An analysis of training of a multilayer perceptron for calculating the flow parameters of a conveyor

Oleh Pihnastyi[1,†], Victoriya Usik[2,†] and Georgii Kozhevnikov[1,*,†]

[1] National Technical University "KhPI", 2, Kyrpychova str., Kharkiv, 61002, Ukraine
[2] Kharkiv National University of Radio Electronics, 14, Nauky Ave., Kharkiv, 61166, Ukraine

Abstract
The paper examines the methodology for modeling many-section conveyor systems using a neural network architecture based on the multilayer perceptron model. Modeling such systems are analyzed, and a rationale for the use of neural networks in the development of systems for controlling they flow parameters is described. Conditions for using such models are determined. A model of a multi-section transport conveyor is developed. To correct synaptic weights when training a neural network in accordance with the method of minimizing the root mean square error, the backpropagation algorithm is used. For nodes in each hidden layer, the same type of activation function is specified, which characterizes the nonlinearity of the layer. The weights are adjusted during the training period for each training example. Initialization of the weights of the neural network is carried out using a pseudorandom number generator, which provides for the possibility of repeating the experiment many times with different activation functions and hyper-parameters for training. Strategies for initializing the weighting coefficients of neural networks are presented and recommendations for the initialization process are given. To train the neural network is used a data set generated using an analytical model for an eight-section transport conveyor. An analysis of the main numerical characteristics of material flow in transport systems used as a training data set for a neural network is presented and the need for data normalization is substantiated. The speed and accuracy of training for multilayer perceptron of different architectures and with different types of activation functions are analyzed. The analysis was performed for both the hidden layer nodes and the output layer.

Keywords
multi-section conveyor, transport delay, belt speed control, conveyor model

1. Introduction

The natural resource extraction industry continues to actively develop to obtain valuable resources. This industry is characterized by aggressive mining conditions, labor-intensive work and huge volumes of cargo transportation. Its effectiveness directly depends on the equipment used. One of the main types of equipment needed in the mining industry is a belt conveyor. Its difference from other transport systems (TS) lies in its continuous operation and ability to carry impressive loads through hard-to-reach places.

This work is devoted to the construction of models of multi-sectional TS based on the use of multilayer feed-forward networks, is a continuation of studies [1,2], in which a multilayer perceptron is applied to predict the flow parameters of a transport system. We consider a multilayer perceptron (MLP) as a type of artificial neural network consisting of multiple layers of neurons.

2. Literature review

The transportation of material in the mining industry makes about 30% in the cost of production even with loading a conveyor by 50–70% [3, 4]. With an increase in the length of the transport route and a decrease in the load factor of a many-sections conveyor, the cost of transportation in the unit cost of products increases nonlinearly. A transport conveyor is a dynamically distributed system in which the

[1] SmartIndustry 2025: 2nd International Conference on Smart Automation & Robotics for Future Industry, April 03-05, 2025, Lviv, Ukraine
* Corresponding author.
† These authors contributed equally.
pihnastyi@gmail.com (O. Pihnastyi); viktoriia.usyk@nure.ua (V. Usik); Heorhii.Kozhevnikov@khpi.edu.ua (G. Kozhevnikov)
0000-0002-5424-9843 (O. Pihnastyi); 0000-0002-3515-4849 (V. Usik); 0000-0002-6586-6767 (G. Kozhevnikov)

connection of input and output parameters takes into account the transport delay. When designing systems for controlling the transportation of a material for a conveyor consisting of several sections, models are used based on the equations of system dynamics [5], the aggregated equation of state [6], the Lagrange equations, the finite element method [7-9]. With an increase in the number of sections to several dozen, these methods lose their relevance due to a significant increase in the complexity of the computational algorithm. In this case, the analytical PiKh-model of the transport conveyor can be applied to elaborate control systems [10,11]. In recent years, quite a lot of research has appeared on the use of MLP for modeling various processes in conveyor TS [12-14].

A common phenomenon in the modern mining industry is the use of multi-section conveyor systems [15-17]. The use of MLP for calculating the flow parameters of a conveyor-type TS is of scientific and practical interest even with several tens of conveyor sections [1]. The tendency to further increase the number of sections in modern conveyor systems makes the use of MLP in models of multi-section conveyor-type systems more and more actual. Unlike other studies the main attention in this research will be paid to the study of the influence of the network architecture on the accuracy of predicting the flow parameters of the transport system.

3. Problem statement

To train MLP, let us use the error back propagation algorithm [18], based on the correction of synaptic weights in accordance with the method of minimizing the mean square error. With forward signal propagation, the synaptic weights that determine the connection between the nodes of the layers of MLP are fixed. Correction of synaptic weights occurs when the signal propagates backward. An activation function is specified for each neuron. When constructing MLP, two types of activation functions were used: the logistic activation function

$$f(x) = \frac{a}{1+\exp(-bx)} \tag{1}$$

linear (ReLU) activation function

$$f(x) = bx\,, \begin{cases} f(x) = a, & if \quad x \geq a/b, \\ f(x) = bx, & if \quad a/b > x > 0, \\ f(x) = 0, & if \quad x \leq 0. \end{cases} \tag{2}$$

Within one layer, for all nodes, the same activation function is set with coefficients fixed for each node of the layer ***a***,***b***. For different layers, the form of the activation function and the coefficients that determine it may differ. For an identical function $f(x)=bx,\ x \in]-\infty;+\infty[$ (without restrictions on the range of values of the function), a multi-layer feed-forward network with a linear activation function can be reduced to a network without hidden layers. network, in this case, transforms into a linear regression model

$$y_v = w_{v0} + w_{v1}x_1 + \ldots + w_{vm}x_m + \ldots + w_{vM}x_M\,, \tag{3}$$

in which the values of the output parameters of the TS y_v are determined through the value of the input parameters x_m. TS model based on a multilayer MLP with a logistic activation function (1) has a distributed form of nonlinearity and high connectivity of network nodes, which significantly complicates the learning process, and as a consequence, the use of such models to describe TS. In this regard, to substantiate and qualitatively analyze the results of a model with a logistic activation function at the nodes of MLP, a model of a multilayer MLP with a linear activation function is considered (2). When conducting a comparative analysis of the results for MLP with different architectures, it should be borne in mind that the used back-propagation algorithm ensures convergence to one of the minima of the function that determines the value of the mean square error. This requires a fairly large number of experiments with different parameters of the activation function and the value of the learning rate. The function of the mean square error of the model (MSE) was used as a criterion for the quality of training when predicting the output flow parameters of TS y_v

$$MSE = \frac{1}{N} \sum_{n=1}^{N} \sum_{v=1}^{V} \left(y_{vtn} - y_{vn} \right)^2 , \tag{4}$$

where y_{vn}, y_{vtn} is the predicted and test value of the output parameter y_v, $v = 1..V$ for the *n*-th row of input parameters x_{mn}, $m = 1..M$ with the total number of elements N in the training set (cardinality of the set N). During training, the weights w_{vm} are adjusted within the training epoch after each training example. During one learning epoch, the weighting coefficients are adjusted N times. MSE (4) is calculated after each epoch. The correction Δw_{vm} for the weight w_{vm} with the output neuron v and the neuron m from the hidden layer (the previous layer adjacent to it) is determined by the delta rule

$$\Delta w_{vm} = -\alpha_{vm} \frac{\partial E_v}{\partial w_{vm}} . \tag{5}$$

where the quality criterion is selected as the sum of squares of errors for each of the nodes of the output layer

$$E_v = \frac{1}{2} \sum_{v} e_{v^2} . \tag{6}$$

Each output neuron is characterized by an error e_v

$$e_v = y_{vt} - y_v , \tag{7}$$

where y_{vn} is the predicted value of the output parameter, y_{vtn} is the value of the output parameter, which is used to train MLP. The distributed error for the hidden layer e_v is determined by the magnitude of the errors in the output layer. In expression (7), the index n is omitted to simplify the notation, but it is assumed that the formula is given for the n– sample of the training data set [2] used to train MLP. To train MLP, a dataset was generated based on the use of the PiKh analytical model to study the flow parameters of MLP. The methodology for generating a data set presented in [2]. The predicted value y_{vn} for $v-$ th node of the output layer is expressed in terms of the values of the nodes x_{mn} of the hidden layer (previous layer)

$$y_v = f\left(\sum_m w_{vm} x_m\right), \tag{8}$$

from where

$$\frac{\partial E_v}{\partial w_{vm}} = \frac{1}{2}\frac{\partial E_v}{\partial e_v}\frac{\partial e_v}{\partial w_{vm}} = \frac{\partial E_v}{\partial e_v}\frac{\partial (y_{vt} - y_v)}{\partial w_{vm}} = \frac{\partial E_v}{\partial e_v}\frac{\partial (y_{vt} - y_v)}{\partial y_v}\frac{\partial y_v}{\partial w_{vm}}. \tag{9}$$

Taking into account that

$$\frac{1}{2}\frac{\partial E_v}{\partial e_v} = e_v, \quad \frac{\partial (y_{vt} - y_v)}{\partial y_v} = -1, \tag{10}$$

$$\frac{\partial y_v}{\partial w_{vm}} = \frac{\partial f\left(\sum_m w_{vm} x_m\right)}{\partial w_{vm}} = \frac{\partial f\left(\sum_m w_{vm} x_m\right)}{\partial \sum_m w_{vm} x_m}\frac{\partial \sum_m w_{vm} x_m}{\partial w_{vm}} = f'\left(\sum_m w_{vm} x_m\right) x_m, \tag{11}$$

the expression for adjusting the weight w_{vm} with the output neuron v and the neuron m from the hidden layer can be represented as

$$\Delta w_{vm} = -\alpha_{vm} e_v (-1) f'\left(\sum_m w_{vm} x_m\right) x_m. \tag{12}$$

For logistic activation function (1)

$$\frac{\partial f(\beta)}{\partial \beta} = \frac{ab\exp(-b\beta)}{(1+\exp(-b\beta))^2} = bf(\beta)\frac{\exp(-b\beta)}{1+\exp(-b\beta)} = bf(\beta)\left(1 - \frac{f(\beta)}{a}\right), \tag{13}$$

considering relation (8), for the nodes of the layer with the logistic activation function, let's write

$$\Delta w_{vm} = \alpha_{vm} e_v b y_v \left(1 - \frac{y_v}{a}\right) x_m. \tag{14}$$

For linear activation function (2)

$$\frac{\partial f(\beta)}{\partial \beta} = \begin{cases} 0, & if \quad f(\beta) = a, \\ b, & if \quad a > f(\beta) > 0, \\ 0, & if \quad f(\beta) = 0, \end{cases} \tag{15}$$

and, correspondingly

$$\Delta w_{vm} = \begin{cases} 0, & if \quad y_v = a, \\ \alpha_{vm} e_v b x_m, & if \quad a > y_v > 0, \\ 0, & if \quad y_v = 0. \end{cases} \tag{16}$$

For the logistic activation function, the coefficients change with each return pass. For a linear activation function, the change in the coefficients occurs only for nodes for which the value y_v is in the range $a > y_v > 0$. Expressions (14), (16) determine the value of the adjustment of the coefficients connecting the node x_m of the hidden layer and the node y_v of the output layer.

To determine the value of adjustment Δw_{km} between the node z_k of the hidden layer, which is closer to the nodes of the output layer y_v and the node x_m of the hidden layer, which is farther to the nodes y_v of the output layer, we will use expressions (14) and (16), replacing for definiteness y_v by z_k, and e_v by ξ_k.

Then Δw_{km} between the two hidden layers will be determined by the dependency

$$\Delta w_{km} = \alpha_{km} \xi_k b z_k \left(1 - \frac{z_k}{a}\right) x_m \quad (17)$$

$$\Delta w_{km} = \begin{cases} 0, & if \quad z_k = a, \\ \alpha_{km} \xi_k b x_m, & if \quad a > z_k > 0, \\ 0, & if \quad z_k = 0. \end{cases} \quad (18)$$

with an unknown error value ξ_k for the node z_k of the hidden layer. To determine the value of the error, let us use the back-propagation algorithm. The hidden layer error ξ_m can be written in the form of the superposition of the errors of the output layers

$$\xi_m = \sum_v e_v \frac{w_{vm}}{\sum_m w_{vm}} \quad (19)$$

The fraction of the error e_v for the node of the output layer, transmitted to the hidden layer, is proportional to the coefficient w_{vm}. The calculation of the error for the nodes of the next hidden layer is determined by the expression

$$\xi_k = \sum_m \xi_m \frac{w_{mk}}{\sum_k w_{mk}} \quad (20)$$

Substituting the calculated error value ξ_k for the node z_k for the hidden layer node into formulas (17), (18), we obtain the correction value Δw_{km} for the coefficient w_{km} between the two hidden layers. For a particular case, when the value $\sum_k w_{mk}$ is comparable in order of the value for different nodes x_m, expression (21) can be simplified to the form

$$\xi_k = \sum_m \xi_m w_{mk} \quad (21)$$

In the present study, when training MLP for the back-propagation algorithm, expression (20) was used. When determining the value of adjustment Δw_{km}, the learning rate α_{km} local and is determined for each weight w_{km} when calculating the local gradient. A rather small absolute intensity of change $\Delta MSE = 10^{-6} MSE$ (4) between learning epochs was taken as a criterion for the convergence of the learning algorithm. Initialization of the weights is performed using a pseudo-random number generator [19] (new Random (long seed)), [20] in the range [0.0; 1.0]. Initialization seed=1000 provides for the possibility of repeating the experiment many times. The value of the activation function coefficient a for each node of the output layer is selected from the ratio $a \geq \max(y_{vtn})$. The maximum value of the output parameters from the training set [2] is $\max(y_{vtn}) = 4.558$.

4. Method

As a starting point for the analysis, let's use the structural diagram of the conveyor of eight sections (see Figure 1). A model of TS with this structure, based on MLP with one hidden layer and a 9-3-2 topology, was studied in [1]. To form the data set required for tutoring MLP, let's use an analytical model [2]. In [2], a detailed analysis of the generated dataset for training MLP is presented. The dataset will be used in this study. The state of the m -th section at the moment of time τ is determined by parameters: $g_m(\tau), \gamma_m(\tau), \xi_m$ are the conveyor belt speed, the material flow at the entrance of the section and the transport route length of the section, respectively.

The output flow of material from a section $\theta_1(\tau, \xi_m)$ is a calculated value. The sections are equipped with accumulating bunkers for combining the input flows of material and for separating the output flows of material [21]. There is no system to control the flow of material leaving the accumulating bunkers. The material flow from the sixth accumulating bunker is distributed between the seventh and eighth sections in a ratio $\gamma_7(\tau)/\gamma_8(\tau) = 2/3$.

Input and output nodes of MLP are numbered in accordance with the designations (Figure 2):

$$x_{3m-2} = \gamma_m(\tau),\ x_{3m-1} = g_m(\tau),\ x_{3m} = \xi_m,\ \text{m=1...M}, \tag{22}$$

$$y_1 = \theta_{17}(\tau, \xi_7),\ y_2 = \theta_{18}(\tau, \xi_8),$$
$$\xi_1 = 1.0; \xi_2 = 0.5; \xi_3 = 0.7; \xi_4 = 0.8; \xi_5 = 1.5; \xi_6 = 1.0; \xi_7 = 1.5; \xi_8 = 0.6. \tag{23}$$

The nodes corresponding to the parameters $x_7 = \gamma_3(\tau)$, $x_{14} = \gamma_6(\tau)$, $x_{19} = \gamma_7(\tau)$, $x_{22} = \gamma_8(\tau)$ are excluded from the input layer of MLP. These values can be calculated through the parameters of the previous sections.

Also excluded are nodes whose values are constant when analyzing the considered transport system $x_{3m} = \xi_m$. Thus, the input layer consists of thirteen nodes, twelve of which are determined by the parameters $\gamma_1(\tau)$, $g_1(\tau)$, $\gamma_2(\tau)$, $g_2(\tau)$, $g_3(\tau)$, $\gamma_4(\tau)$, $g_4(\tau)$, $\gamma_5(\tau)$, $g_5(\tau)$, $g_6(\tau)$, $g_7(\tau)$, $g_8(\tau)$ and a node whose value is constant and equal to 1. Nodes corresponding to calculated parameters $\gamma_3(\tau), \gamma_6(\tau), \gamma_7(\tau), \gamma_8(\tau)$ are excluded.

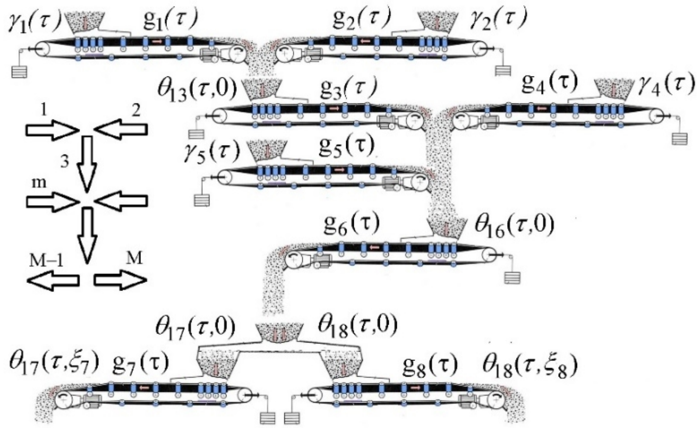

Figure 1: Structure diagram of a multi-section conveyor transport system.

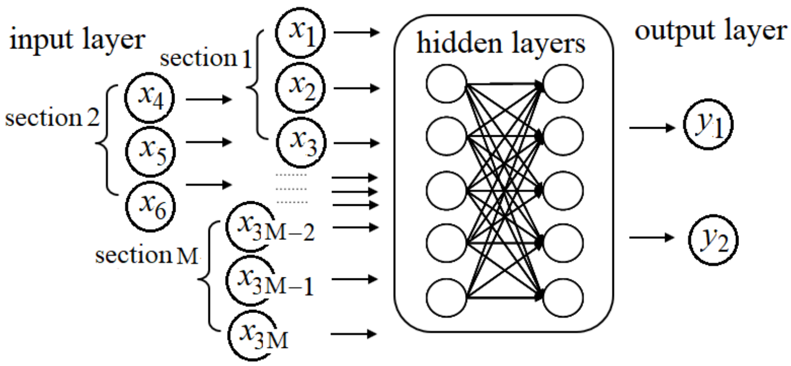

Figure 2: Neural network parameters

The values of the output node correspond to the parameters of the output material flows $\theta_{17}(\tau,\xi_7), \theta_{18}(\tau,\xi_8)$. The quantity of neurons in the hidden layers of the multilayer perceptron will be selected $N_h=15$ from the range $N_h \in [5,30]$ in accordance with the methods for determining the number of hidden neurons [22, 23]

$$N_h=\sqrt{N_i N_o} \approx 5, \quad N_h=\frac{1}{2}\frac{N}{N_i \log_2 N} \approx 30, \tag{24}$$

where the number of samples $N \approx 10^4$ in the training set for $N_i=13$ nodes of the input layer and $N_o=2$ nodes of the output layer.

In this paper, let's will consider in detail the further improvement of TS model using MLP, namely, increasing the precision of predicting the value of the parameters of the output flow of the transport conveyor by changing the architecture of MLP with the same number of neurons in hidden layers. Let's carry out a comparative analysis of models with a rectangular network architecture for hidden layers 13–$N_L \cdot L$–2 (thirteen nodes in the input layer, $N_L \cdot L$ nodes in the hidden layers and two nodes in the output layer), where N_L is the number of neurons in each hidden layer, L is the quantity of hidden layers.

The computational complexity of the algorithm at a constant transport delay is proportional to the number of sections of the conveyor. With a variable of the belt speed, the transport delay is determined by solving the equation [1]

$$\xi_m = \int_{\tau-\Delta\tau(\tau)}^{\tau} g(\alpha)\, d\alpha \tag{25}$$

If the speed $g(\alpha)$ defined as the optimal belt speed, then, as a rule, the solution to equation (24) is numerical with computational complexity $\log_2 N_{\tau ch}$, where $N_{\tau ch} = \tau_{ch} / \Delta\alpha$; τ_{ch} is characteristic time of measuring the input flow parameters of the transport system; $\Delta\alpha$ is step of numerical integration (24). A further increase in the number of sections leads to an increase in the number of equations, which makes it difficult to use the PiKh-model for designing efficient control systems due to the extreme computational complexity of the algorithm. For such a limiting number of sections of the transport conveyor, it is advisable to use models based on MLP [1] or linear regression equations [24]. This result is explained by the trade-off between the algorithm computational complexity and the forecast error of the output parameters. For a model based on MLP, the algorithm computational complexity is proportional to the value $(L+2)N_w^2$, N_w is the average number of nodes for MLP layer. For branched multi-section TS, the inequality $N_w << \sqrt{N_{\tau ch} N_K}$ is valid. This makes it attractive to use MLP models to predict the flow parameters of the transport conveyor. The number of input layer nodes depends on the number of conveyor sections. Each such section is defined by two parameters $x_{3m-2} = \gamma_m(\tau)$, $x_{3m-1} = g_m(\tau)$ (22). The above condition for the applicability of models using MLP substantiates the fact of their small use for the design of control systems for the flow parameters of the transport conveyor. The main application of models using MLP [25, 26] and regression equations [27-29] is associated, as a rule, with predicting the state of the physical characteristics of the elements of a one-section conveyor with a significant number of the regressors. TS model based on MLP with 13-5-1 architecture is used to diagnose the state of wear of a conveyor belt [25]. To control the process of extraction and transportation of material in [26], a model of the main conveyor based on MLP with a different architecture, containing 3, 16, 32, 48 and 64 nodes in a hidden layer, is analyzed. The 3-4-3 architecture is proposed for designing a control system for the start and stop mode of the conveyor section [30]. In work [31], a regression model of a transport conveyor consisting of 18 sections is analyzed, and in work [1] a model using MLP with one hidden layer for TS consisting of 8 sections.

5. Results

The accuracy of predicting the flow parameters of TS depends on the data set for training MLP. Satisfactory accuracy of the prediction of the values of the flow parameters of the transport conveyor can be ensured if there is a sufficiently large data set with the number of samples N, containing the parameters of the transport system, which are lowly correlated with each other and varying in a wide range. The number of hidden neurons (24) is directly related to the value N [22, 23]. The preparation of the set is the key moment in the learning process of MLP, it requires the provision of non-standard modes of functioning of both a separate section and TS as a whole, characterized, as a rule, by excess consumption of resources, which is unacceptable in economic conditions. For a conveyor-type transport system, consisting of several dozen sections, it is economically impossible to provide a set of possible combinations of non-standard modes, which creates an almost insurmountable obstacle to the use of MLP and regression equations when designing systems for controlling the flow parameters of a multi-section transport conveyor. A variant of the solution to this problem was proposed in [2], in which a set for training MLP is formed on the basis of an analytical PiKh-model of a conveyor without directly using an experimental data set. We can to develop a model of a multi-section conveyor based on MLP with one hidden layer (MLP architecture 9-3-2) [1] for the designing of an optimal control system for the flow parameters of TS. The introduction of a control system for the flow parameters of the TS presupposes the presence of non-contact sensors based on machine vision using digital image processing devices [32] to determine the input material flow quantity and the section belt speed. Numerical characteristics for the studied factors of the data set for training MLP are presented in Table 1 for the nodes of the input layer and in Table 2 for the nodes of the output layer of MLP 13–$N_L \cdot L$–2 (thirteen nodes in the input layer, $N_L \cdot L$ nodes in hidden layers and two nodes in output layer, Figure 2) for TS model Figure 1. The numerical characteristics of the input parameters x_k have approximately the same order of values.

Table 1
Basic numerical characteristics of the input parameters of MLP

Factor	$E[x_k]$	$\sigma_k^2 = E[(x_k - m_k)^2]$	σ_k	$\max[x_k]$	$\min[x_k]$
1.input	0,1658	0,0139	0,1179	0,3333	0
1.speed	0,4991	0,0312	0,1768	0,75	0,25
2.input	0,2083	0,0217	0,1473	0,4167	0
2.speed	0,6251	0,0488	0,221	0,9374	0,3126
3.speed	0,7508	0,0703	0,2651	1,125	0,375
4.input	0,2916	0,0426	0,2063	0,5828	0,0006
4.speed	0,8749	0,0957	0,3093	1,3124	0,4376
5.input	0,3335	0,0556	0,2357	0,6667	0
5.speed	0,9999	0,1250	0,3536	1,4993	0,5007
6.speed	1,125	0,1582	0,3978	1,6875	0,5625
7.speed	1,2495	0,1953	0,4419	1,875	0,625
8.speed	1,375	0,2364	0,4862	2,0623	0,6913

Table 2
Basic numerical characteristics of output parameters of MLP

Factor	$E[x_k]$	$\sigma_k^2 = E[(x_k - m_k)^2]$	σ_k	$\max[x_k]$	$\min[x_k]$
7.output	0,4039	0,2625	0,5123	3,8411	0,0328
8.output	0,5914	0,5068	0,7119	4,558	0,0412

If the order of the values of the numerical characteristics of the test data used for training is significantly different, then the test data set should be normalized beforehand. Normalized dataset with an average value m_v and standard deviation σ_v^2:

$$m_v = E[y_v] = 0, \ \sigma_v^2 = E[(y_v - m_v)^2] = 1 \tag{26}$$

provides the ability to compare the result with similar studies. In addition, the normalization of a set of test data is of practical importance for the process of accelerating the training of MLP.

In the considered model of the transport system, the values of the input variables are positive (input material flow, belt speed, etc.). In this case, in accordance with the formula (17), (18), the weight coefficients w_{km}, associated with the $m-$ th neuron of the hidden layer either simultaneously increase or simultaneously decrease, which leads to a change in the direction of the vector of weight coefficients to the opposite, which leads to a zigzag movement on the surface of errors and slowing down the learning process. To speed up the learning process by the back-propagation method, the training set was normalized. Numerical characteristics for the studied factors of the normalized dataset for training MLP are presented in Table 3 and Table 4.

The values of the numerical dimensionless characteristics of the input parameters presented in Table 3 are about the same range of values. We can explain it by the fact that for training MLP, an analytical method for forming a data set [2] was used, which provided values for the input parameters $\gamma_m(\tau)$ и $g_m(\tau)$, Figure 3, Figure 4:

$$\gamma_m(\tau)=\gamma_{0m}+\gamma_{1m}\sin\left(m\pi\tau-\frac{m\pi}{4}\right),\ \gamma_{0m}=\frac{3+m}{24},\ \gamma_{0m}=\gamma_{1m}, \tag{27}$$

$$g_m(\tau)=g_{0m}+g_{1m}\sin\left(m\pi\tau+\frac{m\pi}{3}\right),\ g_{0m}=\frac{3+m}{8},\ g_{1m}=\frac{g_{0m}}{2}. \tag{28}$$

A series of distributions of values $\theta_{11}(\tau,1.0)$ is shown in Figure 5. The values of the input parameters $\gamma_m(\tau)$ and $g_m(\tau)$ have a qualitatively similar series of distribution of values due to relations (27), (28). Input parameters $\gamma_m(\tau)$ and $g_m(\tau)$ (22) using the methodology [2] form the output material flow $\theta_{17}(\tau,1.5)$, $\theta_{18}(\tau,0.6)$ (23), are shown in Figure 6.

Table 3

Basic numerical normalized characteristics of the input parameters of MLP

Factor	$E[x_k]$	σ_k	$\max[x_k]$	$\min[x_k]$
1.input	0	1	1,4213	-1,4067
1.speed	0	1	1,4195	-1,409
2.input	0	1	1,4146	-1,4136
2.speed	0	1	1,4134	-1,4142
3.speed	0	1	1,4112	-1,4175
4.input	0	1	1,4117	-1,4111
4.speed	0	1	1,4145	-1,4135
5.input	0	1	1,4135	-1,415
5.speed	0	1	1,4126	-1,4118
6.speed	0	1	1,414	-1,4141
7.speed	0	1	1,4153	-1,4131
8.speed	0	1	1,4137	-1,4063

The initial filling of the sections of TS with the material is:

$$\psi_m(\tau)=\psi_{0m}+\psi_{1m}\sin\left(m\pi\tau+\frac{m\pi}{4}\right),\ \psi_{0m}=\frac{3+m}{24},\ \psi_{0m}=\psi_{1m}. \tag{29}$$

The series of distribution of values $\theta_{17}(\tau,1.5)$, $\theta_{18}(\tau,0.6)$ is presented in Figure 7, Figure 8.

To accelerate the convergence of MLP learning process when choosing input parameters in the training set, the condition of minimum correlation between the input conditions is adopted [33].

Table 4

Basic numerical normalized characteristics of output parameters of MLP

Factor	$E[x_k]$	σ_k	$\max[x_k]$	$\min[x_k]$
7.output	0	1	6,7089	-0,7244
8.output	0	1	5,572	-0,7729

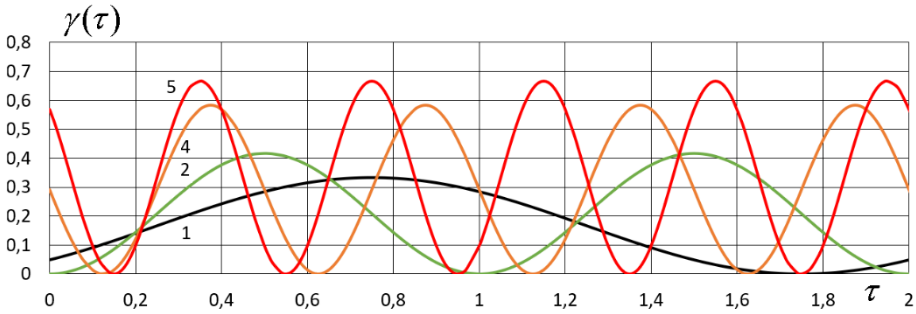

Figure 3: The flow of a material at the input of the m-th section

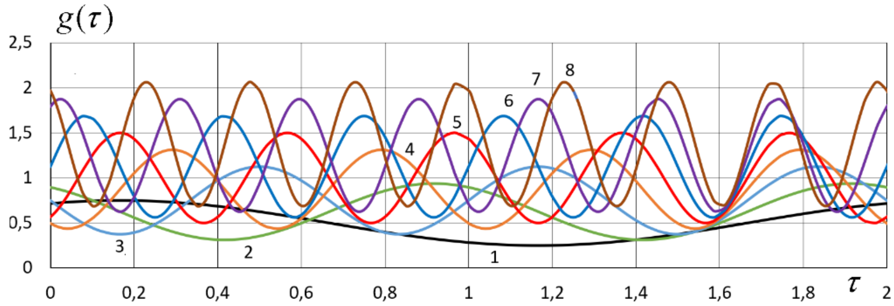

Figure 4: The belt speed m-th section

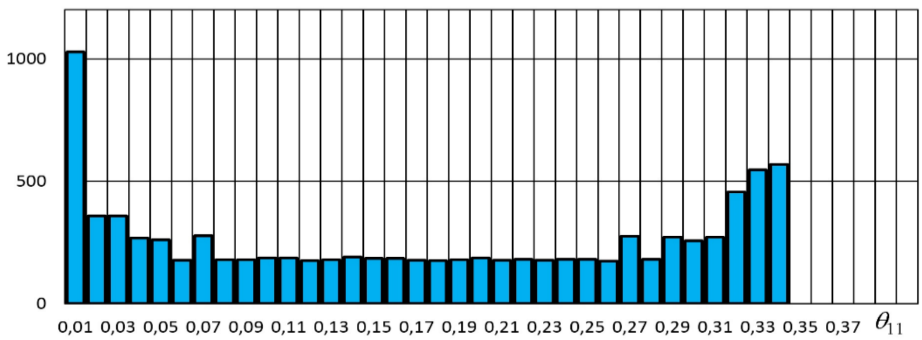

Figure 5: The series of distribution $\theta_{11}(\tau, 1.0)$

There is no correlation between the parameters x_k and x_m if the correlation coefficient between them is less in absolute value $0.2 \geq |r_{km}|$ and is considered strong if $0.8 \leq |r_{km}|$. The correlation coefficients between the input parameters of TS model are presented in Table 5.

Between the twelve parameters that make up the input layer, 66 correlation coefficients r_{km} are defined in Table 5, among which 62 coefficients satisfy the condition $0.01 \geq |r_{km}|$.

This circumstance allows us to assume that the learning process will have satisfactory convergence, and the dataset itself for training MLP has been generated quite successfully.

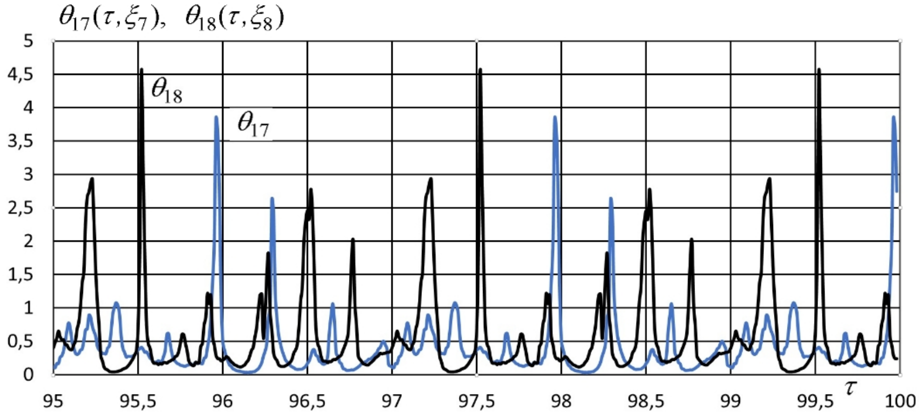

Figure 6: The material flow at the output of the 7-th and 8-th section

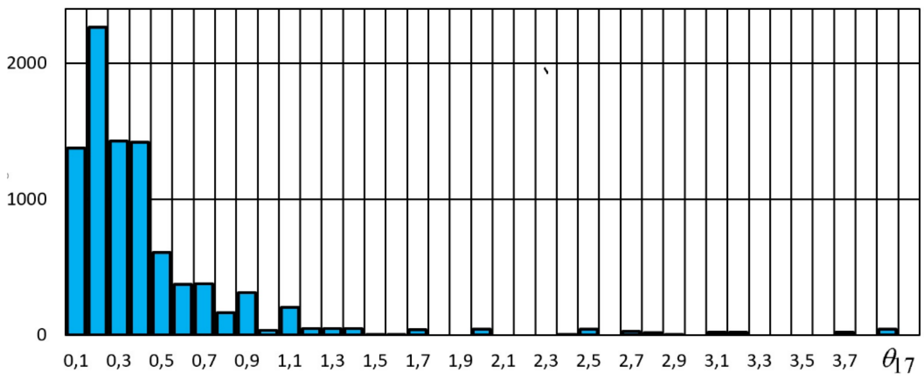

Figure 7: The series of distribution $\theta_{17}(\tau,1.5)$

To improve convergence, one of the parameters in each of the pairs [2.speed-2.input] and [5.speed-5.input] can be removed.

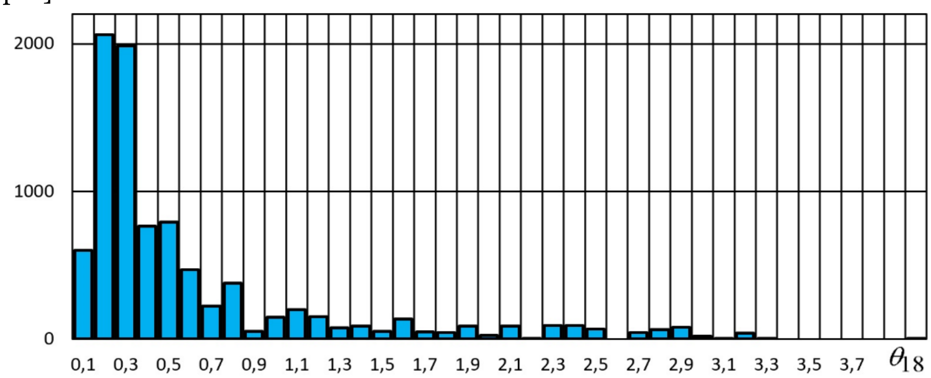

Figure 8: The series of distribution $\theta_{18}(\tau,0.6)$

Table 6 shows the results of analysing the relationship between the parameters of the input layer and the output layer parameters. The parameters that determine the values of the nodes of the input layer, the correlation coefficient of which with the parameters that determine the values of the nodes of the output layer is small, should be excluded from the set for training MLP. Input layer parameters that can become candidates for exclusion from the test dataset for training neural net are [1. speed], [4. speed].

Let's analyse the coefficient w_{km} initialization strategy. For MLP with 13-8-8-2 architecture (13 input nodes, 8 nodes each in two hidden layers and two nodes in the output layer), the optimal initialization strategy is achieved with the relation [30]:

$$\sigma_{k^2} = \sum_{m}^{M} E\left[w_{km^2}\right] = M\sigma_{w^2}, \tag{30}$$

where M is the number of neurons in the nearest layer to the input of MLP.

Table 5

The correlation coefficient between the parameters of the transport system model

Factor	1.speed	2.input	2.speed	3.speed	4.input	4.speed	5.inpt	5.speed	6.speed	7.speed	8.speed
1.input	-0,260	-0,004	0,000	0,000	-0,003	-0,002	0,000	0,000	0,002	0,000	0,000
1.speed		-0,003	0,007	0,000	0,003	0,001	0,000	0,000	-0,002	0,000	0,000
2.input			-0,867	-0,009	0,000	0,000	-0,002	0,001	0,000	0,001	0,000
2.speed				0,008	0,000	0,000	0,001	0,000	0,000	-0,001	0,000
3.speed					0,000	0,006	0,000	0,000	0,000	0,000	0,000
4.input						0,500	-0,005	0,006	0,000	-0,001	0,000
4.speed							-0,007	0,006	0,000	0,000	0,000
5.input								-0,966	-0,006	0,000	0,000
5.speed									0,007	0,000	0,000
6.speed										0,006	0,000
7.speed											0,006

For a uniform distribution law in the range [0.0; 1.0], the variance value follows $\sigma_{w^2} = 1/12$, which determines the distribution law used for the initialization of the weight coefficients. For the input nodes, the main numerical characteristics are presented in Table 1, estimate is $\sigma_{k^2} \sim (0,01 \div 0,2)$.

Table 6

Correlation coefficient between input and output the parameters of the transport system model

Factor	1.input	1.speed	2.input	2.speed	3.speed	4.input	4.speed	5.input	5.speed	6.speed	7.speed	8.speed
7.output	-0,12	0,05	-0,10	0,00	0,09	0,15	0,03	0,26	0,22	-0,25	0.23	0,11
8.output	0,10	-0,10	0,14	-0,2	0,20	-0,13	-0,07	0,19	0,21	-0,24	-0,05	0,44

Whence, for M = 10 neurons in the hidden layer (next to the input layer), can obtain an estimate of the numerical value $\sigma_{w^2} = \sigma_{k^2}/M \sim (0,001 \div 0,02)$ for the initialization of coefficients w_{km}. The estimated value of the numerical value σ_{w^2} shows, that in order to accelerate the convergence of learning MLP, the distribution law of the pseudo-random value used during initialization must be changed, to reduce the variance in the distribution law used to initialize the weight coefficients. To conduct the study, software libraries in Python were used for data processing and analysis: Pandas (version 2.0.0), Pytorch (version 2.0.0).

6. Conclusion

In this work, a comparative analysis of transport conveyor models based on multilayer MLP is carried out. The main focus is on MLP with the logistic activation function and linear (ReLU) activation function. MLP with a logistic activation function demonstrated a shorter learning time for equal MSE values, which is explained by the distributed form of nonlinearity and high connectivity. However, it should be noted that the learning process of MLP with and linear (ReLU) activation function is more stable, which allows achieving lower MSE values in the learning process.

The work touches upon many issues that are important from the point of view of constructing models of conveyor-type TS. The estimation of the method of initialization of the weight coefficients providing the connection between the nodes of MLP is carried out and the rationale for the preparation of a normalized data set for training MLP is given. The importance of the last question is emphasized by the fact that the values of the nodes of the input and output layers are positive, which leads to a simultaneous increase or decrease in the weight coefficients that ensure the connection of the nodes of the input layer with the nodes of the hidden layer, and, as a consequence, to the occurrence of oscillatory processes.

A method for increasing the accuracy of the learning process as a result of removing values from the data set is demonstrated, which characterize the transient mode of operation of the transport system.

This article does not discuss the importance of hidden neurons as detectors of factors that determine the behavior of the flow parameters of the transport system. Also, the problem of reducing the order of a multilayer MLP without losing the quality of model prediction is not considered.

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] O. Pihnastyi, V. Khodusov, Neural model of conveyor type transport system, Proceedings of The Third International Workshop on Computer Modeling and Intelligent Systems, pp.804–818 (2020). http://ceur-ws.org/Vol-2608/paper60.pdf

[2] O. Pihnastyi, Kozhevnikov, G. and Bondarenko, T.: An Analytical Method for Generating a Data Set for a Neural Model of a Conveyor Line, Proceedings of 11th International Conference on Dependable Systems, Services and Technologies, (DESSERT), pp. 202–206 ,Kyiv, Ukraine (2020). https://10.1109/DESSERT50317.2020.9125041

[3] T. Mathaba, X. Xia, A parametric energy model for energy management of long belt conveyors. Energies, 8(12) (2015) 13590-13608. https://doi.org/10.3390/en81212375

[4] Y. Shen, Y. Yu, P. Miao, H. Yu, S. Qu, Energy efficiency optimization of belt conveyors based on finite-time recurrent neural networks, Proceedings of 11th Asian Control Conference (ASCC), pp. 238-244, Gold Coast, QLD, Australia (2017). https://ieeexplore.ieee.org/abstract/document/8287404

[5] M. Yaqot, B. C. Menezes, J. D. Kelly, Real-time coordination of multiple shuttle-conveyor-belts for inventory control of multi-quality stockpiles, Computers & Chemical Engineering, 178 (2023) 108-117. https://doi.org/10.1016/j.compchemeng.2023.108388

[6] D. He, X. Liu, B. Zhong, Sustainable belt conveyor operation by active speed control, Measurement, 154 (2020) 107-113.

https://www.sciencedirect.com/science/article/abs/pii/S0263224119313259#preview-section-abstract
[7] C. A. Wheeler, Development of the rail conveyor technology, International Journal of Mining, Reclamation and Environment, 33(2) (2019) 118–132. https://doi.org/10.1080/17480930.2017.1352058
[8] D. He, Y. Pang, G. Lodewijks, X. Liu, Determination of Acceleration for Belt Conveyor Speed Control in Transient Operation. International Journal of Engineering and Technology, 8(3) (2016) 206–211. http://dx.doi.org/10.7763/IJET.2016.V8.886
[9] 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
[10] O. Pihnastyi and V. Khodusov, Model of a composite magistral conveyor line, Proceedings of the IEEE International Conference on System analysis & Intelligent computing (SAIC 2018). Kyiv, Ukraine, pp. 68-72, 2018; https://doi.org/10.1109/saic.2018.8516739
[11] 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
[12] E. Hamzeloo, M. Massinaei, N. Mehrshad, Estimation of particle size distribution on an industrial conveyor belt using image analysis and neural networks, Powder Technology, 261 (2014) 185-190. URL: https://doi.org/10.1016/j.powtec.2014.04.038
[13] E. Kalay, M. Boğoçlu, B. Bolat, Mass flow rate prediction of screw conveyor using artificial neural network method, Powder Technology, 408 (2022) 117-123. URL: https://doi.org/10.1016/j.powtec.2022.117757
[14] A. Brusaferri, M. Matteucci, S. Spinelli, A. Vitali, Learning behavioral models by recurrent neural networks with discrete latent representations with application to a flexible industrial conveyor, Computers in Industry, 122 (2020) 103-109. URL: https://doi.org/10.1016/j.compind.2020.1032633.
[15] H. Tan, K. Lim, Review of second-order optimization techniques in artificial neural networks backpropagation, Proceedings of the IOP Conference Series: Materials Science and Engineering, p.8 (2019). http://dx.doi.org/10.1088/1757-899X/495/1/012003
[16] Java™ Platform, Standard Edition 8 API Specification. Class Random. URL: https://docs.oracle.com/javase/8/docs/api/java/util/Random.html
[17] D. Knuth, Art of Computer Programming: Combinatorial Algorithms. Addison-Wesley Professional Publishing Co., Inc., USA (2022). P.784.
[18] M. Koman, Z. Laska, The constructional solution of conveyor system for reverse and bifurcation of the ore flow, CUPRUM, 3 (72) (2014) 69–82. http://www.czasopismo.cuprum.wroc.pl/journal-articles/download/113
[19] P. Bardzinski, R. Krol, L. Jurdziak, Empirical model of discretized copperore flow within the underground mine transport system. International Journal of Simulation Modelling (IJSIMM). 18(2) (2019) 279–289. http://www.ijsimm.com/Full_Papers/Fulltext2019/text18-2_279-289.pdf
[20] R. Król, W. Kawalec, L. Gładysiewicz, An effective belt conveyor for underground ore transportation systems. IOP Conference Series: Earth and Environmental Science, 95(4) (2017) 1–9. https://doi.org/10.1088%2F1755-1315%2F95%2F4%2F042047
[21] 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. 17 (2018), 597-608. https://doi.org/10.2507/IJSIMM17(4)445
[22] 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 4(9) (2020) 1320-1329. DOI: 10.18421/TEM94-02
[23] M. G. Abdolrasol, S. M. Hussain, T. S. Ustun, Artificial Neural Networks Based Optimization Techniques: A Review, Electronics, 10(21) (2021) 26-49. https://doi.org/10.3390/electronics10212689

[24] 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, Vol-2711, pp. 468–481
[25] 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. https://www.researchgate.net/publication/314952302
[26] 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, 24 (2019) 1–14. https://actamont.tuke.sk/pdf/2019/n1/1wiecek.pdf
[27] A. Kirjanów-Błażej, L. Jurdziak, R. Burduk, R. Błażej, Forecast of the remaining lifetime of steel cord conveyor belts based on regression methods in damage analysis identified by subsequent DiagBelt scans, Engineering Failure Analysis, 100 (2019) 119-126. https://doi.org/10.1016/j.engfailanal.2019.02.039
[28] 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
[29] 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
[30] O. Pihnastyi, G. Kozhevnikov, Control of a Conveyor Based on a Neural Network, in: Proceedings of the 2020 IEEE International Conference on Problems of Infocommunications Science and Technology, PICS and T (2020) 295–300. https://repository.kpi.kharkov.ua/handle/KhPI-Press/56262
[31] Y. LeCun, Efficient learning and second-order methods, Adaptive Systems Research Dept AT&T Bell Laboratories, Holmdel, NJ USA. http://www-labs.iro.umontreal.ca/~vincentp/ift3390/lectures/YannNipsTutorial.pdf
[32] Y. Gao, T. Qiao, H. Zhang, Y. Yang and X. Xiong, A Contactless Measuring Speed System of Belt Conveyor Based on Optical Flow Techniques, in: Proceedings of the IEEE Access, 7 (2019) 121646-121654. https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8812658
[33] A. Sirunyan, A. Tumasyan, W. Adam, A Deep Neural Network for Simultaneous Estimation of b Jet Energy and Resolution, Computing and Software for Big Science, 4 (2020) 143-152. https://link.springer.com/article/10.1007/s41781-020-00041-z