative method of text emotiveness evaluation on base of the psycholinguistic markers founded on morphological features,” in *4th International Young Scientist Conference on Computational Science [Procedia Computer Science* **66**, 307–316 (2015)].
- Sboev, A., Vlasov, D., Serenko, A., Rybka, R., and Moloshnikov, I., “A comparison of learning abilities of spiking networks with different spike timing-dependent plasticity forms,” *Journal of Physics: Conference Series* **681**, 012013 (2016a).
- Sboev, A., Vlasov, D., Serenko, A., Rybka, R., and Moloshnikov, I., “To the question of learnability of a spiking neuron with spike-timing-dependent plasticity in case of complex input signals,” in *Biologically Inspired Cognitive Architectures (BICA) for Young Scientists. Proceedings of the First International Early Research Career Enhancement School (FIERCES 2016)* [Advances in Intelligent Systems and Computing **449**, 205211 (2016b)].
- Van Rossum, M. C., Bi, G. Q., and Turrigiano, G. G., “Stable hebbian learning from spike timing-dependent plasticity,” *The Journal of Neuroscience* **20**, 8812–8821 (2000).
- Zagorovskaya, O., Litvinova, T., and Litvinova, O., “Elektronnyy korpus studentcheskikh esse na russkom yazyke i ego vozmozhnosti dlya sovremennykh gumanitarnykh issledovaniy [electronic corpus of student essays and its applications in modern humanity studies],” *Mir nauki, kul'tury i obrazovaniya* [World of Science, Culture and Education] **3**, 387–9 (2012).