Improving the Prediction Quality for a Multi-Section Transport Conveyor Model Based on a Neural Network

Oleh Pihnastyi[a] and Olha Ivanovska[b]

[a] National Technical University " Kharkiv Polytechnic Institute", 2 Kyrpychova, Kharkiv, 61022, Ukraine
[b] National Aerospace University "Kharkiv Aviation Institute", 17 Chkalova, Kharkiv, 61070, Ukraine

Abstract

The multi-section transport conveyor model based on the neural network for predicting the output flow parameters is considered. The expediency of using sequential and batch modes of training of a neural network in a model of a multi-section transport conveyor has been investigated. The quality criterion of predicting the output flow parameters of the transport system is written. Comparative analysis of sequential and batch modes of neural network training is carried out. The convergence of the neural network training process for different sizes of the training batch is studied. The effect of the batch size on the convergence rate of the neural network learning process is estimated. The results of predicting the output flow parameters of a multi-section transport system for models based on a neural network that was learned using training batches of different sizes are presented. A nonlinear relationship between the batch size and the convergence rate of the neural network learning process is demonstrated. The recommendations are given on the choice of learning modes for a neural network in the model of a multi-section transport conveyor. The choice of the initialization value of the node participating in the formation of the bias value is investigated. The qualitative regularities characterizing the influence of the choice of the node initialization value on the forecasting accuracy of the output flow parameters of the transport system are studied.

Keywords [1]

Multi-section conveyor, distributed transport system, conveyor belt, belt speed control, accumulation bunker, bias, sequential mode, batch mode

1. Introduction

To design control systems for the speed of the belt [1, 2] and the flow of material entering the conveyor input from the accumulating bunker [3, 4], models of the transport conveyor are used, the foundation of which is the aggregated equation of state of the transport system flow parameters [5], the equations of system dynamics [6], finite element method [7]. The use of the finite element method makes it possible to take into account the uneven distribution of material along the transportation route [8, 9] when designing a control system for the flow parameters of a conveyor system. Taking into account the uneven distribution of material allows you to reduce the cost of transporting material [10] due to the optimal control of the speed of the belt or the flow of material entering the conveyor input from the accumulating bunker. When designing control systems for flow parameters of a multi-section transport system [11, 12], the use of a model based on the finite element method becomes difficult due to the high cost of computing resources. In this case, along with the use of the analytical PiKh-model of the transport conveyor [13], it is advisable to use linear regression equations [14, 15, 16] and models based on the neural network [17, 18, 19] to predict the state of the flow parameters of the transport system. The relevance of using models based on a neural network increases with an increase in the number of sections of the transport system. For a modern transport conveyor [14, 20, 21], containing several dozen separate sections, the use of models based on a neural network is of particular importance.

Information Technology and Implementation (IT&I-2021), December 01–03, 2021, Kyiv, Ukraine
EMAIL: pihnastyi@gmail.com (A. 1); o.ivanovska@khai.edu (A. 2)
ORCID: 0000-0002-5424-9843 (A. 1); 0000-0003-1530-259X (A. 2)

Indeed, the construction of an optimal control algorithm for the flow parameters of the transport conveyor based on the analytical PiKh-model of the transport conveyor [13] already for several sections leads to cumbersome analytical calculations with a fairly simple control quality criterion [22, 23]. With an increase in the number of sections, the complexity of the algorithm increases nonlinearly. This is due to the fact that the transport conveyor is a complex dynamic distributed system, the output flow parameters of the sections of which contain a time-variable transport delay. The use of neural networks for modeling the state of the output flow parameters of the transport conveyor allows obtaining fairly simple approximate models, which makes it possible to design control systems for the flow parameters of the transport conveyor with satisfactory accuracy. In this regard, the issue of analyzing the factors affecting the accuracy of models of transport conveyors based on a neural network becomes relevant.

2. Problem statement

When constructing a model of a multi-section conveyor based on a neural network (multilayer perceptron), it is required to select the neural network architecture, activation function, neural network training method, neural network training rate, initial initialization method, weight coefficients. This list can be significantly expanded and detailed.

A sufficient number of papers are had in which a model of a transport system based on neural networks of different architectures is considered. The architecture of the neural network 13-5-1 (13 nodes in the input layer, 4 nodes in the hidden layer, and one node in the output layer) is used in a model for diagnosing the level of wear of a conveyor belt [17]. To develop a control system for the speed of a transport conveyor belt, a model based on a neural network with architecture 3-4-3 [24] and a model based on a neural network of three levels [25] are considered. Despite the fact that the choice of the neural network architecture has a significant impact on the learning rate and the accuracy of the predicted model of the explained parameters [26, 27, 28] when constructing the transport conveyor model, this issue is not given due attention. The choice of the neural network architecture, like the other parameters of the model listed above, is quite often made by the developer heuristically. In this regard, it becomes necessary to analyze the accuracy of the model depending on the value of the selected factor.

This study analyzes the impact of

a) selection of the bias-node value in the hidden layer of the neural network;

b) selection of the training mode (sequential or batch);

on the learning rate of the neural network and the accuracy of predicting the flow parameters of the transport conveyor model based on the neural network.

The choice was made on the specified meta-parameters of the model for the reason that, in comparison with the issue of choosing the architecture of a neural network, the issues of choosing these parameters when constructing a model of a transport system can have a significant impact on the accuracy of forecasting and are less studied.

3. Main material

3.1. General description of the transport system model

Let us analyze how the choice of the bias-node value in the hidden layer of the neural network and the choice of the training mode affect the accuracy of predicting the flow parameters of a conveyor-type transport system of a multi-section transport conveyor, the structural diagram of which is shown in Figure 1 [29, 30]. The structural diagram of the conveyor proposed for analysis and the corresponding data set were used in a number of works for a qualitative analysis of the parameters of the transport conveyor. A detailed analysis of the dataset that will be used to train the neural network in this study is presented in [30]. Each individual m–th section $(m=1..M)$ in the transport conveyor model has parameters: the section transport route length ξ_m, the material flow at the section input $\gamma_m(\tau)$, the conveyor belt speed $g_m(\tau)$ и the output material flow from the section $\theta_1(\tau,\xi_m)$. The parameters of the model are dimensionless, which makes it possible to use the theory of the similarity of production systems in the analysis of the transport conveyor. The technique for constructing a dataset is presented in the paper [30]. The data set corresponds to the case of a transport system functioning with dynamically changing flows of incoming material (Figure 2) entering the transport system, and dynamically regulable belt speed for each separate section of the conveyor (Figure 3).The flow of material from the 6-th bunker is divided into sections 7 and 8 in the ratio $\gamma_7(\tau)/\gamma_8(\tau)=2/3$. The length ξ_m of the individual sections is different, represented by proportions $\xi_1:\xi_2...\xi_8=\{1:0{,}5:0{,}7:0{,}8:1{,}5:1:1{,}5:0{,}6\}$.

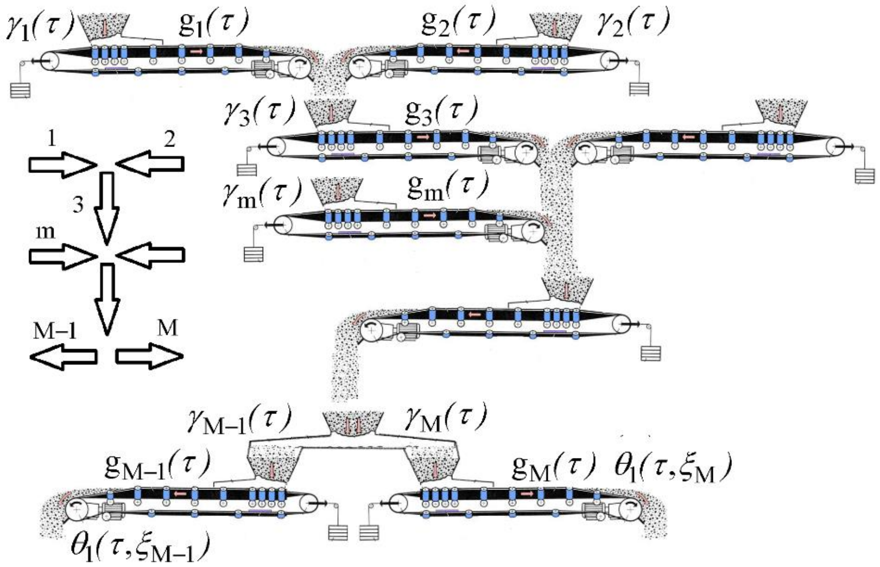

Figure 1: The structural diagram of the transport conveyor

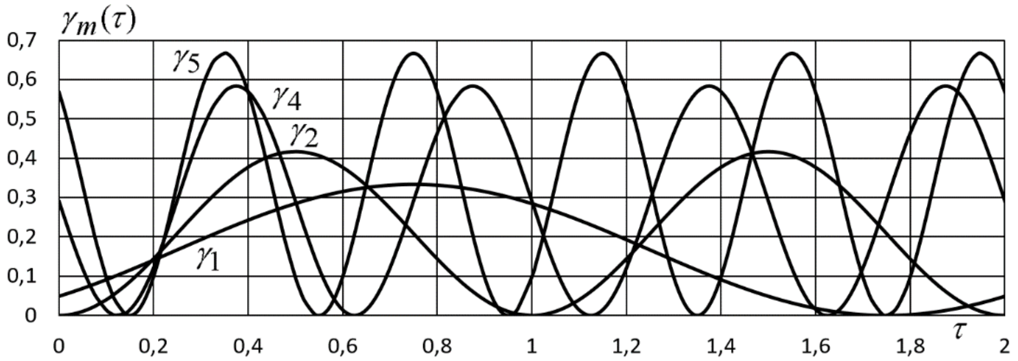

Figure 2: Input material flow $\gamma_m(\tau)$ for the m–th section of the transport conveyor

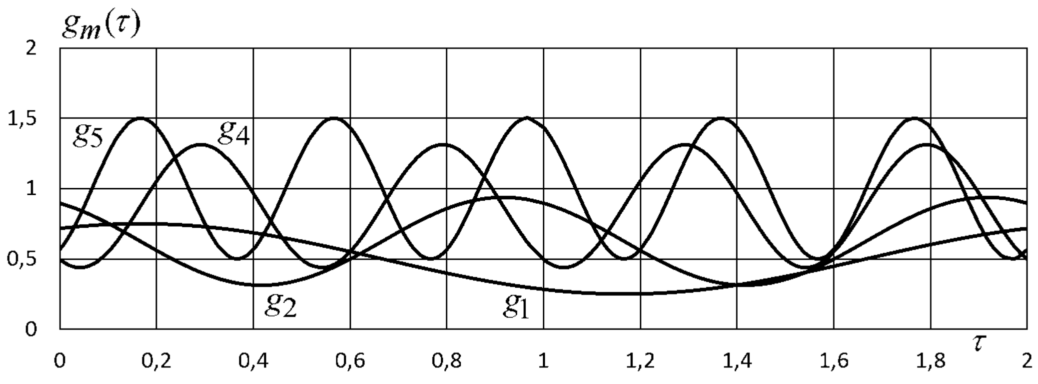

Figure 3: Belt speed $g_m(\tau)$ for the m–th section of the transport conveyor

The values of the output material flow $\theta_1(\tau,\xi_m)$ from the training dataset are characterized by the distribution function shown in Figure 4 and are determined through the values of the input material flow $\gamma_m(\tau)$ and the belt speed $g_m(\tau)$ of a separate section:

$$\theta_{1m}(\tau,1) = (1 - H(1 - G(\tau)))\frac{\gamma_m(\tau - \Delta\tau_{\xi_m})}{g_m(\tau - \Delta\tau_{\xi_m})} g_m(\tau) + \mathrm{H}(1 - G_m(\tau))\psi_m(1 - G_m(\tau)) g_m(\tau),$$

$$G_m(\tau) = \int_0^\tau g_m(\alpha) d\alpha . \quad (1)$$

Equation (1) calculates the output flow taking into account the initial distribution of material along the transport route. The initial distribution of the material $\psi_m(\xi_m)$ does not depend on the parameters $\gamma_m(\tau), g_m(\tau)$, therefore the initial distribution of the material introduces an error in forecasting [30]. In this regard, the data set for training, the characteristic of the parameters of which is shown in Figure 4 was obtained under the condition $G_m(\tau) > \xi_m$. Solving the equations

$$\xi_m = \int_{\tau - \Delta\tau_m(\tau)}^{\tau} g_m(\alpha) d\alpha \quad (2)$$

allows you to calculate the amount of transport delay $\Delta\tau_m(\tau)$ or a separate section and the conveyor as a whole.

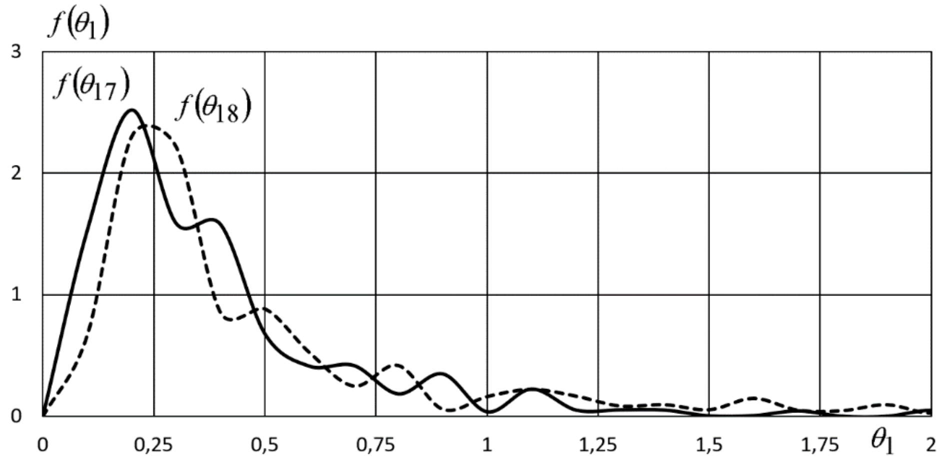

Figure 4: The density of the distribution of the belt speed $g_m(\tau)$ m–th section of the transport conveyor

To analyze the influence of the choice of the bias-node value in the hidden layer of the neural network and the choice of the training mode on the accuracy of predicting the flow parameters of the conveyor-type transport system, a neural network with the architecture 12-15-2 will be used, the input and output nodes of which are numbered as follows (Figure 5):

$$x_{3m-2} = \gamma_m(\tau), \quad x_{3m-1} = g_m(\tau), \quad x_{3m} = \xi_m, \qquad y_1 = \theta_{17}(\tau,\xi_7), \; y_2 = \theta_{18}(\tau,\xi_8). \quad (3)$$

A similar approach to the numbering of input parameters and output parameters of the transport conveyor was proposed in paper [31]. The number of nodes in the hidden layer is set in accordance with the recommendations given in [26, 27, 32, 33]. When forming a neural network, a logistic

$$f(x) = \frac{a}{1 + \exp(-bx)}, \quad a = 4, b = 1, \quad (4)$$

and linear activation function

$$f(x)=\begin{cases} f(x)=a, & if \quad bx>a, \\ f(x)=bx, & if \quad a>bx>-a, \quad a=100,\ b=0{,}5. \\ f(x)=-a, & if \quad bx<-a, \end{cases} \tag{5}$$

was used. The parameters of the activation function are determined as a result of subsequent tuning, which ensures the maximum quality of forecasting the streaming parameters of the transport system.

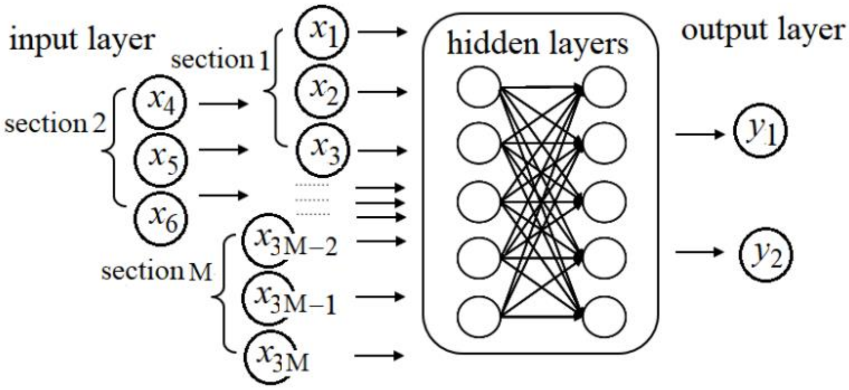

Figure 5: Input and output parameters of the neural network in the transport conveyor model

The training data set contains $N_e \approx 10^4$ examples of the sequential state of the transport system. The training of the neural network has been carried out for 10^4 э epochs. The criterion for the quality of training was the value mean squared error (MSE)

$$MSE = \frac{1}{N_e}\sum_{n=1}^{N_e}\left((y_{1n}-y_{1t\,n})^2+(y_{2n}-y_{2\,n})^2\right). \tag{6}$$

3.2. Sequential and batch modes of neural network training

The sequential neural network training method is to sequentially provide examples of epochs from the training dataset, updating the weights after each training example. For each learning epoch, examples from a set can come in a given order or in random order. The learning error for the n- th example (n=1..N) is determined by the expression

$$E_n = \frac{1}{2}\sum_{v_L=1}^{V_L} e_{Lv_L n}{}^2, \tag{7}$$

$$e_{Lv_L n} = d_{v_L n} - y_{Lv_L n}, \tag{8}$$

where $d_{v_L n}$ is the value of the output neuron in the training set; $y_{Lv_L n}$ is the generated value by the output neuron; V_L is number of nodes in the output layer. Let's assume that the neural network contains $(L+1)$ layers, each of which consists of V_l nodes, $l=0..L$. The input layer corresponds to the value $l=0$, the output layer corresponds to the value $l=L$. The value of the v_l – th node $(v_l=0..V_l)$ in the l – th layer for the n – th training example is determined by the expression

$$y_{lv_l n} = f_l\left(\sum_{v_{l-1}=0}^{V_{l-1}} w_{lv_l v_{l-1}} y_{(l-1)v_{l-1}n}\right), \quad \beta_{lv_l n} = \sum_{v_{l-1}=0}^{V_{l-1}} w_{lv_l v_{l-1}} y_{(l-1)v_{l-1}n}, \tag{9}$$

where $f_l\left(\beta_{lv_l n}\right)$ is the activation function of neurons in the $l-$th layer. The values for bias-nodes are defined as

$$y_{l0n} = y_{lb} = const\,, \quad l = 0..(L-1). \tag{10}$$

To change the weights, let's use the gradient descent method [34]

$$\Delta w_{lv_l v_{l-1}} = -\alpha \frac{\partial E_n}{\partial w_{lv_l v_{l-1}}}, \tag{11}$$

which allows us to determine a direction vector in the weight space $w_{lv_l v_{l-1}}$, that reduces the error value E_n. The parameter α is the learning rate of the error back propagation algorithm. Let's use the previously performed transformations in accordance with the notation Figure 6:

$$\frac{\partial y_{lv_l n}}{\partial w_{lv_l v_{l-1}}} = \frac{\partial f_l\left(\sum\limits_{v_{l-1}=0}^{V_{l-1}} w_{lv_l v_{l-1}} y_{(l-1)v_{l-1}n}\right)}{\partial w_{lv_l v_{l-1}}} = \frac{\partial f_l\left(\beta_{lv_l n}\right)}{\partial \beta_{lv_l n}} \frac{\partial \beta_{lv_l n}}{\partial w_{lv_l v_{l-1}}} = \frac{\partial f_l\left(\beta_{lv_l n}\right)}{\partial \beta_{lv_l n}} y_{(l-1)v_{l-1}n}, \tag{12}$$

$$\frac{\partial E_n}{\partial y_{Lv_l n}} = \frac{\partial}{\partial y_{Lv_l n}} \frac{1}{2} \sum_{v_L=1}^{V_L} {e_{Lv_L n}}^2 = e_{Lv_L n} \frac{\partial e_{Lv_L n}}{\partial y_{Lv_l n}} = e_{Lv_L n} \frac{\partial\left(d_{v_L n} - y_{Lv_L n}\right)}{\partial y_{Lv_l n}} = -e_{Lv_L n}, \tag{13}$$

$$\frac{\partial y_{l_2 0n}}{\partial w_{lv_l v_{l-1}}} = 0\,, \quad l_2 = 1..L-1\,, \quad l_2 \ge l\,, \tag{14}$$

$$\frac{\partial y_{l_2 v_{l_2} n}}{\partial w_{lv_l v_{l-1}}} = \sum_{v_{L-1}=0}^{V_{L-1}} \frac{\partial y_{l_2 v_{l_2} n}}{\partial y_{(l_2-1)v_{l_2-1}n}} \frac{\partial y_{(l_2-1)v_{l_2-1}n}}{\partial w_{lv_l v_{l-1}}} = \sum_{v_{l_2}=1}^{V_{l_2}} \frac{\partial y_{l_2 v_{l_2} n}}{\partial y_{(l_2-1)v_{l_2-1}n}} \frac{\partial y_{(l_2-1)v_{l_2-1}n}}{\partial w_{lv_l v_{l-1}}}, \tag{15}$$

$$e_{(l_2-1)v_{l_2-1}n} = \sum_{v_{l_2}=1}^{V_{l_2}} e_{l_2 v_{l_2} n} \frac{\partial y_{l_2 v_{l_2} n}}{\partial y_{(l_2-1)v_{l_2-1}n}} = \sum_{v_{l_2}=1}^{V_{l_2}} e_{l_2 v_{l_2} n} \frac{\partial f_{l_2}\left(\beta_{l_2 v_{l_2} n}\right)}{\partial \beta_{l_2 v_{l_2} n}} \frac{\partial \beta_{l_2 v_{l_2} n}}{\partial y_{(l_2-1)v_{l_2-1}n}} =$$

$$= \sum_{v_{l_2}=1}^{V_{l_2}} e_{l_2 v_{l_2} n} \frac{\partial f_{l_2}\left(\beta_{l_2 v_{l_2} n}\right)}{\partial \beta_{l_2 v_{l_2} n}} w_{(l_2-1)v_{l_2} v_{l_2-1}n} = \sum_{v_{l_2}=1}^{V_{l_2}} e_{l_2 v_{l_2} n} \frac{\partial f_{l_2}\left(\beta_{l_2 v_{l_2} n}\right)}{\partial \beta_{l_2 v_{l_2} n}} w_{(l_2-1)v_{l_2} v_{l_2-1}n}, \tag{16}$$

and obtain an expression for determining the gradient of the function $E_n\left(w_{lv_l v_{l-1}}\right)$ expressed through the weight coefficients $w_{lv_l v_{l-1}}$

$$\frac{\partial E_n}{\partial w_{lv_l v_{l-1}}} = \sum_{v_L=1}^{V_L} \frac{\partial E_n}{\partial y_{Lv_L n}} \frac{\partial y_{Lv_L n}}{\partial w_{lv_l v_{l-1}}} = -\sum_{v_L=1}^{V_L} e_{Lv_L n} \frac{\partial y_{Lv_L n}}{\partial w_{lv_l v_{l-1}}} = -\sum_{v_L=1}^{V_L} e_{Lv_L n} \sum_{v_{L-1}=1}^{V_{L-1}} \frac{\partial y_{Lv_l n}}{\partial y_{(L-1)v_{L-1}n}} \frac{\partial y_{(L-1)v_{L-1}n}}{\partial w_{lv_l v_{l-1}}} =$$

$$= -\sum_{v_{L-1}=1}^{V_{L-1}} \frac{\partial y_{(L-1)v_{L-1}n}}{\partial w_{lv_l v_{l-1}}} \sum_{v_L=1}^{V_L} e_{Lv_L n} \frac{\partial y_{Lv_l n}}{\partial y_{(L-1)v_{L-1}n}} = -\sum_{v_{L-1}=1}^{V_{L-1}} e_{(L-1)v_{L-1}n} \frac{\partial y_{(L-1)v_{L-1}n}}{\partial w_{lv_l v_{l-1}}} = -\sum_{v_{L-2}=1}^{V_{L-2}} e_{(L-2)v_{L-2}n} \frac{\partial y_{(L-2)v_{L-2}n}}{\partial w_{lv_l v_{l-1}}} =$$

$$= -\sum_{v_{l_2}=1}^{V_{l_2}} e_{l_2 v_{l_2} n} \frac{\partial y_{l_2 v_{l_2} n}}{\partial w_{l v_l v_{l-1}}} = -e_{l v_l n} \frac{\partial y_{l v_l n}}{\partial w_{l v_l v_{l-1}}} = -e_{l v_l n} \frac{\partial f_l\left(\beta_{l v_l n}\right)}{\partial \beta_{l v_l n}} y_{(l-1) v_{l-1} n}.$$

Thus, the gradient of the function $E_n\left(w_{l v_l v_{l-1}}\right)$ has the form

$$\frac{\partial E_n}{\partial w_{l v_l v_{l-1}}} = -e_{l v_l n} \frac{\partial f_l\left(\beta_{l v_l n}\right)}{\partial \beta_{l v_l n}} y_{(l-1) v_{l-1} n}, \tag{17}$$

and can be determined, if the activation function $f_l(\beta)$ for l – th layer, the values of the nodes of the previous layer $y_{(l-1) v_{l-1} n}$, the values of the weight coefficients $w_{l v_l v_{l-1}}$ and the errors $e_{l v_l n}$ for the nodes l – th layer are known

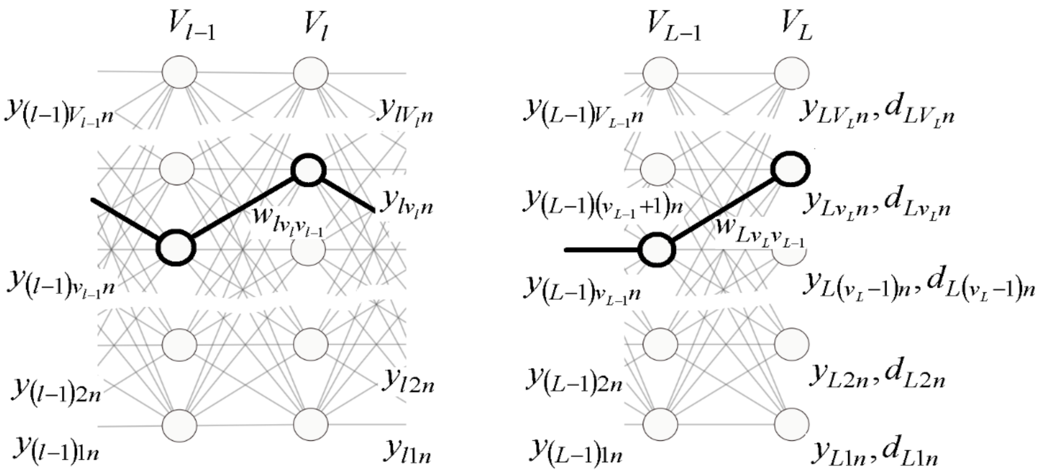

Figure 6: The schema of nodes for calculating the gradient of a function $E_n\left(w_{l v_l v_{l-1}}\right)$

When conducting a multiple experiments with the aim of repeating it, we will use the same sequence of arrival of examples at the input of the neural network within each epoch. This sequence will be the same for each variant of the experiment carried out in this study. In batch mode, the weights are adjusted after a separate batch set of examples is fed to the input of the neural network within an epoch. The cost function is defined as the mean square error for all test cases within a particular epoch

$$E_n = \frac{1}{2N} \sum_{n=1}^{N} \sum_{v=1}^{V} e_{vn}{}^2, \tag{18}$$

By analogy with the sequential training mode, let's define the gradient of the function $E_n\left(w_{l v_l v_{l-1}}\right)$ or the batch training mode:

$$\frac{\partial E_n}{\partial w_{l v_l v_{l-1}}} = \frac{1}{N} \sum_{n=1}^{N} \sum_{v_L=1}^{V_L} \frac{\partial E_n}{\partial y_{L v_L n}} \frac{\partial y_{L v_L n}}{\partial w_{l v_l v_{l-1}}} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{v_L=1}^{V_L} e_{L v_L n} \frac{\partial y_{L v_L n}}{\partial w_{l v_l v_{l-1}}} =$$

$$= -\frac{1}{N} \sum_{n=1}^{N} \sum_{v_L=1}^{V_L} e_{L v_L n} \sum_{v_{L-1}=1}^{V_{L-1}} \frac{\partial y_{L v_l n}}{\partial y_{(L-1) v_{L-1} n}} \frac{\partial y_{(L-1) v_{L-1} n}}{\partial w_{l v_l v_{l-1}}} = -\frac{1}{N} \sum_{n=1}^{N} \sum_{v_{L-1}=1}^{V_{L-1}} \frac{\partial y_{(L-1) v_{L-1} n}}{\partial w_{l v_l v_{l-1}}} \sum_{v_L=1}^{V_L} e_{L v_L n} \frac{\partial y_{L v_l n}}{\partial y_{(L-1) v_{L-1} n}} =$$

$$= -\frac{1}{N} \sum_{n=1}^{N} \sum_{v_{L-1}=1}^{V_{L-1}} e_{(L-1) v_{L-1} n} \frac{\partial y_{(L-1) v_{L-1} n}}{\partial w_{l v_l v_{l-1}}} = -\frac{1}{N} \sum_{v_{L-2}=1}^{V_{L-2}} e_{(L-2) v_{L-2} n} \frac{\partial y_{(L-2) v_{L-2} n}}{\partial w_{l v_l v_{l-1}}} =$$

$$= -\frac{1}{N}\sum_{n=1}^{N}\sum_{v_{l_2}=1}^{V_{l_2}} e_{l_2 v_{l_2} n}\frac{\partial y_{l_2 v_{l_2} n}}{\partial w_{l v_l v_{l-1}}} = -\frac{1}{N}\sum_{n=1}^{N} e_{l v_l n}\frac{\partial y_{l v_l n}}{\partial w_{l v_l v_{l-1}}} = -\frac{1}{N}\sum_{n=1}^{N} e_{l v_l n}\frac{\partial f_l\left(\beta_{l v_l n}\right)}{\partial \beta_{l v_l n}} y_{(l-1) v_{l-1} n}\,.$$

The final expression for calculating the gradient of the function $E_n\left(w_{l v_l v_{l-1}}\right)$ for the batch training mode

$$\frac{\partial E_n}{\partial w_{l v_l v_{l-1}}} = -\frac{1}{N}\sum_{n=1}^{N} e_{l v_l n}\frac{\partial f_l\left(\beta_{l v_l n}\right)}{\partial \beta_{l v_l n}} y_{(l-1) v_{l-1} n}\,, e_{l v_l n} = \sum_{v_{l+1}=1}^{V_{l+1}} e_{(l+1) v_{l+1} n}\frac{\partial f_{l+1}\left(\beta_{(l+1) v_{l+1} n}\right)}{\partial \beta_{(l+1) v_{l+1} n}} w_{(l+1) v_{l+1} v_l n} \tag{19}$$

is similar to the expression for the sequential training mode (11).

Averaging over the number of training examples, on the one hand, makes it possible to give a more accurate estimate of the gradient vector, and, accordingly, simplifies the process of obtaining convergence conditions for the neural network training algorithm, on the other hand, it requires additional computational memory to store the error calculation results $e_{l v_l n}$ and the node value $y_{(l-1) v_{l-1} n}$ or each test case as part of a separate package. Another disadvantage associated with the considered batch learning mode algorithm is the higher probability of stopping the learning algorithm at a local minimum point. When using batch training mode, preprocessing of the training set is required in order to eliminate duplicate training examples. The advantage of the batch mode algorithm is the ability to parallelize calculations since, for each training example, the values of the weight coefficients remain constant.

4. Analysis of results

4.1. Comparative analysis of the results of predicting the output flow parameters of the transport conveyor using sequential mode and batch mode of neural network training

The results of training a neural network for a batch training mode with one, two, four, and eight examples in a batch are shown in Figure 7. The calculation of the gradient was carried out in accordance with expression (19). The learning rate for each mode is selected from the condition

$$\alpha = 0{,}01 N_{batch}\,, \tag{20}$$

where N_{batch} is the number of examples in one batch. Thus, for a numerical experiment with a batch consisting of eight elements, the learning rate was set eight times higher $(\alpha = 0{,}08)$, than for a numerical experiment with a package consisting of one element $(\alpha = 0{,}01)$.

The analysis of the results of the numerical experiment presented in Figure 7 shows that despite the fact that with an increase in the batch size N_{batch} , the learning rate increases proportionally α (20), the number of epochs to achieve the same value MSE increases nonlinearly. This is clearly shown in Figure 8 in the form of equipotential lines $MSE = f\left(N_{batch}, \lg_{10}(epoch)\right)$. Analysis of the results of computational experiments shows the inefficiency of using the batch mode of training a neural network in a distributed model of the transport conveyor, confirming the conclusions made in the works about the general inefficiency of batch training for gradient descent learning [35, 36].

So, for example, the value $MSE = 0{,}2$ for batch size $N_{batch} = 1$ $(\alpha = 0{,}01 N_{batch} = 0{,}01)$ is reached at 12th learning epoch, for $N_{batch} = 2$ $(\alpha = 0{,}02)$ is reached at 19th learning epoch, for $N_{batch} = 4$ $(\alpha = 0{,}04)$ is reached at 113th learning epoch, and finally, for $N_{batch} = 8$ $(\alpha = 0{,}08)$ are required 10^4 learning epoch.

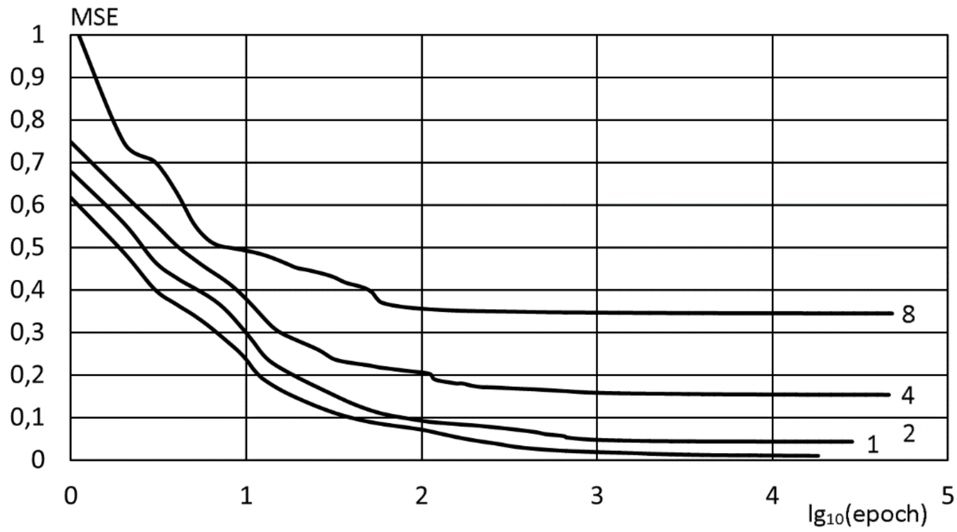

Figure 7: A criterion for the quality of training a neural network with a batch size N_{batch}, consisting of one, two, four, and eight examples with a learning rate $\alpha = 0{,}01N_{batch}$.

When parallelizing the training process for the batch mode with the number of elements in the batch $N_{batch} = 8$, the number of training epochs is required $N_{batch}^3 \sim 10^3$ times more than in the sequential training mode. In both cases, the same $MSE = 0{,}2$ is achieved.

An increase in the convergence of the learning process due to a proportional increase in the learning rate $\left(\alpha = 0{,}01N_{batch}^4\right)$ leads to the development of instability in the learning process and does not give the desired results.

Thus, a numerical experiment shows that when training a transport system model based on a neural network, the batch training mode has slow convergence at a given training rate proportional to the number of elements N_{batch} in the batch.

Figure 9 and Figure 10 show the results of predicting the output flow $\theta_{17}(\tau,\xi_7)$ and $\theta_{18}(\tau,\xi_8)$ depending on the number of training epochs. The prediction quality results are determined by the MSE (6), which corresponds to 0,617; 0,197 and 0,045 after one, twelve, and one hundred and eighty epochs of learning. The sequential neural network training mode demonstrates satisfactory convergence of the neural network training process and high training accuracy.

4.2. Analysis of the choice of the initial value of the node participating in the formation of the bias

The value of the hidden node is determined by expressions (9) and (10). The value of the node participating in the formation of the bias value, as a rule, is taken equal to one: $y_{l0n} = 1$ (10). This approach is common, despite the fact that the choice of value y_{l0n} can have a significant impact on the convergence of the neural network learning process.

The values of the nodes of the neural network are calculated during the forward pass based on the known values of the weight coefficients (9).

In the reverse pass, the values of the weight coefficients are calculated taking into account the values of the nodes of the neural network (11)-(19). A self-consistent change in the weight coefficients and values of the neural network nodes is observed. If the learning process is convergent, then the values of the neural network nodes and the values of the weight coefficients converge to their steady-state values. For hidden layers, the node value y_{l0n} is constant, not formed as a result of a direct pass of the neural network.

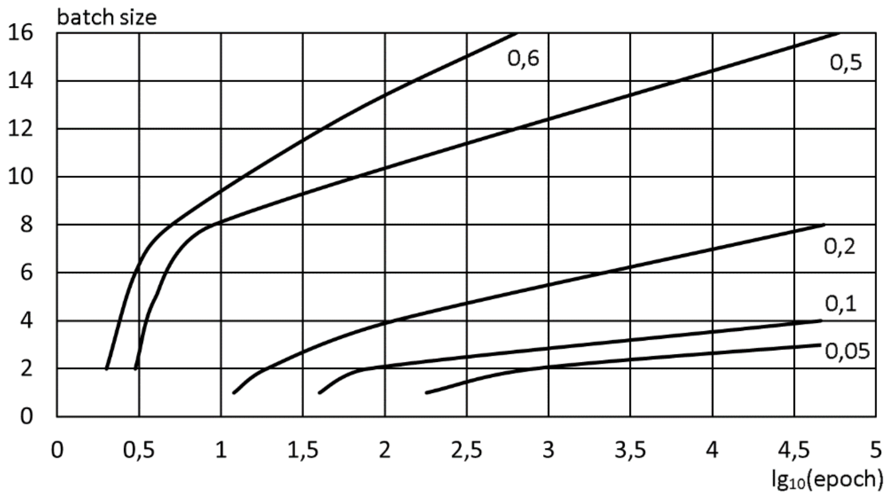

Figure 8: Equipotential lines $MSE = f\left(N_{batch}, \lg_{10}(epoch)\right) = \{0{,}05;0{,}1;0{,}2;0{,}5;0{,}6\}$

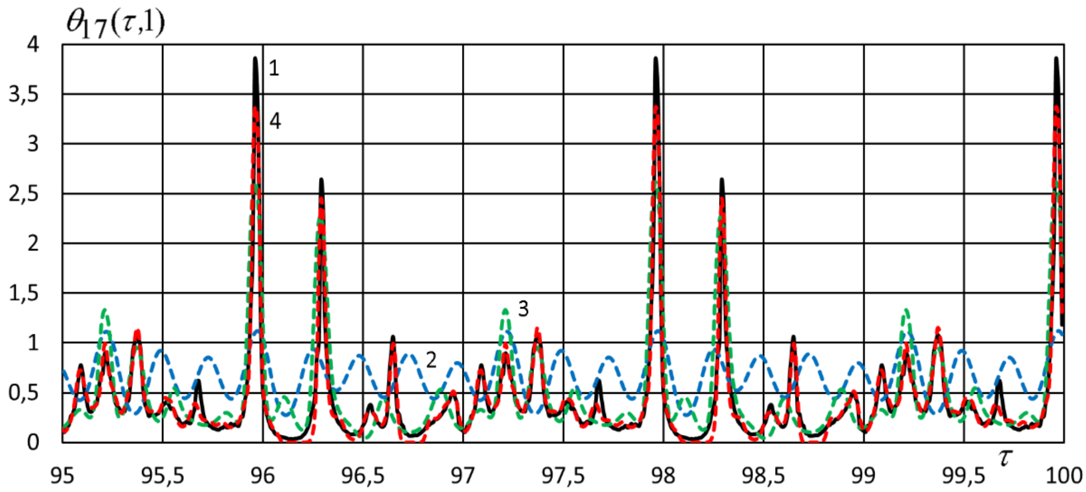

Figure 9: Predicting the value of the output flow $y_1 = \theta_{17}(\tau, \xi_7)$ using sequential learning mode. 1: trainable set; 2: after one learning epoch; 3: after twelve epochs of learning; 4: after one hundred and eighty eras of learning

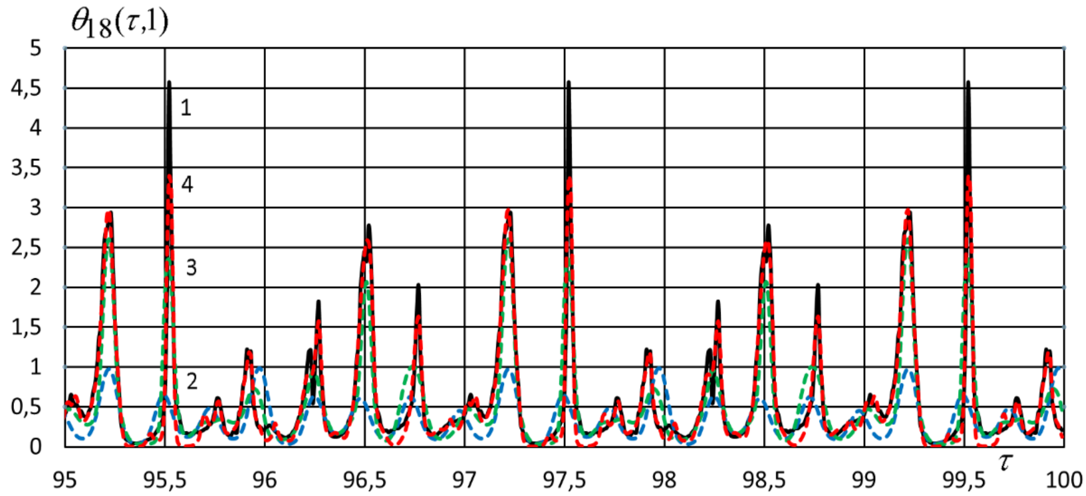

Figure 10: Predicting the value of the output flow $y_2 = \theta_{18}(\tau, \xi_8)$ using sequential learning mode. 1: trainable set; 2: after one learning epoch; 3: after twelve epochs of learning; 4: after one hundred and eighty eras of learning

However, the node value y_{l0n} participates in the process of forming the value of other nodes during the forward pass and in the process of forming the values of the weight coefficients during the reverse pass. If, as a result of training, the values of the hidden nodes $y_{(l-1)v_{l-1}n}$ are small

$$y_{l0n} = 1, \quad \left| y_{(l-1)v_{l-1}n} \right| << 1, \qquad v_{l-1} > 0,$$

then with the same order of values of the weight coefficients $w_{lv_l 0} \approx w_{lv_l v_{l-1}}$ the calculated value of the node is determined by the bias value

$$\beta_{lv_l n} = \sum_{v_{l-1}=0}^{V_{l-1}} w_{lv_l v_{l-1}} y_{(l-1)v_{l-1}n} \approx w_{lv_l 0} y_{(l-1)0n} \approx w_{lv_l 0}, \qquad (21)$$

The experiments carried out for the analyzed neural network with the architecture 12-15-2 (3) confirm the influence of the value of the node participating in the formation of the bias value on the learning process of the neural network in the transport conveyor model. The analysis was carried out for a neural network with logistic activation function and linear activation function. The analysis results are presented in Table 1. Figures 11 and 12 demonstrate the convergence of the learning process for a neural network with logistic activation function and linear activation function with the same architecture of the 12-15-2 neural network (3). In these figures, the solid line represents the learning process for positive values $y_{(l-1)0n}$ and the dashed line for negative values $y_{(l-1)0n}$. It should be noted that the dashed line $MSE\left(y_{(l-1)0n}\right)$, $y_{(l-1)0n} < 0$ with the logistic activation function passes in the vicinity of the solid line $MSE\left(y_{(l-1)0n}\right), y_{(l-1)0n} > 0$. The dependence $MSE\left(y_{(l-1)0n}\right)$ for the logistic activation function is shown in Figure 13.

Table 1

Influence of the bias value of a node on the convergence of the learning process of a neural network (speed learning=0,01; number of epochs=10^4)

Bias (hidden layer) $y_{(l-1)0n}$	Logistic Activation function		Linear Activation function			
	MSE	Number of epochs	MSE	Number of epochs	MSE	Number of epochs
-10,0	0,017074	10^4	0,398227	4	0,516921	10^4
-5,0	0,009092	10^4	0,515637	771	0,516921	10^4
-2,0	0,008957	10^4	0,494928	567	0,496836	10^4
-1,0	0,010552	10^4	0,490902	729	0,492584	10^4
0,0	0,012328	10^4	0,498925	519	0,500923	10^4
1,0	0,011143	10^4	0,490904	731	0,492578	10^4
2,0	0,011278	10^4	0,494926	592	0,496836	10^4
5,0	0,008333	10^4	0,515766	757	0,511629	10^4
10,0	0,015051	10^4	0,397910	4	0,511629	10^4

For the linear activation function, the solid and dashed lines coincide and are visually indistinguishable in Figure 12. The dependence $MSE\left(y_{(l-1)0n}\right)$ for the logistic activation function is shown in Figure 14. For a nonlinear activation function, with an increase in the number of learning epochs, the sensitivity of the quality of the learning process to the bias-node value decreases (Figure 13). For a linear function, the decrease in sensitivity is much slower (Figure 14). For two cases, the symmetry of the function $MSE\left(y_{(l-1)0n}\right)$ with respect to the zero value is observed. The minimum value of the function $MSE\left(y_{(l-1)0n}\right)$ is reached at a point different from the value $y_{(l-1)0n} = 0$. This indicates that the presence of a nonzero bias improves the accuracy of predicting the output material flow.

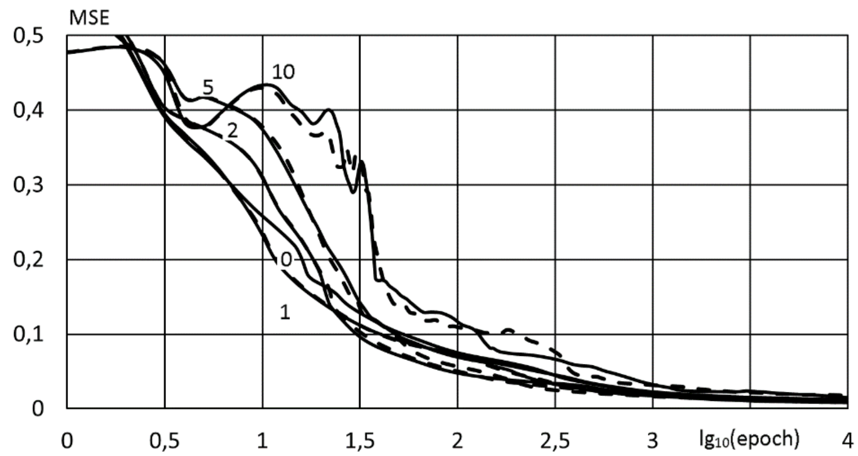

Figure 21: MSE value depending on the number of learning epochs for the bias value $\left|y_{(l-1)0n}\right| = \{0;1;2;5;10\}$ (logistic activation function and 12-15-2 neural network architecture)

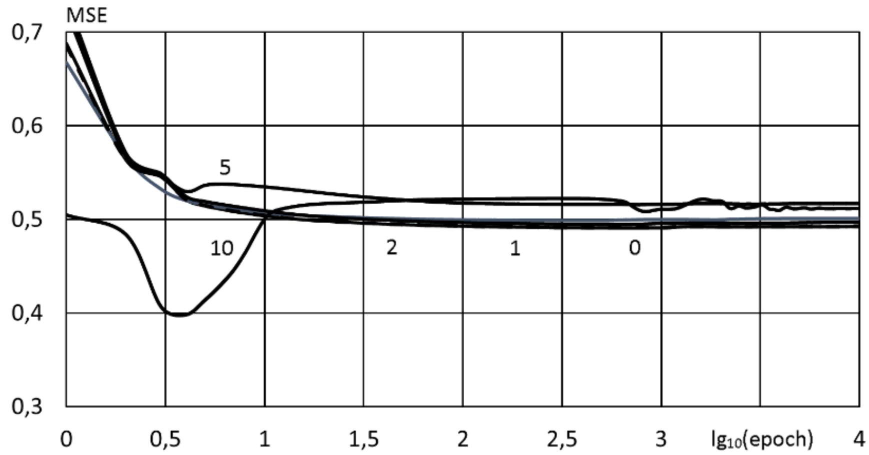

Figure 12: MSE value depending on the number of learning epochs for the bias value $\left|y_{(l-1)0n}\right| = \{0;1;2;5;10\}$. (linear activation function and 12-15-2 neural network architecture)

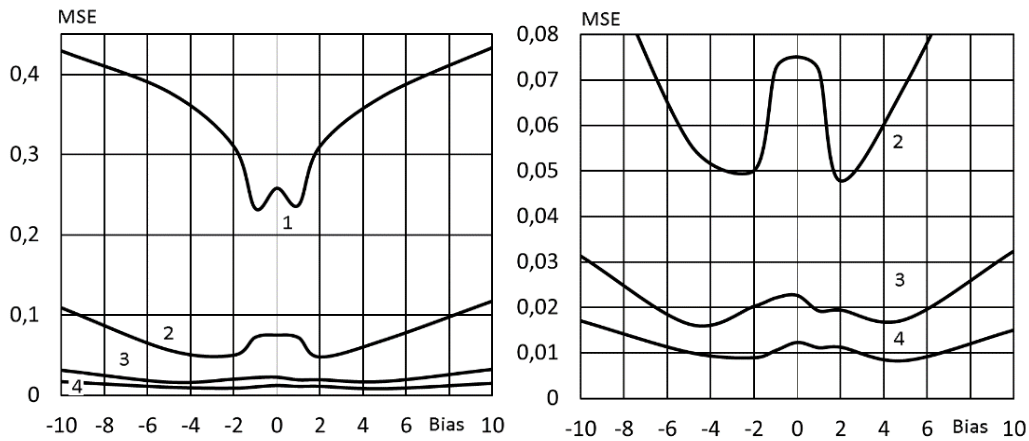

Figure 13: The dependence $MSE\left(y_{(l-1)0n}\right)$ for lg10(epoch)= $\{1;2;3;4\}$ (logistic activation function and 12-15-2 neural network architecture)

4.3. Conclusion

In this work, we analyzed the choice of the neural network training mode, which is used in the transport conveyor model. For comparative analysis, a neural network with architecture 12-15-2 was used. The experiment was carried out for a neural network with a logistic activation function and a linear activation function of its nodes.

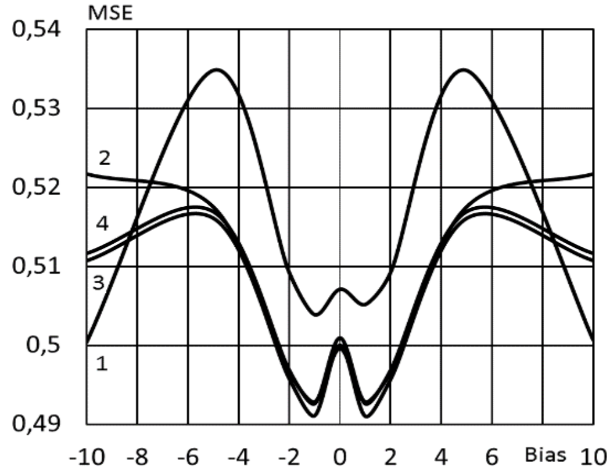

Figure 14: The dependence $MSE\left(y_{(l-1)0n}\right)$ for $\lg_{10}$(epoch)= $\{1;2;3;4\}$ (linear activation function and 12-15-2 neural network architecture)

The learning rate for the analyzed batches is chosen proportional to the size of the training batch. For example, for a batch learning mode of size eight, the learning rate is chosen to be eight times faster than the sequential learning mode. With the same number of epochs, a linear increase in the MSE value is observed depending on the size of the training batch. Thus, at the same learning rate, there is a quadratic dependence of the MSE value on the size of the training batch for the selected neural network architecture. Switching to batch training mode significantly reduces the efficiency of neural network training. It should be noted that there is a quasi-linear relationship between the size of the training batch and the $\lg_{10}$(epoch), value, which characterizes the number of training epochs. In addition, attention is paid to the choice of the value of the node of the hidden layer participating in the formation of the bias. Numerical experiments demonstrate the need to use bias to improve the accuracy of predicting the output flow parameters of the transport system.

Ensuring the convergence of the learning process is one of the important problems in the design of transport conveyor models based on a neural network. The use of the batch mode guarantees the convergence of the algorithm to a local minimum under fairly simple conditions, which explains its use in proving the convergence of the back propagation algorithm. Convergence is increased by parallelizing the learning process. However, as shown in this work, the use of the batch learning mode is ineffective when constructing models for predicting the output flow of material in the multi-section transport conveyor. This is explained by the fact that the multi-section transport conveyor is a complex dynamic distributed system, at the input of which material flow arrives, the value of which is of a stochastic nature. The superposition of flows of individual sections with a transport delay leads to the presence of characteristic peaks in the value of the output flow. Under such conditions, the use of the batch mode of training the neural network leads to a strong smoothing of the peaks, decreasing the value in proportion to the number of examples in the batch, and as a consequence, significantly increases the MSE value, reducing the quality of the prediction of the values of the output flow parameters of the transport system. The peculiarity of the multi-section transport system associated with the presence of peak values as a result of the superposition of material flows of individual sections limits the possibility of using the batch mode of training the neural network when constructing models of the transport conveyor. The same reason can be accepted as an explanation for the low efficiency of the use of a linear activation function in the formation of a neural network. The scientific novelty of the results was obtained in the fact that for the first time the problem of increasing the efficiency of the learning process of a neural network used to build a model of a multi-section transport conveyor was posed. An estimate is given for the decrease in the convergence rate of the neural network learning process depending on the increase in the number of examples in the training batch. Recommendations on the choice of the activation function and bias value for the neural network nodes, as well as the neural network training mode, are presented. These recommendations are valid for multi-section transport systems with ten or more sections. With a small number of sections, the peaks in the output stream are not so pronounced, which allows you to select both sequential learning mode and batch learning mode.

The prospect of further research is the analysis of the choice of the training mode for neural network used in transport conveyor models, depending on its architecture. Of particular interest is the case of using multilayer neural network in models of transport systems. Also, an interesting direction for future research is the influence of the type of activation function used in the neural network on the quality of predicting the flow parameters of the transport system.

5. References

[1] M. Bebic, B. Ristic, Speed controlled belt conveyors: drives and mechanical considerations, Advances in Electrical and Computer Engineering, 18(1) (2018) 51–60. http//dx.doi.org/10.4316/AECE.2018.01007

[2] I. Halepoto, M. Uqaili, Design and implementation of intelligent energy efficient conveyor system model based on variable speed drive control and physical modeling, International Journal of Control and Automation, 9(6) (2016) 379–388. http//dx.doi.org/10.14257/ijca.2016.9.6.36

[3] O. Pihnastyi, G. Kozhevnikov, V. Khodusov, Conveyor model with input and output accumulating bunker, in: Proceedings of 11th International Conference on Dependable Systems, Services and Technologies, 2020, pp.253–258. http://dx.doi.org/10.1109/DESSERT50317.2020.9124996

[4] P. Bardzinski, P. Walker, W. Kawalec, Simulation of random tagged ore flow through the bunker in a belt conveying system, International Journal of Simulation Modelling 4 (2018) 597–608. https://doi.org/10.2507/IJSIMM17(4)445

[5] A. Reutov, Simulation of load traffic and steeped speed control of conveyor, in: Proceedings of IOP Conf. Series: Earth and Environmental, 2017, pp. 1–4. https://doi.org/10.1088/1755-1315/87/8/082041

[6] E. Wolstenholm, Designing and assessing the benefits of control policies for conveyor belt systems in underground mines, Dynamica 6(2) (1980) 25–35

[7] B. Karolewski, P. Ligocki, Modelling of long belt conveyors, Maintenance and reliability 16(2), (2014) 179–187. http://yadda.icm.edu.pl/yadda/element/bwmeta1.element.baztech-ce355084-3e77-4e6b-b4b5-ff6131e77b30

[8] O. Pihnastyi, Control of the belt speed at unbalanced loading of the conveyor, Scientific bulletin of National Mining University 6 (2019) 122–129. https://doi.org/10.29202/nvngu/2019-6/18

[9] A. Semenchenko, M. Stadnik, P. Belitsky, D. Semenchenko, O. Stepanenko, The impact of an uneven loading of a belt conveyor on the loading of drive motors and energy consumption in transportation, Eastern-European Journal of Enterprise Technologies (82) (2016) 42–51. https://doi.org/10.15587/1729-4061.2016.75936

[10] R Kiriia, L. Shyrin, Reducing the energy consumption of the conveyor transport system of mining enterprises. in: Proceedings of international conference essays of mining science and practice, 109 (2019). https://doi.org/10.1051/e3sconf/201910900036

[11] Siemens. Innovative solutions for the mining industry, 2021. URL: www.siemens.com/mining

[12] M. Koman, Z. Laska, The constructional solution of conveyor system for reverse and bifurcation of the ore flow, Rudna mine KGHM Polska Miedź SA, CUPRUM 3(72) (2014) 69–82. http://www.czasopismo.cuprum.wroc.pl/journal-articles/download/113

[13] O. Pihnastyi, V. Khodusov, Calculation of the parameters of the composite conveyor line with a constant speed of movement of subjects of labour, Scientific bulletin of National Mining University 4(166) (2018) 138–146. https://doi.org/10.29202/nvngu/2018-4/18

[14] M. Andrejiova, D. Marasova, Using the classical linear regression model in analysis of the dependences of conveyor belt life, Acta Montanistica Slovaca 18(2) (2013) 77–84. https://actamont.tuke.sk/pdf/2013/n2/2andrejiova.pdf

[15] Y. Lu, Q. Li, A regression model for prediction of idler rotational resistance on belt conveyor, Measurement and Control 52(5) (2019) 441–448. https://doi.org/10.1177/0020294019840723

[16] B. Karolewski, D. Marasova, Experimental research and mathematical modelling as an effective tool of assessing failure of conveyor belts, Maintenance and reliability 16(2) (2014) 229–235. http://www.ein.org.pl/sites/default/files/2014-02-09.pdf

[17] A. Kirjanów, The possibility for adopting an artificial neural network model in the diagnostics of conveyor belt splices, Interdisciplinary issues in mining and geology 6 (2016) 1–11.
[18] D. Więcek, A. Burduk, I. Kuric, The use of ANN in improving efficiency and ensuring the stability of the copper ore mining process, Acta Montanistica Slovaca 24(1) (2019) 1–14. https://actamont.tuke.sk/pdf/2019/n1/1wiecek.pdf
[19] E. Klippel, R. Oliveira, D. Maslov, A. Bianchi, S. Silva, C. Garrocho, Embedded Edge Artificial Intelligence for Longitudinal Rip Detection in Conveyor Belt Applied at the Industrial Mining Environment, in: Proceedings of the 23rd International Conference on Enterprise Information Systems, ICEIS, ISBN 978-989-758-509-8, ISSN 2184-4992, 2021, pp 496–505. https://www.scitepress.org/Papers/2021/104472/
[20] M. Bajda, Błażej R., Jurdziak L. Analysis of changes in the length of belt sections and the number of splices in the belt loops on conveyors in an underground mine, Engineering Failure Analysis 101 (2019) 439–446. https://doi.org/10.1016/j.engfailanal.2019.04.003
[21] .R. Król, W. Kawalec, L. Gładysiewicz, An effective belt conveyor for underground ore transportation systems, in: Proceedings of IOP Conf. Series: Earth and Environmental Science, 2017, 95(4) 1–9. https://doi.org/10.1088%2F1755-1315%2F95%2F4%2F042047
[22] O. Pihnastyi, V. Khodusov, Optimal Control Problem for a Conveyor-Type Production Line, Cybern. Syst. Anal.. Springer Science+Business Media, LLC 54(5) (2018) 744–753. https://doi.org/10.1007/s10559-018-0076-2
[23] O. Pihnastyi, Control of the belt speed at unbalanced loading of the conveyor, Scientific bulletin of National Mining University, 6 (2019) 122–129. https://doi.org/10.29202/nvngu/2019-6/18
[24] Xi. Pingyuan, S. Yandong, Application Research on BP Neural Network PID Control of the Belt Conveyor, JDIM 9(6) (2011) 266–270.
[25] I. Shareef, H. Hussein, Implementation of artificial neural network to achieve speed Control and Power Saving of a Belt Conveyor System. Eastern-European Journal of enterprise technologies 2(110) (2021) 44–53, 2021. https://doi.org/10.15587/1729-4061.2021.224137
[26] M. Paul, M. Chakraborty, Observation on training neural network for diagnosing schizophrenia. International Journal of Advanced Research in Computer Science. 9(1) (2018) 163–166. http://dx.doi.org/10.26483/ijarcs.v9i1.5279. http://ijarcs.info/index.php/Ijarcs/article/view/5279
[27] K. Yotov, E. Hadzhikolev, S. Hadzhikoleva. “Determining the Number of Neurons in Artificial Neural Networks for Approximation, Trained with Algorithms Using the Jacobi Matrix.” TEM Journal (2020): 1320-1329. https://www.temjournal.com/content/94/TEMJournalNovember2020_1320_1329.pdf
[28] A. Kirjanów-Błażej, A. Rzeszowska, Conveyor Belt Damage Detection with the Use of a Two-Layer Neural Network, Applied Sciences 11(12) 5480 (2021) 1–12. https://doi.org/10.3390/app11125480
[29] O. Pihnastyi, V. Khodusov, Neural model of conveyor type transport system. Proceedings of the third international workshop on computer modeling and intelligent systems, in: S. Subbotin (Eds.), Zaporizhzhia Ukraine, 2020, pp.804–818. http://ceur-ws.org/Vol-2608/paper60.pdf
[30] O. Pihnastyi, V. Khodusov, S. Subbotin, Linear Regression Model of the conveyor type transport system. Proceedings of the 9th International Conference "Information Control Systems & Technologies", Odessa, Ukraine, September 24–26, 2020, published on CEUR Workshop Proceedings (CEUR-WS.org, ISSN 1613-0073), Vol-2711, pp. 468–481
[31] J. Baek, Y. Choi, Deep Neural Network for Ore Production and Crusher Utilization Prediction of Truck Haulage System in Underground Mine. Applied Sciences; 9(19):4180 (2019) 1-21 https://doi.org/10.3390/app9194180
[32] K. Sheela, S. Deepa, Review on Methods to Fix Number of Hidden Neurons in Neural Networks, Mathematical Problemsin Engineering, (2013) 1–11. https://doi.org/10.1155/2013/425740
[33] D. Stathakis, How many hidden layers and nodes? International Journal of Remote Sensing, 30 (2009) 2133–2147. http://dx.doi.org/10.1080/01431160802549278
[34] A. Sarapardeh, A. Nait, A. Hajirezaie, Applications of Artificial Intelligence Techniques in the Petroleum Industry. Gulf Professional Publishing. (2020) ISBN: 9780128186800 . P. 322
[35] N. Keskar, J. Nocedal, D. Mudigere, M. Smelyanskiy, On large-batch training for deep learning: Generalization gap and sharp minima. 2017. Paper presented at 5th International Conference on Learning Representations, ICLR 2017, Toulon, France. https://arxiv.org/abs/1609.04836
[36] D. Wilson, T. Martinez, The general inefficiency of batch training for gradient descent learning, Neural Networks, 16(10) (2003) 1429–1451. https://doi.org/10.1016/S0893-6080(03)00138-2