

One approach to control of a neural network with variable signal conductivity

A. Olshansky¹, A. Ignatenkov²

¹Railway Signalling Institute JSCo, 27 bld. 1 Nizhegorodskaya str., 109029, Moscow, Russia

²Samara State Transport University, 2 V Svobody str., 443066, Samara, Russia

Abstract

The article focuses on a new extent to synthesize a control strategy of learning of a special artificial neural network with variable signal conductivity and on features of neural networks with variable signal conductivity. The purpose of the research is to create a new control strategy based on analysis of the neural network error signal. Also the article contains an approach how to compute the control strategy. The main result is contained in the necessity to realize the control strategy on the basis of preliminary training of the neural network with specific techniques. Authors also suggest a technique of computing the crucial trajectory of the network's error decreasing. The trajectory may be computed using signal processing methods.

Keywords: neural networks; control; signal processing; preliminary training

1. Introduction

Artificial neural networks (ANN) are well known in application to transport planning, scheduling etc. Some papers are based on Hopfield's classical neural network with minimization of the energy function [1,2]. In [3] authors propose a hybrid scheme with the Hopfield's network and some heuristic methods to create timetable of CPU load. In [4] authors solve some transport problem by the multilayer perceptron with one hidden layer. All these works are not dedicated to creating the timetable with complex analysis of transport constraints. Besides all of these networks are not being trained with rigor methods. Moreover in [5,6] an approach to consider the neural network's teaching as an optimal control problem is presented. Hence in this paper authors are going to present an attempt to set the problem of rational control of the special neural network which solves rail scheduling problems.

2. The object of the study

The object of our analysis is a special neural network with variable signal conductivity described in [7]. Its topology is given in fig.1

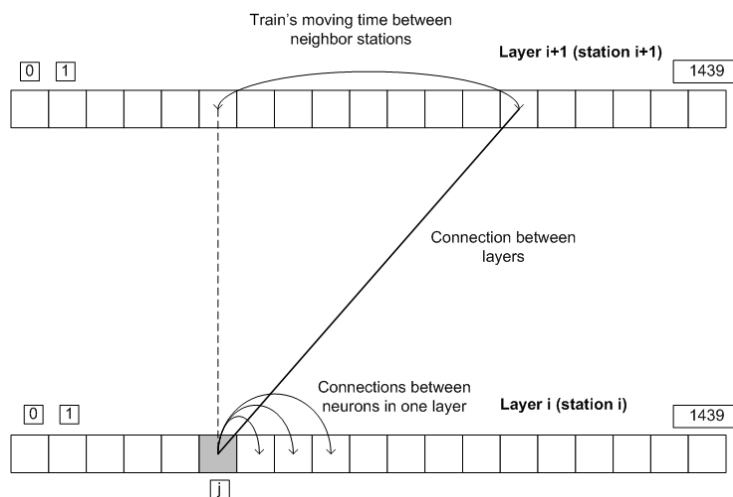


Fig.1. The architecture of the special neural network.

It is given a double-track railway section, consisting of several runs with the set times of the train moving in even and odd directions. There is an input vector $X = \{x_0, x_1, \dots, x_{1439}\}$, consisting of zeros and ones characterizing moments of train departure from the specified stations of the section. Also we know the desired vector of the train arrivals to the final station, which is the desired output value of the zero layer $X^{(0)}$ of the ANN.

The number of layers is equal to the number of railway stations. Each layer has 1440 neurons, which is equal to the number of minutes in the period of twenty-four hours.

Each neuron of the i^{th} layer is connected to each neuron of the next layer (total number is 1440 links). In addition, each neuron is associated with several neurons on the left (i.e., with neurons with a smaller number) and on the right (with neurons with a larger number).

Each matrix of weights W between two layers with numbers $i, i+1$ is a square matrix where the number of rows and columns is equal to 1440.

$$W_{i,i+1} = \begin{pmatrix} w_{0,0} & \dots & w_{0,1339} \\ \vdots & \ddots & \vdots \\ w_{1339,0} & \dots & w_{1339,1339} \end{pmatrix},$$

where w_{ij} is the weight value on the link connecting the neuron with the i^{th} layer number and the neuron with the j^{th} number of the adjacent layer.

Every neuron may be characterized by its state. Possible states of the neuron are: "active", when the input signal can be received at the input of the corresponding neuron, "sleep", when the value of the potential of the given neuron is zero, "off", when we cannot receive signals from the previous layer. The state "sleep" exists for even and odd directions.

Weights of constraints are initially specified randomly by real numbers from 0 to 0.1. Later they change during the process of the neural network's learning. The transit of the signal through the connections between the neurons of neighboring layers displays the process of traversing the train on a distance between the stations. Pay attention to minimum travel time between two stations (which is integer), all weights of links from neuron with number j from 0 to $j+t$ (where t is minimal running time) are taken equal to minus infinity. These weights never change.

Calculation of the ANN output is performed by sequential calculation of the potentials of active neurons in adjacent layers along the signal path using the sigmoid activation function.

Link weights define the level of competition between neurons for the right to receive a signal (train) in the next layer (station). Thus, the transmitted signal from one layer of the neural network to the other can't violate the rule of minimum travel time.

The direct calculation of the network is performed as follows.

1. The vector $X = \{x_0, x_1, \dots, x_{1439}\}$, which is a sequence of zeros and ones, is being fed to the first layer. The ones mean that the corresponding minute is associated with the passage of the train. X^l means a vector of neurons' values from a layer with the number l .

2. For all links issuing from the layers with numbers $n, n-1, \dots, 1$ the following holds:

$\forall k \in (0, 1, \dots, 1439)$ if $X_k^{(l)} > 0$, then $\exists j \in (0, 1, \dots, 1439): w_{k,j} = \max\{w_{km}\}, X_j^{l-1} = 0$, then we establish $X_j^{l-1} := f(X_k^{(l)}, w_{k,j})$,
where

w_{km} is the weight at the connection between the neurons of the layers k and m ,
 k is the number of the neuron of the current layer,
 j is the number of the layer adjacent to the current.

The meaning of the condition described above is that: for each neuron with the number k of the current layer with a positive output value in the next layer, we search for such a neuron that the value of their connection weight is the maximum of all the neuron bonds with the number k with the next layer.

3. The activation function is $f(x, w) = \begin{cases} 0, & \text{if } x = 0, \\ \frac{1}{1+e^{-x*w}}, & \text{if } x > 0 \end{cases}$ (1)

where

$f(x, w)$ is the value of the activation function,
 w is the weight value of the neuron with the maximum-weighted link, winning in the competition of neurons,
 x is the input to the neuron-winner.

4. With the output of the network Y we take the vector of values of the last layer $X^{(0)}$.

Moreover Y_d is the desired output of the network. It also consists of a sequence of zeros and ones whose meaning is identical to the sense of the input vector X .

We introduce the concept of "train number", which we denote by r . We say that the train passes through the station with the number l_1 per minute k_1 and the station with the number l_2 per minute k_2 , if the equality is fulfilled: $r(X_{k_1}^{l_1}) = r(X_{k_2}^{l_2})$
 $\forall r \forall l \exists! k: r(X_k^l) = r$, i.e. all trains pass through each station at exactly one point-minute. By train numbers trains are tracked for moving through all stations. The train number determines the category of the train and current train classification for scheduling.

The neural network is being trained according to the following algorithm.

1. Calculation of the ANN error.

A network error is calculated using the formula:

$$E = \frac{\sum_b (k_d - k_Y)^2 + \sum_b (\tau * k'_d)^2}{b},$$
 (2)

where

b is the number of trains that are required to be plotted in the schedule.

$r(Y_{k_Y}) = r(Y_{k_Y})$ is the number of the train for all trains that have reached the last layer,

k'_d is the index numbers of the elements of the vector Y_d , for which the network signal did not reach the last layer,

Y_{k_d}, Y_{k_Y} are the values of the elements of the target vector Y_d and the actual arrival vector Y_k respectively.

τ is the penalty coefficient for the train that has not reached the last station.

This type of error was introduced in order to emphasize the physical meaning of the network requirements (all trains must reach the layer with number 0).

2. Within the given number of learning epochs, while $E > \Delta$, where Δ is the forward accuracy of the laying of trains, perform the following steps:

For all the layers with numbers $l \in (0, 1, \dots, n-1)$, for all neurons of the l^{th} layer with the number $j \in (0, 1, \dots, 1439)$ it is set: if $X_j^{(l)} > 0$, then for the train identifier $r(X_j^{(l)}) \exists ! i: r(X_j^{(l)}) = r(X_i^{(l+1)})$. The meaning of this expression is searching in the layer $(l+1)$ of the neuron with the number i , from which the train with the identifier $r(X_j^{(l)})$ has come to the neuron j of the layer l .

We calculate the new weights of the l^{th} row of the matrix of $W_{l+1,l}$ by recalculating values of the weights according to the follows:

- reducing the weights that are situated “on the left” from the maximum weight position according to the formula:

$$\forall m \in (0, 1, \dots, i-1): w_{i,m} = w_{i,m} - \eta \cdot x_i^l \cdot f'(x_i^{l+1})$$

- reducing the maximum weight by the formula:

$$w_{i,j} = w_{i,j} - \eta \cdot x_i^l \cdot f'(x_i^{l+1})$$

- increasing the weights that are situated “to the right” from the position of maximum weight by the formula:

$$\forall m \in (i+1, i+2, \dots, i+s): w_{i,m} = w_{i,m} + G \cdot \eta \cdot x_i^l \cdot f'(x_i^{l+1}) \cdot |w_{i,j} - w_{i,m}|$$

In the equations the following notations are accepted:

f' is the derivative of the activation function (1),

$s = \sqrt{E}$ is an indicator characterizing the width of a segment within which there is a positive correction of the balance,

G is the coefficient introduced for accelerated growth of weights “on the right” from the position of the maximum-weighted link,

η is a speed of network training,

x_i^l is the output of the neuron i of the layer l ,

m is the position of the neuron in the layer relative to the value of the neuron with the maximum weight,

$w_{i,j}$ is the value of the maximum weight,

$w_{i,m}$ is the weight value for the neuron at position number m ,

$f'(x_i^{l+1})$ is the value of the derivative of the activation function in the subsequent layer $(l+1)$ for the neuron numbered i ,

x_i^l is the output of the neuron i of the layer l .

The physical meaning of learning is this: when the weights of connections “to the left” from the connection with the maximum weight are being decreased, the chance of the signal of the selected neuron to pass through the reduced connection became smaller at the next attempt (epoch). “On the right” of this connection, on the contrary, the values of the weights are being increased.

After each epoch (after a new calculation of the network values), the learning rate η increases by a value equal to $\frac{\eta}{e}$, i.e.

$$\eta(e) = \eta_{e-1} + \frac{\eta_e}{e}$$

where e is the number of epochs.

After selecting all the signal trajectories and printing them on the paper we'll get a variant of a train schedule created by the artificial neural network during its training process.

During its training process every ANN has its own error signal. The error signal is an important indicator of ANN's behavior. The further investigations are devoted to using the ANN error signal to improve its training process. Typically the dynamics of the error function (error signal) looks like in fig.2

3. Methods

In particular case when the error function could be described as sum of sinusoidal-based harmonics with different frequencies and amplitudes we may use the results obtained in [8]. In general case we are not sure in this signal error representation.

To analyze the behavior of the error function we plot its autocorrelation function (fig.3).

It gave us an assumption that it is possible to decompose the signal. The goal of this decomposition is to filter the main components of the error function. After filtering we should try to implicate the decomposition for a rational control scheme to train the network.

According to the useful practice in stochastic market signal processing we have successfully implicated LOESS techniques [9] to decompose the signal of the neural network error function. The analyzed signal, as we discovered, consists of three perceptible components: a trend part, a periodic signal and the irregular components.

The trend curve of the neural network error function provides guides for synthesis of the rational control.

4. Discussions about ways of control implementation

During the research authors have planned, realized and analyzed several series of computational experiments with variable key parameters of the neural network functioning including initial trainspeed, desired mean of the error, number of trains to

scheduling etc. It is set that we have stable and iterative character of error function view (fig.2), trend function view (fig.4). So we conclude that we should find rational control of the neural network based on this curves (fig.4).

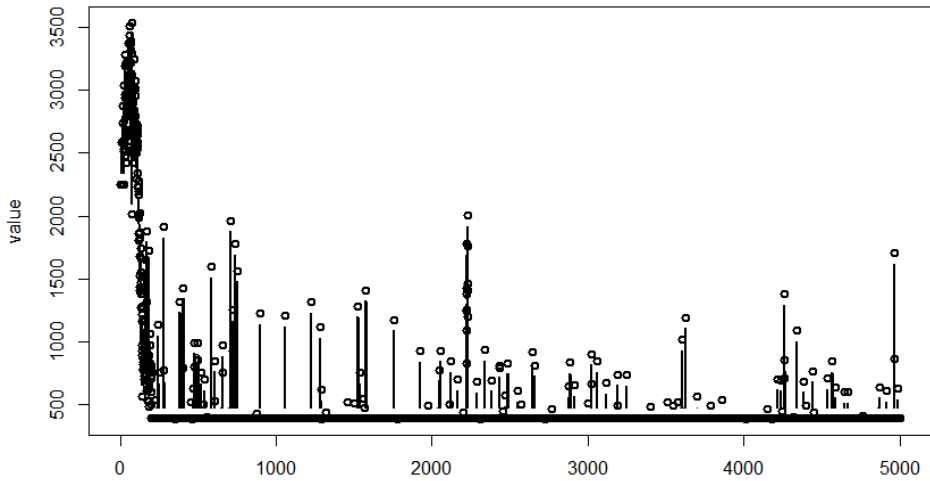


Fig.2. One example of error function.

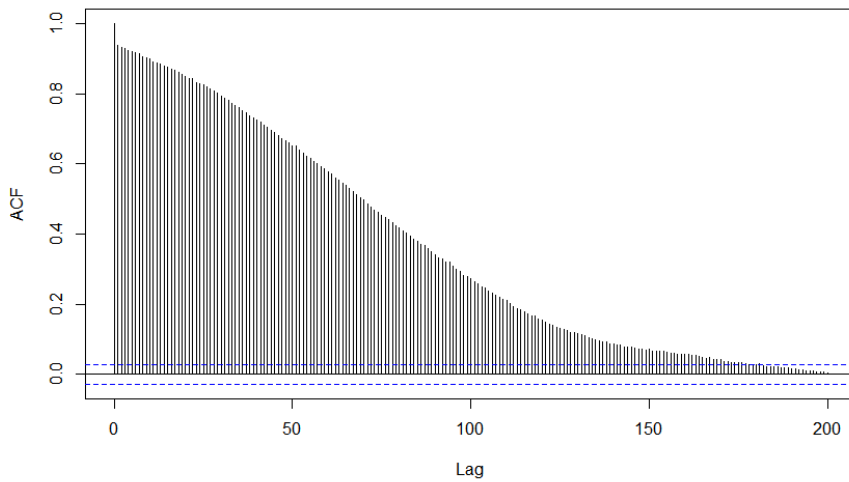


Fig.3. ACF of the error function.

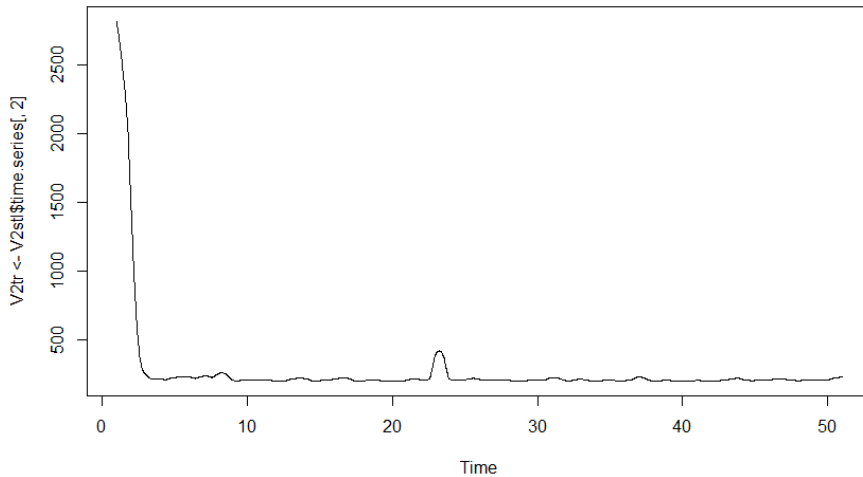


Fig.4. An example of STL-decomposition of the error function (trend).

In consideration of the enormous quantity of links between neurons it is impossible to solve the control problem in terms of dynamics of every neuron link (and the system of differential equations). So it is potentially useful to apply some special techniques to simplify and generalize control influence like:

- Pre-amplification of the bundle of links in the concrete areas of the neural network;
- Pre-dimintion of the bundle of links which are rather useless to desired trajectory of signal distribution (e.g. the links which may create too earlier or too later arrivals etc.);
- Simultaneously pre-amplification and pre-dimintion of the links;
- Mutation of links' weights in bounded area;

- Swap of weights between the links of selected pair of neurons.

Another way to invent the control strategy is implementation of inverse general neural network control.

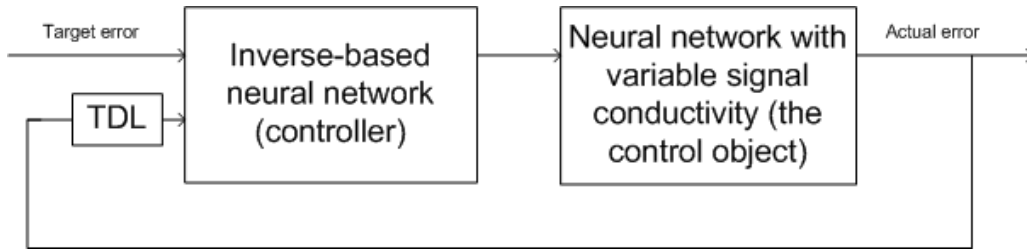


Fig.5. A principal scheme of inverse-based neural network control.

The main idea of the scheme given in fig.5 is as follows. We realize a control strategy for the neural network with variable signal conductivity (which is a controllable object) using another neural network (which is a controller). We have a target error curve (fig.4) and we know all statements of every weight per every moment so we may represent every weights change like a control step act. In epoch $(k-1)$ we know all the picture of weights changes and hence the total of used control acts. Also we know the target level of the error function in epoch number k and the actual level of the error function in epoch k . So we may describe this situation as:

$$U(k - 1) = f(E_k, E_k^t) \tag{3}$$

where:

$k - 1$ is a number of previous epoch,

k is a number of current epoch,

E_k is an observable meaning of the error function

E_k^t is a target meaning of the error function given by STL-decomposition

$U(k - 1)$ is the set of used control acts,

$f()$ is a reaction of the inverse-based neural network.

An equation (3) describes the inverse-based neural network training mode.

In the work circuit given in fig.5 we feed into the inversed neural network's input the target error in $(k+1)$ epoch and actual error in current epoch. The output of the inversed neural network should be interpreted as the control signal applicable to the neural network with variable signal conductivity. The last described scheme will work properly if the inversed-based neural network is trained relevant to behavior of the control object. Let us consider some issues of the implementation of the scheme considered in fig.5.

In [8] there was an attempt to synthesize global control strategy for artificial neural network with variable signal conductivity solving Bellman optimal control feedback task. The main conclusion of investigation [8] is that a multilayer neural network with variable signal conductivity is an output-controllable system, but not a fully-controllable system. Authors got a principal control curve, but it is rather difficult to implement the founded control function. E.g., if the neural network has only 1440 neurons in a layer and every neuron has only 100 active links with non-zero weights values for only 10 layers (is equal to 10 stations), we get about $k \times 10^6$ active weights. Hence we need to embed $k \times 10^6$ control regulators to realize the founded in [8] the only control curve. It is potentially inconvenient and it is only theoretical result because we need to research the form of functional dependence between the error level $E(t, u^*)$ and all set of $k \times 10^6$ weights and regulators (u^* in this case means the founded optimal control curve).

To avoid this authors offer training a supervising neural network using a special database to store any step of the considered neural network with its all weights and neural statuses. Authors suggest a supervisor neural network which feeds desired error level and observed error level and returns one parameter of concrete weight value in the multilayer neural network with variable signal conductivity. It leads to creation of the ensemble of multilayer perceptrons which will give us every weight parameter of considerable neural network. The structure of the database is given below.

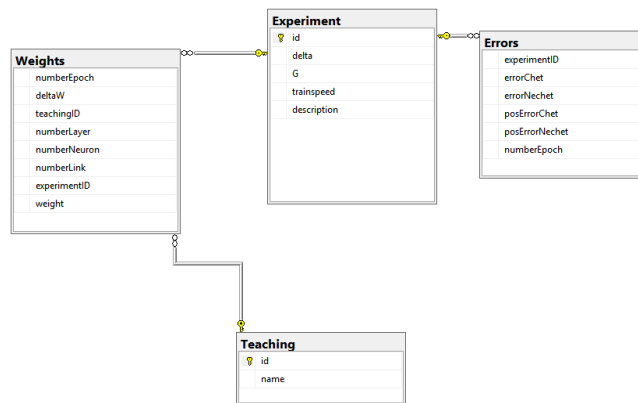


Fig.6. A structure of the database for storing the neural network statuses.

All the fields of the database are described in the table 1.

Table 1. The database fields' description.

Name of the database table	Name of the field	Data type	Description
Experiment	Id	int	Experiment identification number
	delta	int	Desired accuracy of the network
	G	int	Coefficient of the weight growth/decrease non-uniform rate of the links weights
	trainspeed	decimal(18, 5)	Initial train speed
	description	nchar(100)	comments
Errors	experimentID	int	Experiment identification number
	errorChet	decimal(18, 5)	Level of the even signals (for even trains schedule) error
	errorNechet	decimal(18, 5)	Level of the odd signals (for odd trains schedule) error
	posErrorChet	decimal(18, 5)	Positive even signal level error
	posErrorNechet	decimal(18, 5)	Positive odd signal level error
	numberEpoch	int	Number of epochs
Weights	numberEpoch	int	Number of epochs
	deltaW	float	Difference of the concrete weight level
	teachingID	int	Teaching type identifier
	numberLayer	int	Number of layer
	numberNeuron	int	Number of neuron
	numberLink	int	Number of link
	experimentID	int	Experiment identification number
Teaching	weight	float	Value of the concrete weight level
	Id	int	Teaching type identifier
	name	nchar(20)	Teaching type name

A principal control circuit is given in fig.7.

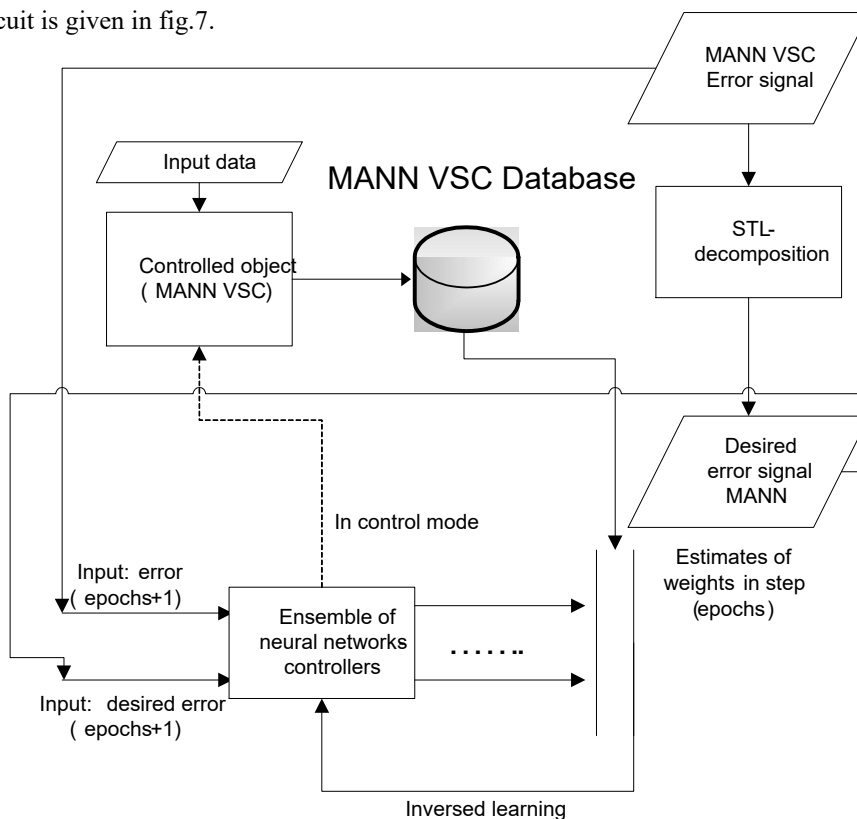


Fig.7. A structure of the control circuit and the database.

On fig.7: MANN VSC is a multilayer artificial neural network with variable signal conductivity, MANN VSC Database is a special database described in table 1, “Ensemble of neural networks controllers” is a supervisor neural network.

The main idea of fig.7 is the follows. Entrance of the controllable neural network feeds the moments of train departures and its desired arrival moments. The multilayer neural network with variable signal conductivity is functioning according to its algorithms and rules and is returning a set of error curves. These curves are being decomposed by STL to create a set of desired error signals. During this moment all statuses of the controllable neural network are being saved in the database for further storage. Values of desired error signals and real error signals are entering the entrance of the neural network supervisor ensemble, statuses of the controllable neural network are entering the output layers of the ensemble. We get inversed training of the ensemble because the order and the training procedure are implemented as in fig.5.

In current mode all trained weights for each epoch would be given to the MANN VSC entrance.

So in the present paper authors describe a new prospective approach to train the neural networks for transport scheduling.

5. Conclusions

1. To improve existing training methods of the neural networks with variable conductivity of signals it is possible to use some rigor mathematical disciplines. The theoretical basis for it can be synthesized of the settlements from Theory of Optimal Control and Theory of Signal Processing. It allows us to construct various transport timetables in more effective way.
2. Each neural network despite the kind of solved problems should be detected and registered. Its error signal should be processed and filtered to distinguish main component from signal series.
3. According to the trend component of the signal the system of weights and links of the neural network must be modified using one of techniques described above.
4. It worth sharing a new control scheme working with an inverse-based neural network control. The specific feature of the considered task is the neural network as a controllable object. This representation is innovative because in classical case we have a technical or a chemical systems described by its evolutionary equations as the controlled system and a neural network as a controller.

Acknowledgements

Our gratitude to Professor Ivanov B.G. and Kopeykin S.V. (Samara State Transport University) and Professor Prokhorov S.A. (Samara State Aerospace University) for constructive critical feedback and very beneficial advice.

References

- [1] Hopfield JJ, Tank DW. Neural Computation of Decisions in Optimization Problems. *Biological cybernetics* 1985; 52; 3: 141–152.
- [2] Chen RM, Huang YM. Competitive neural network to solve scheduling problems. *Neurocomputing* 2001; 37(1): 177–196.
- [3] Kostenko VA, Vinokurov AV. Local-optimal scheduling algorithms based on the use of Hopfield networks. *Programmirovaniye* 2003; 4: 27–40. (in Russian)
- [4] Martinelli DR, Teng H. Optimization of railway operations using neural networks. *Transportation Research Part C: Emerging Technologies* 1996; 4(1): 33–49.
- [5] Fahotimi O, Dembo A, Kailath T. Neural network weight matrix synthesis using optimal control techniques. USA, Stanford University. URL: <http://papers.nips.cc/paper/191-neural-network-weight-matrix-synthesis-using-optimal-control-techniques.pdf>.
- [6] Becerikli Y, Konar AF, Samad T. Intelligent optimal control with dynamic neural networks. *Neural networks* 2003; 16(2): 251–259.
- [7] Ignatenkov AV. Model of an artificial neural network for plotting the traffic schedule of trains on a two-track section. *International Scientific Conference Proceedings “Advanced Information Technologies and Scientific Computing”*. Samara Scientific Center of RAS Publishing, 2016; 619–623. (in Russian)
- [8] Ignatenkov A, Olshansky A. Extent of error control in neural networks. *Cornwell University Library*. URL: <https://arxiv.org/abs/1608.04682> (03.01.2017).
- [9] Cleveland RB, Cleveland WS, McRae JE, Terpenning I. STL: A seasonal-trend decomposition procedure based on Loess. *Journal of Official Statistics* 1990; 1(6): 3–73.