Для вирішення задач апроксимації широко використовуються математичні моделі у вигляді штучних нейронних мереж (ШНМ). Використання цієї технології передбачає двох етапний підхід. На першому етапі визначається структура моделі ШНМ, а на другому етапі здійснюється навчання для отримання максимального наближення до еталонної моделі. Максимальне значення наближення до еталону визначається складністю архітектури ШНМ. Тобто, підвищення складності моделі ШНМ дозволяє підвищувати точність апроксимації, а, відповідно, і результату навчання. При цьому визначення структури моделі ШНМ, що здійснює апроксимацію із заданою точністю, визначається як процес оптимізації.

Однак підвищення складності ШНМ призводить не тільки до підвищення точності, а і до підвищення часу обчислювального процесу.

Таким чином, показник «задана точність» не може використовуватися в задачах визначення оптимальної архітектури нейронної мережі. Це пов'язано з тим, що результат вибору структури моделі і процесу її навчання, котрий базується на забезпеченні необхідної точності апроксимації, може зайняти неприйнятний для користувача часовий проміжок.

Для вирішення завдання структурної ідентифікації нейронної мережі використовується підхід, у рамках якого здійснюється визначення конфігурації моделі за критерієм ефективності. У процесі реалізації розробленого методу узгоджується часовий чинник вирішення завдання і точністю апроксимації.

Запропонований підхід дозволяє обгрунтувати принцип вибору структури і параметрів нейронної мережі, спираючись на максимальне значення показника ефективності використання ресурсів

Ключові слова: штучна нейронна мережа, оптимізація структури, апроксимація функцій, критерій ефективності

UDC 007.5
DOI: 10.15587/1729-4061.2019.164591

DEVELOPMENT OF A METHOD FOR STRUCTURAL OPTIMIZATION OF A NEURAL NETWORK BASED ON THE CRITERION OF RESOURCE UTILIZATION EFFICIENCY

I. Lutsenko
Doctor of Technical Sciences, Professor
Department of Information and Control Systems**
E-mail: morev.igor11@gmail.com
O. Mykhailenko
PhD, Associate Professor
Department of Power Supply and Energy Management****
O. Dmytriieva
PhD, Associate Professor
Department of Management and Administration
Kharkiv National Automobile and Highway University
Yaroslava Mudroho str., 25, Kharkiv, Ukraine, 61002
O. Rudkovskyi
Doctor of Economic Sciences, Professor
Department of Geoengineering
Warsaw University of Life Sciences
Nowoursynowska str., 166, Warszawa, Poland, 02-787
V. Mospan
PhD, Associate Professor*
D. Kukharenko
PhD, Associate Professor*
H. Kolomits
Assistant***
A. Kuzmenko
Senior Lecturer***
*Department of Electronic Devices**
**Kremenchuk Mykhailo Ostrohradskyi National University
Pershotravneva str., 20, Kremenchuk, Ukraine, 39600
Department of Electromechanics*
****Kryvyi Rih National University
Vitaliya Matusevycha str., 11, Kryvyi Rih, Ukraine, 50027

1. Introduction

The development and application of mathematical models, whose operation is similar to the principles of functioning of the human nervous system, namely the biological neural network, is one of the areas of scientific research that have been advancing rapidly over recent time. This is evidenced by the large number of publications not only of scientific, but also scientific-popular, character. These models are used to solve a wide range of scientific and practical tasks, among which one can highlight the classification, clustering, and approximation of functions. As regards the latter, it is considered [1] that applying the model of an artificial neural network (ANN) makes it possible to approximate functions of arbitrary complexity. The specified problem typically comes down to deriving a mathematical notation of objects

and processes, information about the structure or parameters of which is incomplete or completely absent. This can employ other structures as well, such as the autoregression models AR, MA, ARMA, ARX, OE [2], models based on filters with the finite (FIR) and infinite pulse characteristic (IIR) [3], or models based on the systems of orthonormal functions (OBF) [4, 5], as well as and their non-linear interpretations. However, the higher performance speed of computational processes within ANN, which is especially important in hardware-software implementation [6], when compared with others, predetermines a widespread use of the neural-network structures.

When constructing models based on ANN, problematic issues are their structural and parametrical identification.

In the general case, ANN contains three types of layers: input, output, and hidden. In this case, in contrast to the first two, there may be several hidden layers. Each layer contains a certain number of nodes (neurons). If the number of neurons in the input and output layers remains unchanged and is defined by the set task, then the number of nodes in hidden layers can be arbitrary. Thus, during structural identification, one can separate two variables – the number of hidden layers and the number of neurons in them. In this case, an increase in the number of layers and neurons leads, on the one hand, to an increase in the accuracy of a neural network, and on the other hand, to the compromised performance of parametric identification.

At the same time, one should also take into consideration that computing hardware resources are limited, and the required accuracy of approximation of the model's output to test data may be unattainable. Therefore, when choosing an optimal architecture, there must be a compromise established between the quality of the received model and the duration of its training.

Thus, the need to construct a method for determining the optimal configuration of ANN to solve a task on the approximation of functions of arbitrary complexity predetermines the relevance of our research.

2. Literature review and problem statement

There has been a series of studies into development of methods for determining the optimal structure of ANN. Paper [7] categorized methods of structural identification into three groups. The first includes those methods that simplify a neural network by eliminating its separate elements (nodes and links) that don not essentially affect the output of the model. The second includes those methods that gradually increase the number of hidden layers and the number of nodes within them in order to achieve the required accuracy of the model. The third includes those methods that employ evolutionary algorithms.

Paper [8] reports a method to simplify a neural network by excluding links whose weights' values are below a certain fixed boundary value (magnitude-based pruning – MP). The process of optimizing the structure of the model consists of three stages. The initial structure of a network is formed first, which consists of a large number of hidden layers with a significant number of nodes within each of them, followed by its training. Next, one determines and eliminates the links that have lower values for the weights. At the final stage, an estimation of the parameters for a simplified neural structure is performed. The process is repeated iteratively until the achievement of the permissible level of the model's accuracy. Despite the simplicity of implementation, the authors do not define at what level it is necessary to set a threshold that indicates a need to exclude a link. They also do not disclose the accuracy to which the determining of a network's architecture is advisable. The proposed method implies the reproduction of a complete learning cycle after each change in the structure that leads to high computational load.

The main feature of the approach, based on excluding communication lines that have lower values of the weights, is a significant growth of the model's error at each iteration.

Paper [9] considers the Optimal Brain Damage (OBD) method, which implies determining those communication links and their number whose elimination will not lead to a significant increase in the overall error in a neural network. Hessian matrix is calculated for this purpose, the elements of which are the second derivatives form the network's error based on the parameters $w_{ij}^{(k)}$. Given that the computational load in the calculation of such a matrix is very significant, the authors proposed to simplify it to a diagonal form that led to the declining quality of this method. The parameters and, therefore, communication lines that correspond to the elements of the matrix with low values for the second derivative are excluded from the network. The process of optimizing the architecture proceeds to the moment of achieving an acceptable level of the model's error. OBD-method, similarly to MP-method, is implemented iteratively, but it makes it possible to remove the links whose absence will not essentially affect the accuracy of the model. Despite this, the need, after each exclusion of links, to carry out a complete cycle of network training and to calculate the Hessian matrix leads to a decrease in the speed of structural identification.

A key feature of the OBD procedure is to determine the Hessian after the convergence of the parametric identification process, which greatly influences the duration of determining the structure of a neural network. Study [10] proposed to determine the significance of links until the achievement of a local minimum of the error's function at a direct run of the neural network – the method of Early Brain Damage (EBD). The authors introduced the criterion of significance of EBD, which represents the second derivative from the difference between the error's function for the value of weight at the convergence of the learning algorithm and for a zero weight value. However, the work does not establish a sufficient number of iterations to train a network at which calculation of the criterion would make it possible to correctly estimate the significance of parameters of its communication lines. The difference of the architecture optimization procedure is the exclusion from the structure of half the connections with lower values for the EBD criterion. However, the appropriateness of excluding this very number of links is not explained by the authors.

A method of Optimal Brain Surgeon (OBS) [11] advances the principles of the OBD method. It also employs the Hessian matrix for assessing the significance of the weight of a link. The criterion used is the ratio of the square of a value for weight to the double value for the diagonal element of the inverse Hessian that matches it. To be excluded are the weights with the lowest values for the criterion. The advantage of the method is that it requires only a single cycle of the direct run of a neural network to convergence. Then the criterion of significance is calculated. Upon excluding a direct link, its weight and the diagonal element of the inverse Hessian matrix that corresponds to it is used to compute the updated values for the remaining weights. As a result, evaluating

the significance does not lead to the simplified Hesse matrix, which improves the quality of assessment. A special feature of the method is a great computational load predetermined by the necessity to calculate the inverse Hessian.

Another technique to optimize the architecture is to exclude not the links but neurons (nodes). Such an approach makes it possible to considerably simplify a neural network, because eliminating one node in a hidden layer leads to the removal of all input and output links related to it.

Paper [12] suggested the method NoiseOut, which makes it possible to combine neurons with a high level of activations correlation. To determine such a pair of neurons, the output values for a test sample are imposed with an additive disturbance. Identification of the structure is performed in the process of training the model. However, the authors do not define the degree of correlation between two neurons when one of them may be excluded. Only an ideal case is provided, when the correlation equals unity, which in real processes, especially when adding the noises of random character, is impossible. They also did not establish which of the two nodes is subject to exclusion and the way the exclusion of certain links would affect the process of determining values for the weights of the remaining links, given that the form of errors' functions in the output and hidden layers can change dramatically. Determining the structure of ANN ends when the accuracy of the network is below an established threshold, but the authors did not specify what it should be.

Paper [13] developed a method for excluding the nodes based on assessing the significance of a neuron for three criteria. The first one assesses a node based on the function of the entropy of its significance, which depends on the number of elements in a test sample that led to the activation or deactivation of the neuron. Activating the node is understood by the authors as establishing at its output the value greater than zero while applying a sigmoid activation function. Two other criteria are average values for the weights of the input and output links of the node. Evaluation of neurons is performed after completing the learning process followed by the exclusion of nodes with lower magnitudes for a significance criterion. Along with the node, all its input and output links are removed, which leads, as already noted, to a significant deterioration in the accuracy of a neural model. To improve quality, the authors propose to repeat the procedure of parameters assessment. Special features of the method are the need to perform learning cycles before and after removing a neuron, to evaluate significance of a node separately, based on one of the three suggested criteria, rather than comprehensively, the uncertainty about a threshold level at which the neuron is considered to be activated. The authors did not provide a specific condition for removing a neuron.

Work [14] advances the ideas proposed in study [13]. The authors introduce an integrated function to evaluate the significance of a neuron, which combines the function of the node's importance entropy and the functions of significance entropy by the input and output links of this neuron input, which are introduced instead of average values for the respective weights of input–output. Based on the sigmoid function, which employs the developed integrated criterion as an argument, the authors defined the regions of intersection of individual entropies at which it is advisable to remove the hidden layer node. This method, similarly to the previous case, implies, before starting the procedure of structure determination and after the removal of a single node, training to the convergence of the algorithm. In addition, the authors do not substantiate the boundary level of neuron activation when determining the function of its importance entropy. The proposed integrated criterion of significance does not take into consideration the duration of network learning.

Paper [15] proposed a method for constructing a neural network that implies the determining of a hidden layer to which a node can be added. In this case, a neuron can be introduced to the existing or newly created layer regardless of the number of nodes that they already contain. The number of neurons in separate layers can be different. The devised criterion is integrated and contains two expressions: the difference in errors of the neural network for the preceding and current learning epochs and the absolute value for average difference between the initial values of the two previously added neurons for each count of reference sample. Each condition is assigned with thresholds. Depending on which of these values has been reached, they determine the place for adding a new neuron to the structure of a neural network.

The application of the considered method implies training a model in two stages. First, the weights of communication lines for the added neuron are initiated at zero values, for those existing – random magnitudes. Then they approximate the values for the parameters of this node to the optimal ones based on the error back propagation algorithm. The process of determining the parameters stops at the threshold of the first expression in the criterion considered above. It is believed that in this case a local minimum of the error's function is achieved. At the final stage, values for the weights of links between the existing nodes are superimposed with Gaussian additive noise at zero mean and a single variation coefficient. After this operation, they assess the values for weights in line with the error back propagation method. Special features of the method include the need for retraining the model after adding each new node, insufficient substantiation of expressions for the criteria and the conditions for determining a place of adding a node to the structure of the network, as well as boundary levels for these expressions.

In [16], authors constructed a genetic algorithm to determine the optimal architecture of a neural network with a single hidden layer. The structure of the model in this case is represented as a binary string whose bits are divided into three groups. The first one includes bits that define the limits of change in the values for weights during initialization and in the learning process. The second one includes bits that determine the number of inputs to a network that are used in training. The third one includes those bits that define the number of nodes in the hidden layer. A set of random binary strings is used to form the initial population. Next, by applying a conjugate gradient method, each neural network out of the population is trained until reaching a minimum of the root-mean-square error. Based on this function, a fitness objective function is developed, which is used for the selection of rows subject to reproduction (selection) and subsequent crossbreeding. The process of crossbreeding is realized by obtaining a pair of descendants by exchanging the parts of binary strings within a pair of parents. The mutation operation is used to determine randomly the number of bits that make up the parts that are exchanged within a pair of parents. The selection of two individuals that were initially selected for crossbreeding is also performed randomly. Next, they train the individuals from the new population, then, by employing the fitness function, the selection is performed, followed by the repeated procedures of crossbreeding and mutation. Determining the structure is completed when all individuals

in a population converge to a single architecture. A special feature of this approach is the large computational load predetermined by the necessity to carry out the learning process for each copy of a neural network in populations when performing selection at each iteration of the algorithm. The proposed variant of binary encoding of the structure makes it possible to determine the number of nodes only in one hidden layer, while the constructed fitness function ignores the time required for the estimation of model parameters. In addition, the algorithm is aimed at determining the required number of nodes without the optimization of connections between the neurons of separate layers by removing communication lines with the low level of significance. It is argued that the application of the developed method would make it possible to obtain a globally optimal network architecture.

Paper [17] considers a genetic algorithm, which makes it possible to optimize the structure of the internal links in a neural network. In this case, a binary string is composed of codes of the individual communication lines. The weight of each line is described by a four-bit figure. At zero value, it is considered that there is no connection between neurons. When determining the architecture, links can be both eliminated from the structure of the model and created or restored. Direct implementation of stages in the algorithm is given in the article in the closed form. The disadvantages of the method include the complexity of the binary string. Thus, for encoding a neural network, which includes three layers of two neurons at the input and hidden layers and one at the output, as well as six communication lines, a twenty-four-bit string is used. In addition, at such an encoding the values for weights are represented only by positive integers, which complicates the search for a minimum of the error function during learning and, in some cases, does not provide for the convergence in the algorithm of model parameters estimation. The procedure of structural identification does not imply determining the optimal number of nodes. Similar to the method proposed in [16], the process of selection at all iterations of the algorithm necessitates training each copy of a neural network in the initial and new populations.

In paper [18], authors combined all three approaches that were considered above in order to determine the structure of a neural network with several hidden layers. In particular, the method of encoding the architecture of the model for subsequent optimization using an evolutionary algorithm, which, in contrast to the case discussed in [17], implies the representation of weights of communication lines not by the binary but real numbers. In this case, a copy of the neural network is also represented by a string. A connection with a zero value of the weight is considered to be excluded from the model. The process of structural identification is carried out as follows. At the beginning, an initial population is formed with the specified number of elements. The basic architecture consists of a single neuron in the hidden layer and a single line of communication between this node and a single neuron in the input layer, which is chosen randomly. Next, by applying an error back propagation method, they train those neural networks that match the elements within the original population, over a fixed number of epochs. By using a fitness function, they select a pair of elements for crossbreeding. The fitness function used is the mean square error of the network. The crossbreeding process implies merging the structures of the two networks into one overall. For example, if the initial networks include: the first one – a single hidden neuron and three links, second – two neurons and five links, then the network-descendant would consist of three neurons and eight communication lines. After crossbreeding, they perform the mutation of the population of descendants by adding one link to each model, chosen randomly. Next, the training of the received networks is performed. The next stage implies assessing the level of significance of neurons in the hidden layer in the structures of models obtained as a result of mutation. To this end, they apply the criterion, which is calculated as the square root of the module for the value of weight of the communication line between individual nodes in the hidden and output layer. A node in the hidden layer with the lowest value is removed from the network, others are divided into two groups with higher and lower values, respectively. For each neuron in the latter group, a random number is generated with a uniform probability distribution. If it is less than 0.5, then such a node is also deleted.

At the last stage, one selects from the derived structures, by using a coefficient of survival, the most suitable, in terms of subsequent crossbreeding, copies of models and the next iteration of the algorithm begins. The process of architecture optimization ends with a selection of the best model, based on the mean square error criterion, after running a fixed number of iterations (generations) of the evolutionary algorithm. In this case, the process of simplifying the structure of a neural network is not substantiated. In particular, the authors did not define the condition for removing a neuron based on the criterion of significance, they did not set any threshold value to divide nodes into groups and did not indicate why it is necessary to carry out such a differentiation. Also unclear is the probabilistic approach to removing neurons from a group of lesser significance, because, as the result of this operation, one could exclude those nodes from the network that have higher values for the criterion than those that remained.

One should specifically note the limit of the number of epochs when training a network and the number of iterations of the evolutionary algorithm. An increase in the size of a network increases its computational complexity. Therefore, there may be a case when the process of estimating the parameters of a model with an insignificant number of nodes would demonstrate a fast convergence and a better accuracy over a fixed number of epochs than the model with a larger number of nodes. The result may imply that not the best model from original population is selected for further crossbreeding. Limiting the number of iterations of the evolutionary algorithm can ensure obtaining a locally, rather than globally, optimal structure of the network.

Thus, our analysis of previous studies has revealed the absence of a method for optimizing the architecture of ANN using a verified criterion, which would relate the accuracy of the structure obtained to the time required to estimate the model's parameters.

3. The aim and objectives of the study

The aim of this work is to construct a method for determining the optimum ANN structure using a cost approach based on a comparison of the complexity of a neural network structure, the time required to train it, and the accuracy of the resulting model.

To accomplish the aim, the following tasks have been set:

– to determine prognostic estimates for the accuracy of ANN, the time required to train it, and the complexity of the model's architecture;

– to define an approach to forming the magnitudes of expert estimates for the input and output informational products of ANN;

– to construct a method for determining the optimum ANN structure using a verified indicator for the efficient utilization of resources.

4. Construction and examination of the method for structural optimization of artificial neural networks

4. 1. Essence of the method

Our review has demonstrated that there are different methods at present aimed at determining the structure of ANN as well as training methods.

The stage of synthesis of the ANN structure, especially its inner layers, is weakly connected with the peculiarities in the functioning of the examined object and is carried out without proper theoretical substantiation, often by trial and error. Thus, every change in the structure of ANN requires additional substantiation of the positive impact of the proposed changes to the structure of ANN. Such a substantiation is carried out by conducting a numerical experiment related to ANN learning and subsequent evaluation of the accepted indicators of training quality. Among those that are currently considered is the time required for training in order to obtain a certain accuracy of model's work using a test sample.

Thus, at present, structural optimization of ANN is the process of search optimization, that is the subject of special studies, and it can take an extremely long time.

The iterative process of such structural optimization of ANN is shown in Fig. 1.

The choice of ANN structure predetermines the number of weights and coefficients of nonlinearity, whose values must be defined. The learning module is responsible for it.

At the output of the learning module there form the values for variables in the structure, which are set for ANN.

Next, the outputs of ANN and the examined functional converter (FC) receive variables and the outputs of ANN and FC display the result of ANN operation and a reference.

Comparison unit is used to make a decision on achieving the required accuracy. If the predefined accuracy is not achieved over the cycle of learning, the structure of ANN is changed and the process is repeated. In this case, the approach to the selection of an appropriate structure of ANN is iterative.

An analysis reveals that such a process could not lead to the choice of the optimal structure of ANN. It cannot lead in the sense that the optimum structure is the best structure, based on the definition for the criterion of optimization. However, the criterion «accuracy» is one of the indicators for the process but not the criterion for the best solution. Accuracy can be further improved.

On the other hand, there may occur a situation in the process of moving towards the assigned accuracy that achieving it requires an inacceptable duration of the computing process. This means that the requirements to accuracy must be reduced. It also means that the indicator «accuracy» is not the only parameter that counts in the process of making a decision.

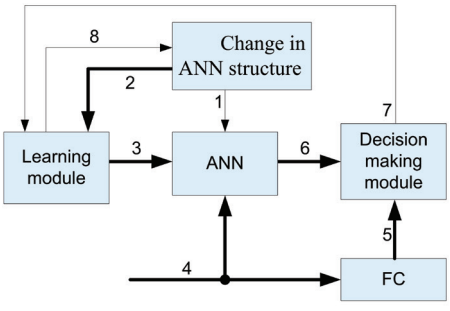

Fig. 1. Principal diagram of existing method for determining the structure of ANN: 1 — supply channel of new ANN structure; 2 — weights and coefficients of nonlinearity; 3 — values for weights and coefficients of nonlinearity; 4 — submission of test variables vector; 5 — vector of reference values; 6 — ANN response vector; 7 — channel of signal transmission «solution not found»; 8 — channel of signal transmission about the need to change the structure of ANN; 9 — channel of signal transmission «solution found»; FC — functional converter

Thus, the iterative process of structural optimization is not formalized, because it relies on a subjective approach to decision making. This is due to the fact that the researchers do not use all the necessary information for making a decision in the explicit form.

The essence of the proposed method is based on the explicit use, among others, of the indicator «operation duration» for decision making (Fig. 2).

In addition, there is no need to use an iterative approach when determining the structure of ANN close to optimal. This relates to that the increasing complexity of a network leads to a projected increase in accuracy and prolongs the time of calculation (Fig. 3).

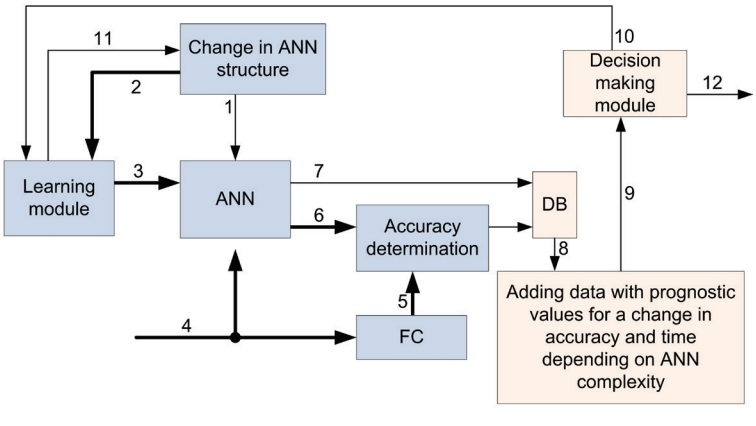

Fig. 2. Principal diagram of the proposed method for determining the structure of ANN: 1 — supply channel of the ANN new structure; 2 — weights and coefficients of nonlinearity; 3 — values for weights and coefficients of nonlinearity; 4 — submission of test variables vector; 5 — vector of reference values; 6 — ANN response vector; 7 — duration of operational process; 8 — transmission of packet with experimental data; 9 — data from a study taking into consideration prognostic values; 10 — channel of signal transmission «solution not found»; 11 — channel of signal transmission about the need to change the structure of ANN; 12 — channel of signal transmission «solution found»; FC — functional converter; DB — database

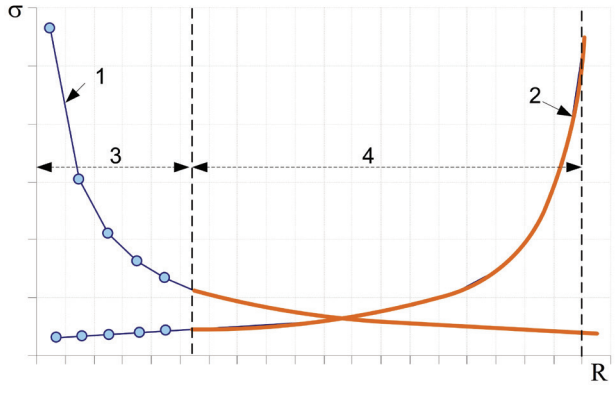

Fig. 3. Experimental data and data on extrapolation of the indicators «accuracy» and «calculation duration» depending on the complexity of ANN: 1 — accuracy of calculation; 2 — calculation duration; 3 — experimental data; 4 — prognostic value for a change in accuracy and calculation duration

The proposed method (Fig. 2) is based on the acquisition of experimental data in the process of a consistent increase in the complexity of ANN, the use of the method of technical forecasting and the optimization criterion, which makes it possible to comprehensively assess the ratio of complexity, accuracy, and calculation duration.

4. 2. Implementation of the method for ANN identification

The process of constructing and using an artificial neural network (ANN) requires the use of computing resources by hardware. In this case, there is a functional relationship between the time of parametric identification and calculation of the output value for the model, which can be defined as the process of training a neural network structure, and its quality.

Improving the accuracy of ANN model requires an increase in the number of hidden layers and the number of nodes within them, as well as an increase in the time required to estimate the parameters. Thus, increasing the value of the obtained result is accompanied by a growth in the complexity of ANN structure and an increase in computational load.

There emerges a task on comparing the expert evaluation of resources required to train ANN (RE), training duration (TO), and expert estimate of the obtained result (PE).

In this case, determining the best architecture and parameters of ANN reduces to the optimization problem based on the criterion of maximally efficient utilization of resources $E=f(\text{RE, PR, TO})$.

In turn, the scientific task is to determine the values for components of the criterion (RE, TO, PE) to ensure a possibility for the comparative assessment of different variants in the architecture of ANN.

To this end, within the framework of our research, we performed a qualitative evaluation of ANN, in the form of a model of a multilayer perceptron with a single hidden layer. The reference function that was used for approximation was a nonlinear function of the form $y=1/x$. In this case, every stage of the research was accompanied by an increase in the complexity of the neural-network structure by increasing the number of neurons in the hidden layer (Fig. 4).

One can see (Fig. 1) that the number of ANN parameters, which must be defined during learning, increases linearly in proportion to the configuration complexity.

To conduct a comparative analysis using the criterion of resource utilization efficiency, it was proposed to abandon the traditional method of parameter identification using the error back propagation algorithm and to apply the uniform search method. Such an approach makes it possible to more accurately interpret results from estimating the time required to train ANN.

To determine the optimal model parameters, one needs to set intervals for a change in the values for weights of communication lines between nodes $\left\{w_{ij}^{(k)} \in \mathrm{N} \,|\, w_{ij\min}^{(k)} \le w_{ij}^{(k)} \le w_{ij\max}^{(k)}\right\}$ and the coefficient of the form f_i of nonlinear activation functions of neurons in a hidden layer.

Prior to conducting computational experiments, we defined intervals for a change in the parameters for the compared models. They are chosen so that the uniform search process is implemented at a step, which provides sorting the values for the interval over an equal number of iterations. That is, a step must be multiple to the range of change in the value for a parameter.

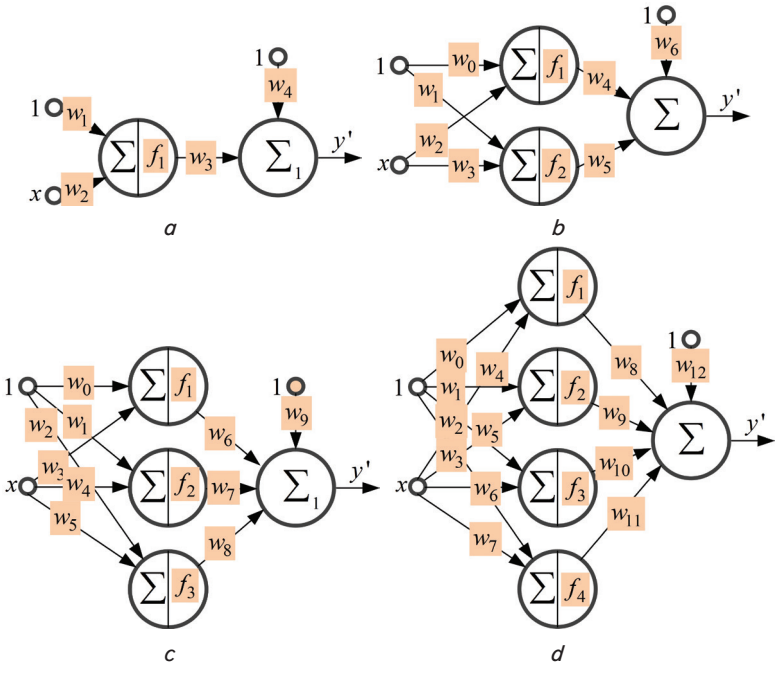

Fig. 4. Increase in the number of model parameters with an increase in the number of neurons in the hidden layer: *a* — five parameters at a single neuron; *b* — nine parameters at two neurons; *c* — thirteen parameters at three neurons; *d* — seventeen parameters at four neurons

As a result, the process of ANN training is carried out by sorting all values for model parameters at a different step, but over the same number of iterations.

After completing the parameter assessment procedure, we determined the learning duration and a value for the mean square deviation in the derived values for the model's output from test data.

Because the complexity of a neural network is growing linearly, the expert evaluation of the input product of ANN

learning operation (RE) is determined based on the number of model parameters that are subject to assessment. The RE component reflects an expert assessment of the problem set, of energy costs and hardware involved in the computational process. The PE component is defined by expert estimates of the freed hardware resources and an expert evaluation of the approximation quality of the original function (AE).

The first stage in obtaining the AE component from the efficiency criteria implies determining the function of interpolation for a change in the error of ANN model (Table 1), while obtaining the predicted values at an increase in the complexity of model's structure is performed by interpolating the function (Fig. 5).

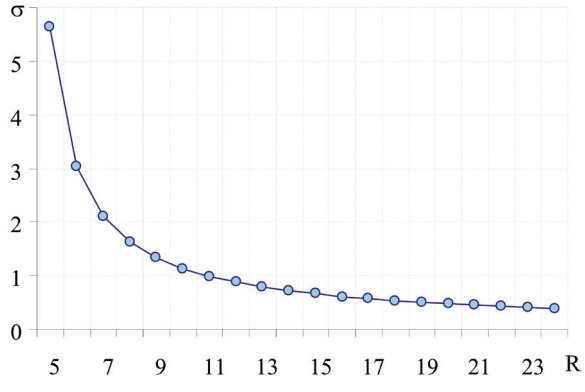

Fig. 5. A decrease in the model's error at an increase in the complexity of ANN structure

In ED olumn (Table 1), unity denotes data that were obtained experimentally.

Accept that the error of the model with a minimum configuration, that is with a single neuron in the hidden layer, has zero cost. Increase in the value of the result from approximating the function by an artificial neural network at an increase in the model's structure complexity can be determined from the following expression:

$$AE = (\sigma_1 - \sigma(R))\, f(\mathrm{A}, \mathrm{TO}) = (\sigma_1 - \sigma(R))\left(\frac{1}{1+e^{-A\cdot\ln(TO+D)}}\right),$$

where σ_1 is the error of ANN model at minimal configuration; $\sigma(R)$ is the function of change in the error due to the neural network's structure complexity; $f(\mathrm{A}, \mathrm{TO})$ is the nonlinear function of expert evaluation of model's accuracy; A is the form factor of a nonlinear function; TO is learning duration, seconds of model time; D is the displacement of the linearized time domain.

When carrying out computational experiments, it was accepted that 10 thousand seconds of model time corresponds to 1 second of real time.

Here the nonlinear function of expert evaluation of cost (Fig. 6) takes into consideration the fact that a significantly increase in model's accuracy at the initial stage of an increase in the complexity of ANN leads to a slight growth in the cost of the result. Then the cost of the result is growing rapidly, and further reduction of model's error does not lead to a proportional increase in the cost of the result.

The next stage implies determining the form of the interpolation function to obtain the predicted values for the time required to train ANN at an increase in the model's structure complexity (Fig. 7).

Correct determination of the efficiency criterion necessitates adjustment of its components. Given that the complexity of the model is growing linearly, and performance speed of the parametric identification process at an increase in the number of parameters – exponentially, then, in order to form a point of extremum, it is advisable to linearize the function of dependence of learning duration on the number of parameters.

Table 1

Estimated data obtained based on interpolation (1–13) and extrapolation (14–20) points

N	R	TO, c	Ln(TO)	Error	Accuracy	$f(R, D)$	RE	PE	AE	R	E	ED
1	5	1.13E-05	0	5.641	0.359	0.018	5	5.01	0.006	292	2.21E-05	1
2	6	1.28E-04	2.04	3.032	2.968	0.047	6	6.14	0.141	544	0.000259	0
3	7	1.45E-03	4.47	2.108	3.891	0.119	7	7.46	0.464	1123	0.000413	0
4	8	1.64E-02	6.89	1.629	4.370	0.269	8	9.17	1.176	1484	0.000792	0
5	9	1.8E-01	9.32	1.334	4.666	0.5	9	11.33	2.333	1899	0.001228	1
6	10	2.1	11.75	1.133	4.867	0.731	10	13.56	3.558	2629	0.001353	0
7	11	2.39E+01	14.17	0.987	5.013	0.881	11	15.41	4.415	3858	0.001144	0
8	12	2.71E+02	16.60	0.876	5.124	0.952	12	16.88	4.881	5719	0.000853	0
9	13	3.07E+03	19.03	0.788	5.212	0.982	13	18.12	5.118	8332	0.000614	1
1o	14	3.47E+04	21.46	0.717	5.283	0.993	14	19.25	5.247	11820	0.000444	0
11	15	3.93E+5	23.88	0.658	5.341	0.997	15	20.33	5.328	16322	0.000326	0
12	16	4.46E+06	26.31	0.609	5.391	0.999	16	21.38	5.386	21990	0.000245	0
13	17	5.05E+07	28.74	0.567	5.433	0.999	17	22.43	5.431	28993	0.000187	1
14	18	5.72E+08	31.16	0.530	5.469	0.999	18	23.47	5.469	37513	0.000146	0
15	19	6.48E+09	33.60	0.499	5.501	1.0	19	24.50	5.501	47748	0.000115	0
16	20	7.34E+10	36.02	0.471	5.529	1.0	20	25.53	5.529	59904	9.23E-05	0
17	21	8.32E+11	38.45	0.446	5.554	1.0	21	26.55	5.554	74204	7.48E-05	0
18	22	9.42E+12	40.87	0.424	5.576	1.0	22	27.58	5.576	90881	6.14E-05	0
19	23	1.07E+14	43.30	0.404	5.596	1.0	23	28.60	5.596	110179	5.08E-05	0
20	24	1.21E+15	45.73	0.385	5.614	1	24	29.61	5.614	132357	4.24E-05	0

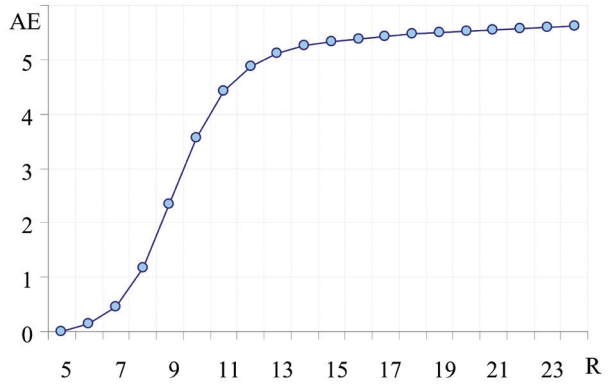

Fig. 6. Nonlinear function of change in expert evaluation of ANN accuracy

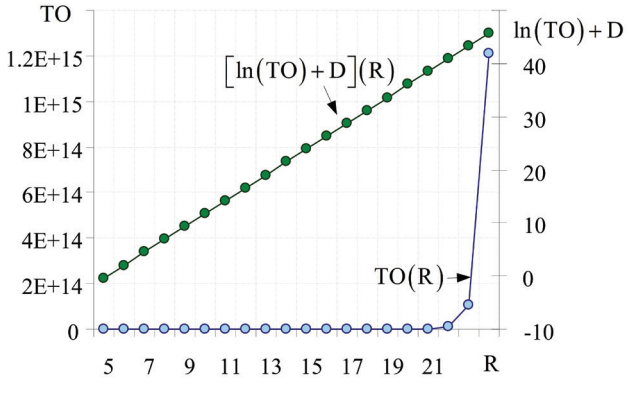

Fig. 7. Linearization of temporal dependence

The criterion of optimization used is an estimation indicator [19], which was verified for its use as the efficiency criterion [20–22]:

$$E=\frac{(PE-RE)^2}{RE\cdot PE\cdot\left[\ln(TO)+D\right]^2}.$$

Processing the results from computational experiments makes it possible to build a dependence of ANN application effectiveness for the approximation of a nonlinear function of form $y=1/x$ at an increase in the model's complexity (Fig. 8).

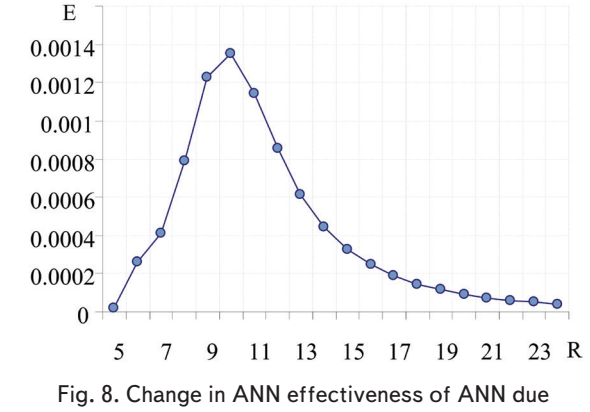

Fig. 8. Change in ANN effectiveness of ANN due to the complexity of its configuration

Fig. 8 shows that the optimum number of nodes in the hidden layer of ANN for solving the examined approximation problem is 10 neurons.

5. Discussion of research results, related to determining the structure of a neural network

Artificial neural networks were created as computing objects that simulate the functioning processes of the human brain. However, the creation of ANN is rather an attempt to reproduce the mechanism of information transformation than the fully-fledged structure, capable to independently determine its architecture, depending on the specificity of the problem being solved.

The architecture of ANN is currently determined experimentally, depending on the field of its application. And one of the most challenging tasks is to determine the intervals in a change in model parameters, specifically the weights of communication lines and the coefficients for the form of nonlinear functions of activation.

The proposed approach makes it possible to formalize the most important procedure – the choice of ANN architecture's complexity considering the accuracy of the model and the time required to train it. In this case, we apply a cost approach that naturally relates such parameters as the complexity of configuration of a multi-layered network with a single hidden layer, learning time, and accuracy of the resulting model.

But there are no limitations in the application of the method for predicting the efficiency of functioning of more complex structures, for example, at an increase in the number of neurons not only vertically (Fig. 9).

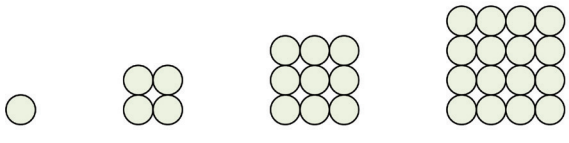

Fig. 9. Schematic representation of the method for changing the ANN structure at every next step at an increase in the number of hidden layers and at a simultaneous increase in their vertical

Thus, there is an opportunity to extend the scope of research.

In biological neural networks, the issue on structural and parametric optimization is obviously resolved using technologies that are not investigated enough as they affect such aspects of human activity as abstract thinking. In this case, the speed of biological processes is also an unattainable benchmark, despite significant progress in this area.

6. Conclusions

1. We have defined a task on the choice of the optimal ANN structure, predetermined by the necessity of adjusting the accuracy of the model obtained during parametric identification and a nonlinear growth in learning duration. To solve this problem, the approach has been proposed based on deriving predictive estimates, which could relate a growth in the learning time at an increase in the complexity of ANN architecture to the accuracy of the model obtained.

2. We have devised a cost approach to determining the magnitudes for expert assessments of the input and output informational products of ANN, which ensured the possibility to align in time the complexity of a model's structure with a level of deviation in the model's output from test data. The essence of the proposed approach implies defining an expert estimate

of the complexity of a problem being solved and an expert estimate of the cost of the result that has a particular accuracy. In this case, a value of the resulting solution is nonlinearly connected with the indicator «accuracy of calculation».

3. We have constructed a method for determining the optimum ANN structure in the form of a model of a multi-layered perceptron with a single hidden layer, based on a comparison of prognostic estimates for the efficiency of resource utilization. In this case, initial data for obtaining such estimates are: an expert value for the network configuration complexity, the time required to train it, and an expert value for accuracy of the obtained model.

References

1. Gorban' A. N. Generalized approximation theorem and computational capabilities of neural networks // Siberian J. of Numer. Mathematics. 1998. Vol. 1, Issue 1. P. 11–24.
2. Nelles O. Nonlinear System Identification. From Classical Approaches to Neural Networks and Fuzzy Models. Springer, 2001. 785 p. doi: https://doi.org/10.1007/978-3-662-04323-3
3. Diniz P. S. R. Adaptive Filtering: Algorithms and Practical Implementation. Springer, 2008. doi: https://doi.org/10.1007/978-0-387-68606-6
4. Mykhailenko O. Research of adaptive algorithms of laguerre model parametrical identification at approximation of ore breaking process dynamics // Metallurgical and Mining Industry. 2015. Issue 6. P. 109–117.
5. Mykhailenko O. Ore Crushing Process Dynamics Modeling using the Laguerre Model // Eastern-European Journal of Enterprise Technologies. 2015. Vol. 4, Issue 4 (76). P. 30–35. doi: https://doi.org/10.15587/1729-4061.2015.47318
6. Haykin S. Neural Networks and Learning Machines. 3rd ed. Pearson, 2009. 938 p.
7. A structure optimization algorithm of neural networks for large-scale data sets / Yang J., Ma J., Berryman M., Perez P. // 2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE). 2014. doi: https://doi.org/10.1109/fuzz-ieee.2014.6891662
8. Learning both Weights and Connections for Efficient Neural Network / Han S., Pool J., Tran J., Dally W. // Proceedings of Advances in Neural Information Processing Systems. 2015.
9. Liu C., Zhang Z., Wang D. Pruning deep neural networks by optimal brain damage // INTERSPEECH 2014. 2014. P. 1092–1095.
10. Tresp V., Neuneier R., Zimmermann H. G. Early Brain Damage // Proceedings of the 9th International Conference on Neural Information Processing Systems NIPS96. 1996. P. 669–675.
11. Optimal Brain Surgeon on Artificial Neural Networks in Nonlinear Structural Dynamics / Christiansen N. H., Job J. H., Klyver K., Hogsbrg J. // In Proceedings of 25th Nordic Seminar on Computational Mechanics. 2012.
12. Babaeizadeh M., Smaragdis P., Campbell R. H. NoiseOut: A Simple Way to Prune Neural Networks // Proceedings of 29th Conference on Neural Information Processing Systems (NIPS 2016). Barcelona, 2016.
13. Reshaping deep neural network for fast decoding by node-pruning / He T., Fan Y., Qian Y., Tan T., Yu K. // 2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2014. doi: https://doi.org/10.1109/icassp.2014.6853595
14. Takeda R., Nakadai K., Komatani K. Node Pruning Based on Entropy of Weights and Node Activity for Small-Footprint Acoustic Model Based on Deep Neural Networks // Interspeech 2017. 2017. P. 1636–1640. doi: https://doi.org/10.21437/interspeech.2017-779
15. A New Adaptive Merging and Growing Algorithm for Designing Artificial Neural Networks / Islam M., Sattar A., Amin F., Yao X., Murase K. // IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics). 2009. Vol. 39, Issue 3. P. 705–722. doi: https://doi.org/10.1109/tsmcb.2008.2008724
16. Arifovic J., Gençay R. Using genetic algorithms to select architecture of a feedforward artificial neural network // Physica A: Statistical Mechanics and its Applications. 2001. Vol. 289, Issue 3-4. P. 574–594. doi: https://doi.org/10.1016/s0378-4371(00)00479-9
17. Finding Optimal Neural Network Architecture using Genetic Algorithms / Fiszelew A., Britos P., Ochoa A., Merlino H., Fernández E., García-Martínez R. // Advances in Computer Science and Engineering Research in Computing Science. 2007. Vol. 27. P. 15–24.
18. Yang S.-H., Chen Y.-P. An evolutionary constructive and pruning algorithm for artificial neural networks and its prediction applications // Neurocomputing. 2012. Vol. 86. P. 140–149. doi: https://doi.org/10.1016/j.neucom.2012.01.024
19. Lutsenko I. Definition of efficiency indicator and study of its main function as an optimization criterion // Eastern-European Journal of Enterprise Technologies. 2016. Vol. 6, Issue 2 (84). P. 24–32. doi: https://doi.org/10.15587/1729-4061.2016.85453
20. Development of a verification method of estimated indicators for their use as an optimization criterion / Lutsenko I., Fomovskaya E., Oksanych I., Koval S., Serdiuk O. // Eastern-European Journal of Enterprise Technologies. 2017. Vol. 2, Issue 4 (86). P. 17–23. doi: https://doi.org/10.15587/1729-4061.2017.95914
21. Development of test operations with different duration in order to improve verification quality of effectiveness formula / Lutsenko I., Fomovskaya O., Vihrova E., Serdiuk O., Fomovsky F. // Eastern-European Journal of Enterprise Technologies. 2018. Vol. 1, Issue 4 (91). P. 42–49. doi: https://doi.org/10.15587/1729-4061.2018.121810
22. Development of the method for modeling operational processes for tasks related to decision making / Lutsenko I., Oksanych I., Shevchenko I., Karabut N. // Eastern-European Journal of Enterprise Technologies. 2018. Vol. 2, Issue 4 (92). P. 26–32. doi: https://doi.org/10.15587/1729-4061.2018.126446