©2019 International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies

International Transaction Journal of Engineering, Management, & Applied Sciences & Technologies

http://TuEngr.com

PAPER ID: 10A16A

NEURAL NETWORKS WITH PSEUDO-RANDOM DISTRIBUTION OF RELATIONSHIPS USING THE EXAMPLE OF MERCURY ELECTROLYZER OPERATION MODE MODELING

Olga Anatolievna Medvedeva [a], **Aleksandr Nikolaevich Ivanov** [b], **Nikolay Danilovich Morozkin** [b], **Svetlana Anatol'evna Mustafina** [b*]

[a] *Institute of Computational Mathematics and Information Technology, Kazan Federal University, RUSSIA*
[b] *Department of Applied Informatics and Numerical Methods, Bashkir State University, RUSSIA.*

ARTICLEINFO

Article history:
Received 29 April 2019
Received in revised form 02 August 2019
Accepted 19 August 2019
Available online 16 September 2019

Keywords:
Neural networks Modeling; Machine learning; Hyperparameters; Electrolysis; Pseudorandom distribution of connections.

ABSTRACT

This article analyzed the applicability of artificial neural networks to solve the problems of physicochemical process modeling using the example of the mercury electrolyzer operation mode used in caustic soda production. This paper also described the basic qualities of the existing neural networks and the ways of their training. The authors propose the solution to the problem of modeling based on the networks with pseudo-random distribution of connections. This paper described the architecture of these networks, three learning algorithms are proposed. The implementation of neural networks with pseudo-random distribution of connections was performed by Python programming language. The article presents the comparative learning results of different networks with different sets of hyperparameters. Also, the determination of the optimal settings of neural networks allows achieving high learning efficiency. The resulting neural network model described the electrolysis process adequately in accordance with the available source data.

1. INTRODUCTION

Neural networks are widely used to solve a wide range of tasks, such as the forecasting of economic indicators, pattern recognition, the modeling of various processes, etc. [1]. The effectiveness of the neural network is directly determined by the quality of its training [2]. The purpose of neural network training is to minimize the calculation error while maintaining the network response in case of source data change [3]. Learning is achieved by changing the weights of the synaptic connections of neurons [4].

*Corresponding author (Svetlana Anatol'evna Mustafina) Email: samustafina@strbsu.ru

Earlier, Verbos [5] described the method of neural network training, the algorithm for backpropagation of error, which is based on neural network error functional reduction using the gradient descent method. The change of neuron weight in the algorithm occurs in the opposite direction to the gradient. This fact requires that the activation function of neuron was differentiable. The error functional, calculated for the output layer neuron training is transmitted to the neurons of the underlying layers in accordance with their contribution to the final discrepancy.

The error function can be represented by

$$E = \hat{y} - y, \tag{1}$$

where $\hat{y}$ and y the true and the calculated value of the output parameter.

Then the weight change value will be written as follows:

$$\Delta w_i = \Delta \cdot a_i^{-1}, \tag{2}$$

- for the output layer:

$$\Delta = E \cdot a'^{(s)}, \tag{3}$$

- for hidden layers:

$$\Delta = \Delta^{+1} \cdot w_i^{+1} \cdot a'^{(s)}, \tag{4}$$

where Δ is the error value for a given neuron; Δ^{+1} – the error value for the neuron of the next layer; $a'^{(s)}$ – the value of the activation function derivative from the accumulator function; a_i^{-1} – the neuron activation value of the previous layer (input value for this neuron); w_i^{+1} – the weight of the synaptic connection between the given neuron and the neuron of the next layer.

The EBP algorithm has been proposed by the author for a multi-layered neural network without using a gradient method. This algorithm consists of two steps. First, fictitious teacher signals for the outputs of each hidden layer unit are algebraically determined by an error backpropagation (EBP) method. Then, the weight parameters are determined by using an orthogonal projection (EBP-OP) method, or an exponentially weighted least squares (EBP-EWLS) method. It is shown that the algorithm is also applicable to an interconnected neural network.

The method of error backpropagation (EBP) remains one of the most common and effective methods of neural network training to this day. However, this method has certain limitations [6]. One of the limitations is associated with the use of gradient descent, namely, the slowdown of learning when the objective function value reaches the “plateau” and the lack of protection against convergence at the local minimum. It is necessary to calculate the values of the activation function derivative for the gradient descent method. However, not all activation functions are smooth and have a nonlinear derivative over the entire domain. In this regard, sigmoidal functions that meet the requirements of nonlinearity and differentiability are used most often as activation functions [7].

It should also be noted that the error back-propagation method can be used exclusively in the networks whose graphs do not contain loops. Such networks are called non-recursive ones. These include, for example, multilayer perceptrons [8].

A neural network with a pseudo-random distribution of connections presents most of the generated graphs with different cycles, and their number directly depends on the degree of this network connectivity. Also, the presence of loops (when the corresponding restriction is removed) as the elementary cycles of unit length leads to the impossibility of this training method used in an unchanged form. Neural networks are typically used to build classification or regression models i.e. they take inputs and predict either a discreet result (classification) or a continuous value (regression). An example of classification would be to take a collection of pixels and predict whether the image represents a given object.

Thus, the issue of appropriate algorithm development arises that are applicable for such network training, as well as those with high learning efficiency in a wide range of network parameters.

A critical issue of Neural Network-based large-scale data mining algorithms is how to speed up their learning algorithms. This problem is particularly challenging for Error Back-Propagation (EBP) algorithm in Multi-Layered Perceptron (MLP) Neural Networks due to their significant applications in many scientific and engineering problems. In this paper, we propose an Adaptive Variable Learning Rate EBP algorithm to attack the challenging problem of reducing the convergence time in an EBP algorithm, aiming to have a high-speed convergence in comparison with standard EBP algorithm. The idea is inspired by adaptive filtering, which led us into two semi-similar methods of calculating the learning rate. Mathematical analysis of AVLR-EBP algorithm confirms its convergence property. The AVLR-EBP algorithm is utilized for data classification applications.

Simulation results on many well-known data sets shall demonstrate that this algorithm reaches a considerable reduction in convergence time in comparison to the standard EBP algorithm. The proposed algorithm, in classifying the IRIS, Wine, Breast Cancer, Semeion and SPECT Heart datasets shows a reduction of the learning epochs relative to the standard EBP algorithm.

2. MATERIALS AND METHODS FOR PROBLEM SOLUTION, ACCEPTED ASSUMPTIONS

In order to train the networks with a pseudo-random distribution of links, the authors of the article propose several modifications of error back-propagation method, which are able to carry out the process of such architecture network learning effectively:

- electric circuit method;

- the method of error backpropagation with graph bypassing by depth;

- the method of error backpropagation with graph bypassing by width.

2.1 ELECTRIC CIRCUIT METHOD

It is acceptable to take one of the following values as the criterion for neuron input significance at the current stage of learning:

- communication weight;

- input signal value;

- the input signal weighted value (the product of input activation and link weight);

- the derivative value of the daughter neuron activation function;

- the product of the derivative activation function value of the daughter neuron and the weight of the corresponding connection.

During computational experiments, it was found that in most cases neural networks had the greatest learning efficiency, in which the absolute value (by module) of the input activation was used as the criterion for the input significance. The significance value is taken equal to 1 for the pseudo connection that implements the displacement of the neuron adder function.

Step-by-step algorithm for a neural network learning by electrical circuit method.

1) Running the learning procedure for each output neuron of the network by the order of their numbering.

2) The random development of a unique learning process identifier.

3) Each of the output neurons undergo several successive learning iterations. All actions are performed with the same process ID.

A step-by-step learning algorithm for individual neurons using the electrical circuit method:

1) For the current neuron, the identifier is set equal to the learning process identifier.

2) They calculate the value of the derivative activation function for the given neuron.

3) They evaluate the significance of the neuron inputs.

4) Among the unvisited nodes (the identifiers of which do not coincide with the process identifier), the most significant input is selected by the absolute value of the input activation in case of their availability.

5) For this input, the weight is adjusted by the error back-propagation method.

6) If this input is connected to a real neuron and does not implement an offset, then the learning algorithm is applied recursively to this neuron. The value of the error functional for such a neuron is calculated by the product of the current error and the weight of the corresponding connection.

2.2 ERROR BACK PROPAGATION METHOD WITH GRAPH BYPASSING BY DEPTH

The following algorithm is performed in turn for each output neuron of the network:

1) For the current neuron, the identifier is set equal to the learning process identifier.

2) They calculate the value of the derivative activation function of the neuron.

3) For each input, weights are adjusted according to the error back-propagation method. If this input is connected to a real and unvisited neuron, then the learning algorithm is applied recursively to this neuron. The value of the error functional for such a neuron is calculated by the product of the current error and the weight of the corresponding connection.

2.3 ERROR BACK PROPAGATION METHOD WITH GRAPH BYPASSING BY WIDTH

The following algorithm is performed in turn for each output neuron of the network:

1) For the current neuron, the identifier is set equal to the identifier of the learning process.

2) The value of the neuron activation function derivative is calculated.

3) For each input, weights are adjusted according to the error back-propagation method.

4) The recursive application of the learning algorithm is performed for each input that is connected to a real and unvisited neuron. The value of the error functional for such a neuron is calculated by the product of the current error and the weight of the corresponding connection.

2.4 ALGORITHM TESTING

The abovementioned neural network learning algorithms were tested by the simulation of the mercury electrolysis cell operation mode.

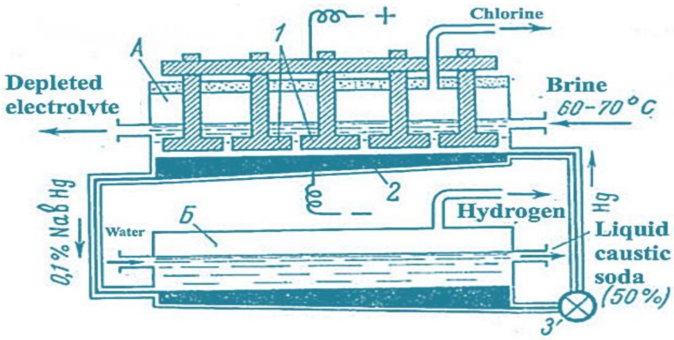

Figure 1: Mercury Electrolyzer Device

This device consists of the horizontal bath A with a salt solution. At the same time there is the layer of mercury (liquid sodium amalgam) at the bottom of the bath, which is the cathode 2. The anode 1, consisting of metal oxide plates, is dipped into salt solution using special frames.

According to the reference data [9], the theoretical value of the voltage required for the implementation of the electrolysis process makes 3.1 V. However, the practical value of the required voltage is somewhat higher. This is due to the resistance of the salt solution, which is greatly enhanced after its filling with chlorine bubbles.

Voltage is one of the most important indicators responsible for electrolysis process quality and safety. Therefore, an urgent task is to simulate this process in order to calculate the voltage for various device operation modes. The task is reduced to an approximation of the electrolysis cell voltage dependence on the abovementioned factors [10].

3. RESULTS

In order to train neural networks based on experimental data, the training sample was prepared in CSV format. The sample volume makes 525 records [11].

Each entry contains the following values:

1) 1) Current power, kA. The values of integer type located in the interval [90; 425]. Input parameter.
2) 2) Solution temperature, °C. Values of integer type located in the interval [60; 80]. Input parameter.
3) 3) The distance between electrodes, mm. The values of integer type located in the interval [5; 7]. Input parameter.
4) 4) Voltage, V. Real type values with the accuracy of up to 2 characters, located in the interval [3.41; 5.56]. Output parameter

Using the settings in the normal method, the input parameter values were normalized [12] to the interval [-1; 1].

Since during the electrolyzer operation the voltage is capable of changing over a wide range, the output value was normalized from the interval [0; 1] in the interval [3; 6].

Neural networks were used for training with the following set of parameters:

- stabilization of activations is absent;
- logistic activation function;
- learning rate - 0.1;
- random key values in the range [1; 10^9];
- the steepness of activation function is equal to one;
- initialization of weights with random values in the range [-0.5; 0.5] with a uniform distribution law;
- there is no regularization of scales;
- 2nd order error functional;
- loop development prohibition;
- multiple bond development prohibition;
- the number of input neurons - 3;
- all network inputs are continuous;
- the number of output neurons - 1;
- the output is continuous;
- the number of hidden neurons - 4;
- network connectivity - 3.

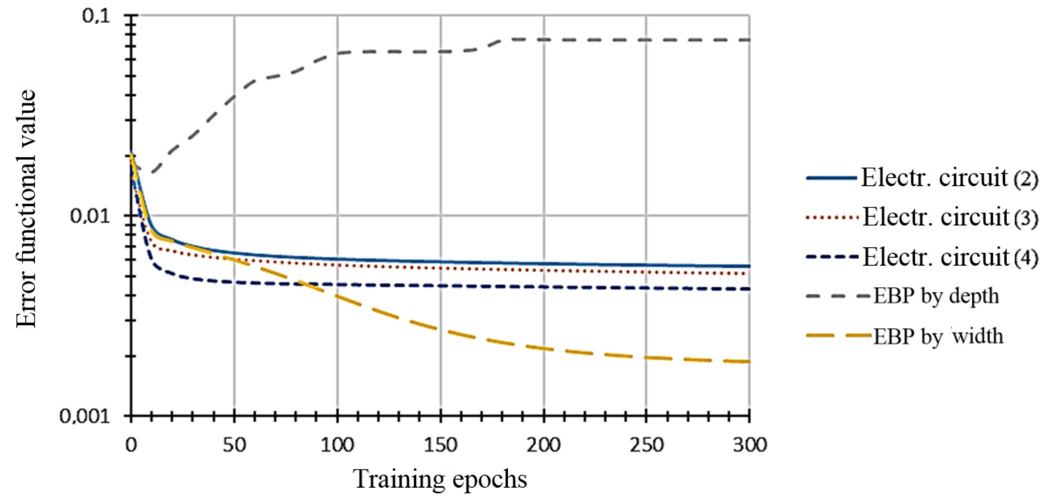

Figure 2: The graph of error functional average values in the learning process

Figure 2, the learning process was carried out using various learning algorithms in order to compare their effectiveness. They tested the algorithms of the “electrical circuit” with 2, 3, and 4 repeats of training, as well as the algorithms of error backpropagation with the bypass of the graph by depth and width.

As a training sample, they used the entire array of source data. The sample records were processed in random order at each iteration of learning. The duration of training was 300 epochs.

The example was considered as recognized if the absolute error was no more than 0.1 V.

In order to assess statistically the efficiency of the algorithms for each training option, the calculation was made with 10 repetitions at different network key values (Figures 2 and 3).

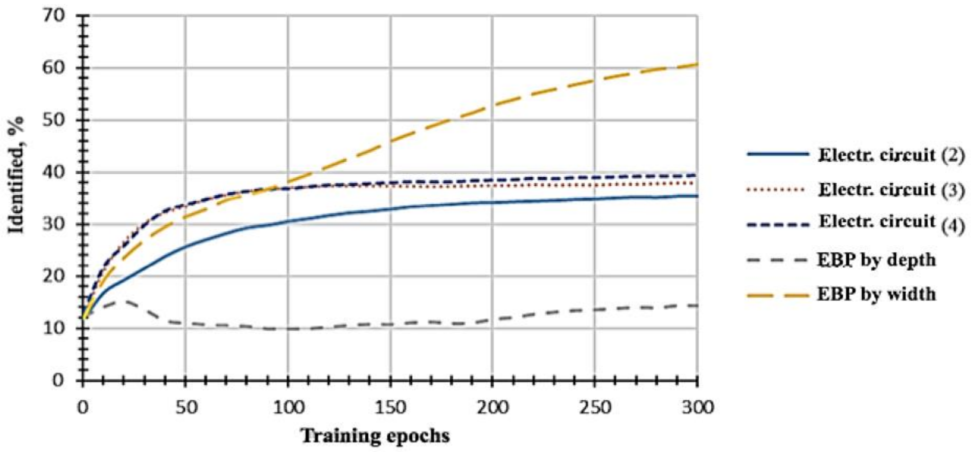

Figure 3: The graph of average learning efficiency by various methods

Figure 4 shows that during the approximation problem solution, the error backward propagation (EBP) algorithm with the graph bypass by width provides the highest efficiency. On the contrary, the algorithm of error backpropagation with the graph bypass by depth leads to the error value increase in most cases. They also performed a comparison of the most efficient neural networks for each of 10 repetitions (Figures 4 and 5).

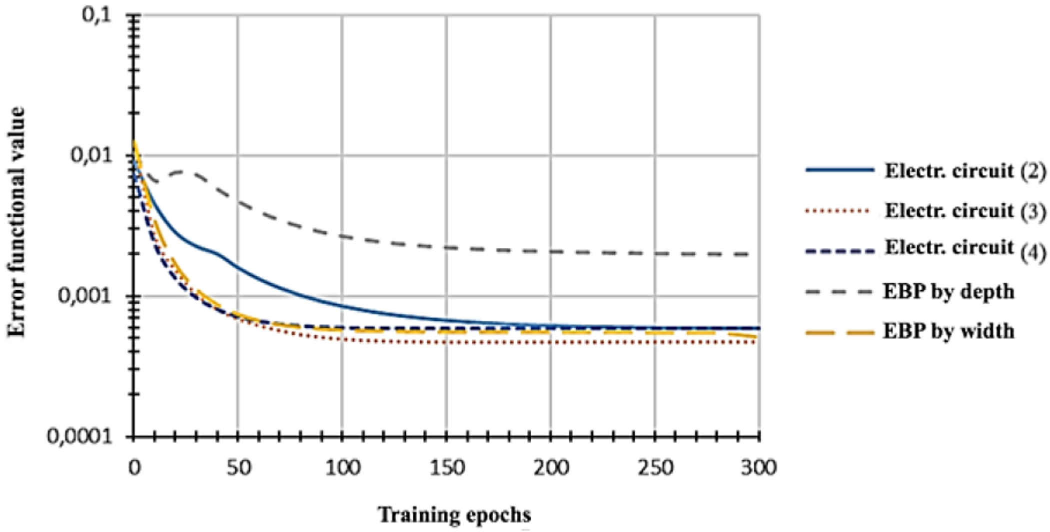

Figure 4: The graph of the smallest error functionals in the learning process

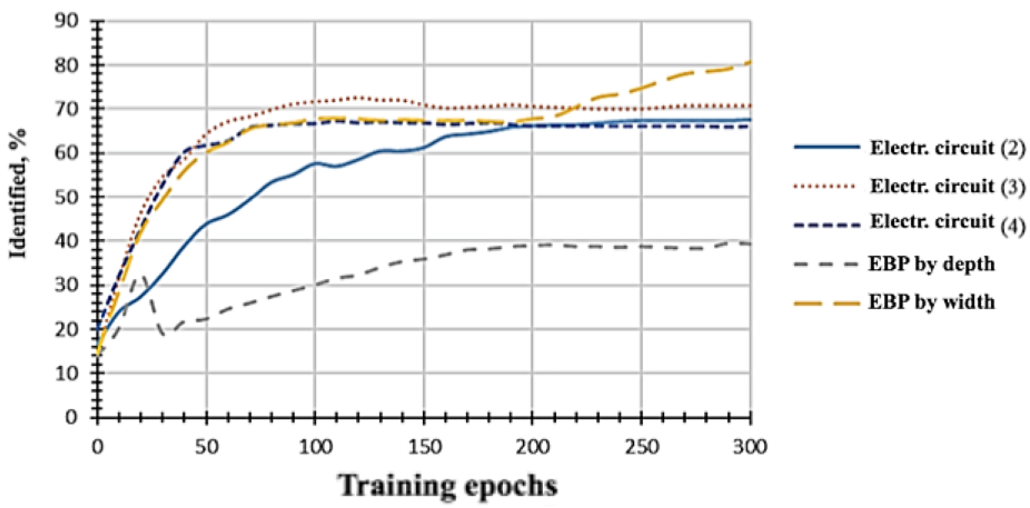

Figure 5: The graph of best learning result efficiency by various methods

According to network training schedules that showed the best result, it follows that the method of error back-propagation by depth can also lead to error decrease and to recognized example number increase. However, the low rate of error reduction and the high probability of the opposite effect suggest that this method is ineffective.

It should also be noted that the considered task of the electrolyzer operation mode modeling does not require the analysis of the previous values, that is, it is static [13]. However, due to the presence of cycles in the structure of applied neural networks, the accumulations of activations occur from previous sets of input data. In fact, the input data are processed as the values of a certain time series [14–16].

In order to eliminate the influence of the previous examples, they considered the neural networks which turned on activation stabilization. A neuron was considered as stabilized if during the reprocessing of the same data set, the value of its activation changed by 0.0033 maximum. This value was chosen due to the fact that this error will be 0.01 V during the output signal normalization, that is, it will correspond to the experimental data accuracy.

Other parameters were left with the same values as in the previous case. The learning outcomes are presented in Figures 6-9.

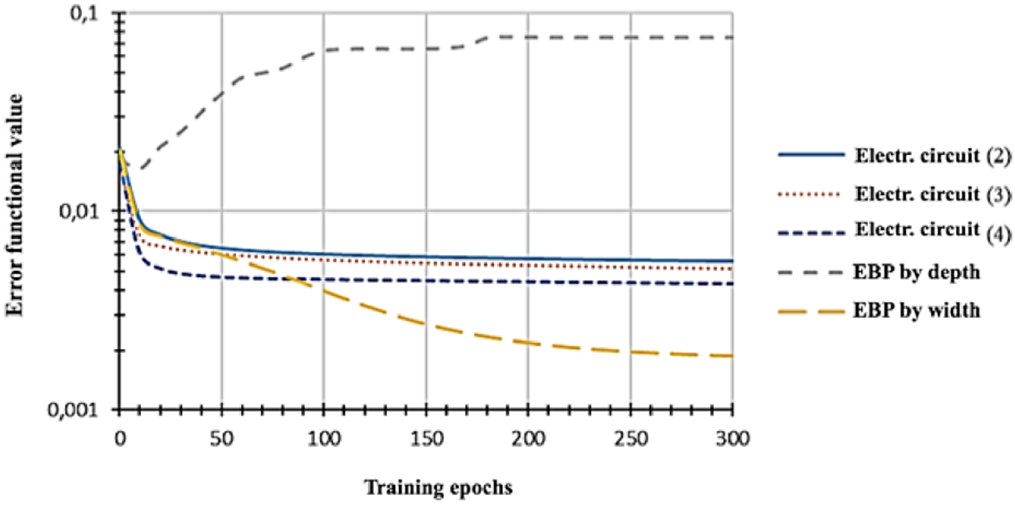

Figure 6: The graph of the error functional average values in the process of neural network learning with stabilization

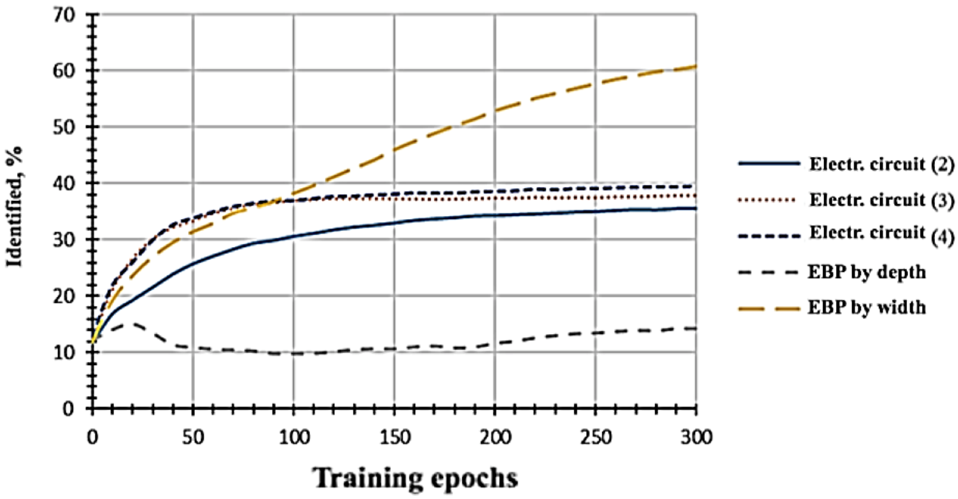

Figure 7: The graph of average learning efficiency of neural networks with stabilization by various methods

4. DISCUSSION

Estimates of the average learning efficiency for various methods at the end of 300 epochs are presented in Table 1 depending on stabilization presence or absence.

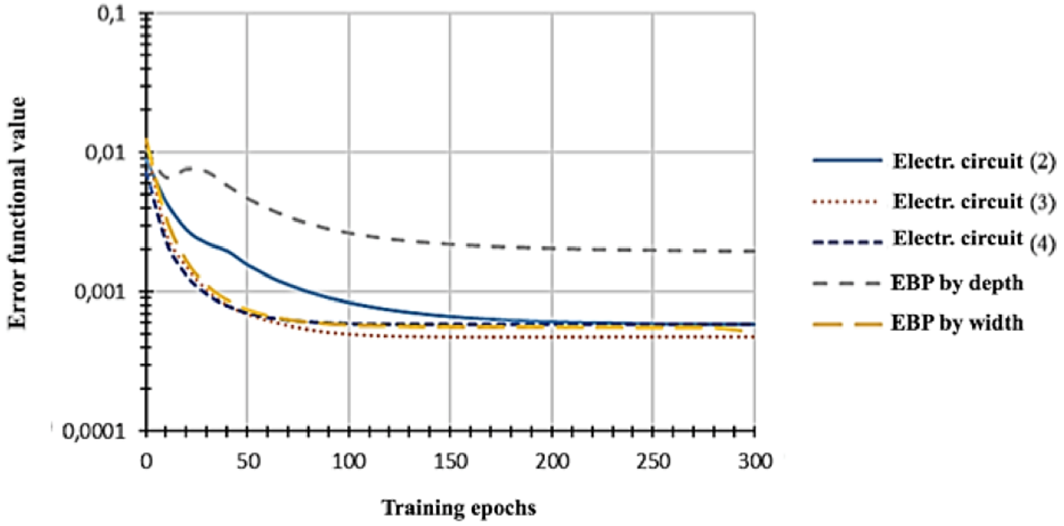

Figure 8: An error functional concerning the best learning results of neural networks with stabilization by various methods

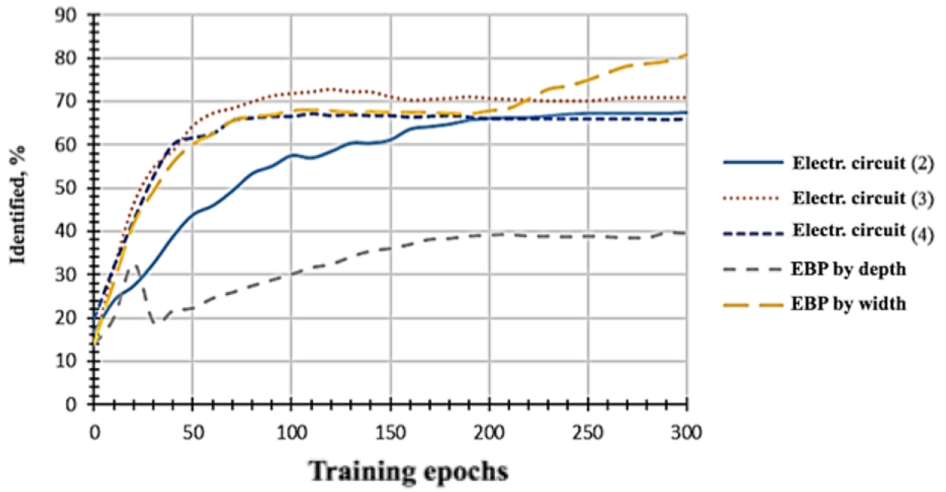

Figure 9: The best learning result effectiveness concerning the neural networks with stabilization by various methods.

Table 1: Average efficiency of learning.

Method	Without stabilization, %	With stabilization, %
Electric circuit method, 2 repeats of training	27,8287	35,4476
Electric circuit method, 3 repeats of training	35,2572	37,9047
Electric circuit method, 4 repeats of training	42,8380	39,4666
Error backpropagation method with graph bypassing by depth	15,5810	14,3239
Error backpropagation method with graph bypassing by width	47,3524	60,7047

Thus, the use of neuron activation stabilization leads to the learning process efficiency increase:

- for the method of error backpropagation with the graph bypassing by width;
- for the method of the electric circuit with small values of learning repetitions (as compared to the network connectivity).

A slight decrease of example recognition efficiency for other methods is associated with result statistical error.

Also, they performed the evaluation of the execution period for various learning algorithms. The computer with the following characteristics was used for testing: Intel Core i5-7200U processor with 2 cores, the processor frequency makes 2.7 GHz with frequency increase to 3.1 GHz by Turbo Boost technology, DDR4 memory - 12 GB, operating system - 64-bit Windows 10 Home for a single language, version 1803, build 17134.81, Python 3.6.5.

The diagram of the learning process average duration is shown in Figure 10. The results of benchmarking indicate that the duration of training for different algorithms varies slightly. The use of stabilization (with the chosen accuracy) leads to the duration increase only by 0.1%.

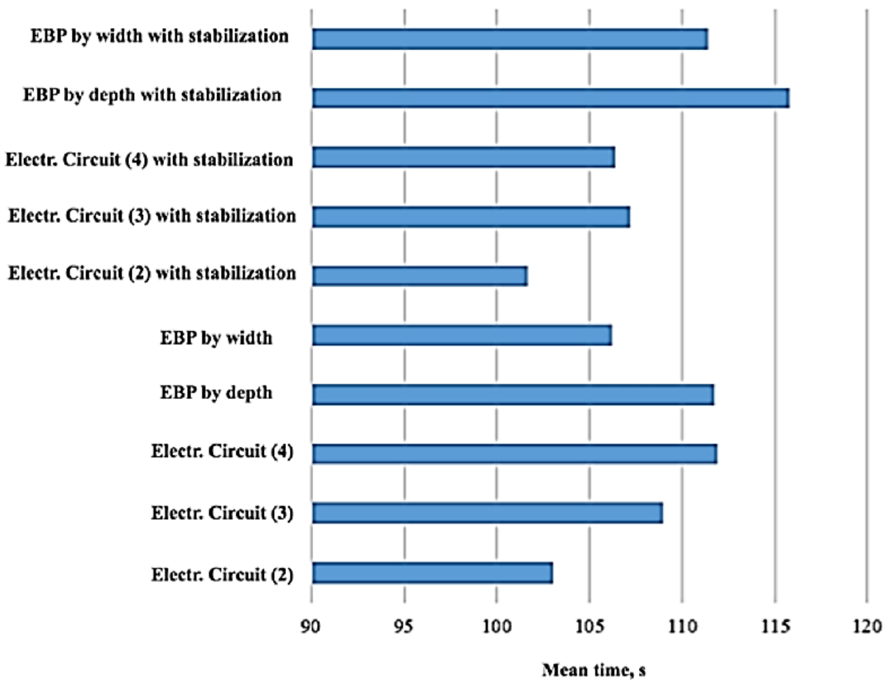

Figure 10: The average duration of various learning algorithms.

5. CONCLUSION

The achieved learning efficiency makes 93.333% of recognized examples. The highest efficiency was shown by the error backpropagation algorithm with the graph bypassing by width. This observation is consistent with the fact that the neural network bypassing by width describes most accurately the process of error backpropagation in multilayer neural networks: layer by layer from the output layer to the deep ones. Since the operation mode of the mercury electrolyzer was modeled for a static model - without taking into account time, the use of algorithms with output value stabilization showed the greatest efficiency.

Thus, the result shows that these algorithms are effective for neural network training with a pseudo-random distribution of connections. In order to solve the regression problems, the modified error back-propagation algorithm with the graph bypass by width and signal stabilization showed the highest efficiency.

6. AVAILABILITY OF DATA AND MATERIAL

Information used and generated from this work is available by contacting the corresponding author.

7. ACKNOWLEDGEMENT

The work is performed according to the Russian Government Program of Competitive Growth of Kazan Federal University. The study was carried out with the financial support of the Russian Foundation for Basic Research in the framework of the research project No. 17-47-020068 and the project No. 13.5143.2017/8.9, carried out by the university as part of the government assignment from RF Ministry of Education and Science.

8. REFERENCES

[1] Pyatakovich, V.A., Vasilenko, A.M., & Khotinsky, O.V. (2017). Analytical design and initial structures of artificial neural network, technical implementation of mathematical neuron model. *Internet journal "Naukovedenie"*, 9(3): (40), p. 89.

[2] Shaymukhametov, D.R., Mustafina, S.A., & Shaymukhametova, D.V. (2017). Direct simulation of methylbutene dehydrogenation process to isoprene based on artificial neural networks. *Bulletin of the Technological University*, 20(24): 123-126.

[3] Starikov, A. (2011). Neural networks - mathematical apparatus. BaseGroup Labs.

[4] Aralbaeva, A.R., Karamova, A.I., & Mustafina, S.A. (2016). Neural network modeling of industrial catalytic processes. From the collection "Mathematical modeling of processes and systems. Proceedings of the Vth All-Russian Scientific and Practical Conference dedicated to the Academician A.N. Tikhonov 110th anniversary, 76-79.

[5] Werbos, P.J. (1974). New tools for the behavioral sciences. Ph.D. thesis, Harvard University, Cambridge, MA, 453 p.

[6] Zlobin, V.K., & Ruchkin, V.N. (2011). Neural networks and neurocomputers. SPb.: BHV-Petersburg, 256 p.

[7] Haykin, S. (1999). Neural networks: a comprehensive foundation. Second edition. Prentice Hall, 864 p.

[8] [8] Gorbachevskaya, E.N. (2012). Classification of neural networks. *University Bulletin*, 2(19): 128-134.

[9] Yakimenko, L.M., & Pasmanik, M.I. (1976). Handbook for chlorine, caustic soda and basic chlorine product manufacture. M.: Chemistry, 600 p.

[10] Shulaeva, E.A., Shulaev, N.S., & Kovalenko, Yu.F. (2016). Modeling of the electrolyzer parameters during caustic soda, chlorine and hydrogen production. *Oil and gas business*, 14(4): 103-107.

[11] Lagunov, N.A., & Mezentseva, O.S. (2014). Analysis and experimental study of neural network training quality dependence on the quality of training sample parameters. *Bulletin of the North Caucasus Federal University*, 5: 15-21.

[12] Nikulin, V.N., Kanishchev I.S., & Bagaev I.V. (2016). The methods of data balancing and normalizing to improve the quality of classification. *Computer tools in education*, 3: 16-24.

[13] Sinchukov, A.V. (2016). Modern classification of mathematical models. *Innovation Science*, 3-1(15): 214-215.

[14] Tatyankin, V.M. (2014). The use of multilayer neural networks for time series prediction. Priority trends of science and education development, 3(3): 195-197.

[15] Kataseva, D.V., Katasev, A.S., Kirpichnikov, A.P., & Abyanov, B.E. (2016). Fuzzy neuro model for time series analysis and forecasting. Bulletin of the Technological University, 19(13): 127-131.

[16] Kolesenkov, A.N., & Konkin, Yu.V. (2015). Neural network modeling for time series prediction. The dynamics of complex systems - XXIst century, 9(3): 10-13.

Professor Dr. Svetlana Anatol'evna Mustafina is Professor at Bashkir State University, Russia. She holds a Doctor of Physio-Mathematical Sciences degree. Her scientific interests include Mathematical Modeling of Chemical-Technological Processes.

Olga Anatolievna Medvedeva is an Associate Professor at the Department of Fundamental Informatics and Information Technology and the Department of Programming Technologies, Institute of Computational Mathematics and Information Technology, Kazan Federal University (KFU). She is a Candidate of physical and mathematical sciences at Kazan Federal University. She has a Master's degree in Mathematics, Computer Science and Computer Engineering from Faculty of Physics and Mathematics, Sterlitamak State Pedagogical Academy. Her research encompasses Physical chemistry, mathematical models of chemical reactions".

Aleksandr Nikolaevich Ivanov is a Graduate Student at Bashkir State University, Russia. He is a Senior Researcher at Alexander Butlerov Institute of Chemistry. He is a Candidate of Science. His research is related to Chemistry and Math Modeling.

Professor Dr.Nikolay Danilovich Morozkin is Rector of Bashkir State University, Russia. He is Professor at the Department of Applied Informatics and Numerical Methods, Bashkir State University. He holds a Doctor of Physical and Mathematical Sciences degree from Kazan Federal University, Russia. His research is in the field of Math Modeling.