IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications
8-10 September 2003, Lviv, Ukraine

Multilayer Neural Networks Training Methodic

Vladimir Golovko [1], Nikolaj Maniakov [2], Leonid Makhnist [2]
1) Brest State Technical University, Laboratory of Artificial Intelligence,
Moskovskaj 267, 224017 Brest, Republic of Belarus, gva@bstu.by
2) Brest State Technical University, Department of High Mathematic,
Moskovskaj 267, 224017 Brest, Republic of Belarus, nvmaniakov@bstu.by

Abstract: *Is proposed three new techniques for training of multilayer neural networks. Its basic concept is based on the gradient descent method. For every methodic are showed formulas for calculation of the adaptive training steps. Matrix algorithmization for all of this techniques are very helpful in its program realization.*

Keywords: - *Multilayer Neural Networks, Gradient Descent Method, Adaptive Training Step*

1. INTRODUCTION

The most used neural networks architecture is two-layer feedforward nonlinear neural network. That is because of its universal approximation property [1,2,3]. Based on this conception were build forecasting networks for chaotic time series [4], neural classifiers, dynamical object descriptors and so on. So all proposed below techniques of finding of adaptive training steps will be discussed for this neural networks type.

Lets examine two layer heterogeneous neural network with m_0 neural elements in input layer, m_1 - in hidden layer and m_2 - in output layer. Every neuron of input layer have a synaptic connections $w^{(1)}_{i_0 i_1}$ $\left(i_0=\overline{1,m_0},\ i_1=\overline{1,m_1}\right)$ with all neural elements of hidden layer, and every neuron of hidden layer have a synaptic connections $w^{(2)}_{i_1 i_2}$ $\left(i_1=\overline{1,m_1},\ i_2=\overline{1,m_2}\right)$ with all neuron of output layer. Each neuron in hidden layer have activation function F_1, in output layer – F_2. The input of network are vectors $\overline{x^k}=\left(x_1^k,\dots,x_{m_0}^k\right)^T$, $\left(k=\overline{1,L}\right)$. Outputs of hidden layer are values $y^{(0),k}_{i_0}=x^k_{i_0}$, which forms vectors $Y^{(0),k}=\left(y^{(0),k}_1\ \ y^{(0),k}_2\ \ \dots\ \ y^{(0),k}_{m_0}\ \ -1\right)^T$.

Output value of i_1-s neural element of hidden layer for input vector $\overline{x^k}$ is defined as:

$$y^{(1),k}_{i_1}=F_1\left(S^{(1),k}_{i_1}\right)$$

where

$$S^{(1),k}_{i_1}=\sum_{i_0=1}^{m_0} w^{(1)}_{i_0 i_1} y^{(0),k}_{i_0}-T^{(1)}_{i_1},\quad i_1=\overline{1,m_1},\quad k=\overline{1,L}$$

In such way is formed a vector:

$$Y^{(1),k}=\left(y^{(1),k}_1\ \ y^{(1),k}_2\ \ \dots\ \ y^{(1),k}_{m_1}\ \ -1\right)^T.$$

Output value of i_2-s neural element of hidden layer for input vector $\overline{x^k}$ is defined as:

$$y^{(2),k}_{i_2}=F_2\left(S^{(2),k}_{i_2}\right)$$

where

$$S^{(2),k}_{i_2}=\sum_{i_1=1}^{m_1} w^{(2)}_{i_1 i_2} y^{(1),k}_{i_1}-T^{(2)}_{i_2},\quad i_2=\overline{1,m_2},\quad k=\overline{1,L}.$$

The task of network training with fixed activation function [5] consist of finding the weights $w^{(1)}_{i_0 i_1}$ $\left(i_0=\overline{1,m_0},\ i_1=\overline{1,m_1}\right)$, $w^{(2)}_{i_1 i_2}$ $\left(i_1=\overline{1,m_1},\ i_2=\overline{1,m_2}\right)$ and thresholds $T^{(1)}_{i_1}$ $\left(i_1=\overline{1,m_1}\right)$, $T^{(2)}_{i_2}$ $\left(i_2=\overline{1,m_2}\right)$, which minimize some net error E_s. We take a mean-square error as criterion function:

$$E_S=\frac{1}{2L}\sum_{k=1}^{L}\sum_{i_2=1}^{m_2}\left(y^{(2),k}_{i_2}-t^k_{i_2}\right)^2.$$

Matrix

$$W^{(i)}=\begin{pmatrix} w^{(i)}_{11} & w^{(i)}_{21} & \dots & w^{(i)}_{m_{i-1}1} \\ w^{(i)}_{12} & w^{(i)}_{22} & \dots & w^{(i)}_{m_{i-1}2} \\ \dots & \dots & \dots & \dots \\ w^{(i)}_{1m_i} & w^{(i)}_{2m_i} & \dots & w^{(i)}_{m_{i-1}m_i} \end{pmatrix}_{m_i\times m_{i-1}}$$

and vectors $\overline{T^{(i)}}=\left(T^{(i)}_1,\ T^{(i)}_2,\dots,\ T^{(i)}_{m_i}\right)^T$, $i=1,2$ will be called approximate solution or solution (by least square method) of system

$$F_2\left(\sum_{i_1=1}^{m_1} w^{(2)}_{i_1 i_2}\cdot F_1\left(\sum_{i_0=1}^{m_0} w^{(1)}_{i_0 i_1} y^{(0),k}_{i_0}-T^{(1)}_{i_1}\right)-T^{(2)}_{i_2}\right)=t^k_{i_2}$$

if E_s reach it minimal value.

Based on gradient descent method the changes of weights and thresholds are executed in compliance with the formulas:

$$w^{(1)}_{j_0 j_1}(t+1)=w^{(1)}_{j_0 j_1}(t)-\alpha^{(1)}\frac{\partial E_s(t)}{\partial w^{(1)}_{j_0 j_1}},$$

$$T^{(1)}_{j_1}(t+1)=T^{(1)}_{j_1}(t)-\alpha^{(1)}\frac{\partial E_s(t)}{\partial T^{(1)}_{j_1}},$$

$$w^{(2)}_{j_1 j_2}(t+1)=w^{(2)}_{j_1 j_2}(t)-\alpha^{(2)}\frac{\partial E_s(t)}{\partial w^{(2)}_{j_1 j_2}},$$

$$T_{j_2}^{(2)}(t+1)=T_{j_2}^{(2)}(t)-\alpha^{(2)}\frac{\partial E_s(t)}{\partial T_{j_2}^{(2)}}.$$

2. MATRIX ALGORITHMIZATION OF TRAINING PROCESS

Because of very complicate formulas for weights and threshold changes (that is because of not simple derivative of mean-square net error) is proposed matrix algorithmithation. It is very helpful in program realization of net training [6].

Theorem. Modifications of synaptic connection in multilayer heterogeneous neural network are produced accordance to the formulas:

$$w_{j_{n-1}j_n}^{(n)}(t+1)=w_{j_{n-1}j_n}^{(n)}(t)-\alpha^{(n)}\cdot\frac{1}{L}\cdot\sum_{k=1}^{L}C^{(n)}\cdot M_{j_nj_{n-1}}^{(n)}\cdot Y^{(n-1),k}$$

$$T_{j_n}^{(n)}(t+1)=T_{j_n}^{(n)}(t)-\alpha^{(n)}\cdot\frac{1}{L}\cdot\sum_{k=1}^{L}C^{(n)}\cdot M_{j_n(m_{n-1}+1)}^{(n)}\cdot Y^{(n-1),k},$$

where $C^{(n)}$ calculate recurrently:

$$C^{(n)}=C^{(n+1)}\cdot W^{(n+1)}\cdot MF_n\ ,\quad C^{(N)}=\varepsilon^k\cdot MF_N$$

$$\varepsilon^k=\left(\left(y_1^{(2),k}-t_1^k\right)\ \left(y_2^{(2),k}-t_2^k\right)\ \dots\ \left(y_{m_2}^{(2),k}-t_{m_2}^k\right)\right),$$

and $$MF_n=\begin{pmatrix}F_n'\left(S_1^{(n),k}\right) & 0 & \dots & 0\\ 0 & F_n'\left(S_2^{(n),k}\right) & \dots & 0\\ \dots & \dots & \dots & \dots\\ 0 & 0 & \dots & F_n'\left(S_{m_n}^{(n),k}\right)\end{pmatrix}$$

are $m_n\times m_n$ matrixes, $M_{j_nj_{n-1}}^{(n)}$ - are $m_n\times(m_{n-1}+1)$ matrixes consisting of zero elements with only one element in position j_nj_{n-1} with value equal to one.

Synaptic connection changes begin from the last layer down to first.

Proof. Let us calculate the gradient of network error for element k of taught sequence:

$$\frac{\partial E_s^{(k)}}{\partial w_{j_{n-1}j_n}^{(n)}}=\frac{\partial\left(\sum_{i_N=1}^{m_N}\frac{1}{2}\left(y_{i_N}^{(N),k}-t_{i_N}^k\right)^2\right)}{\partial w_{j_{n-1}j_n}^{(n)}}=\sum_{i_N=1}^{m_N}\left(y_{i_N}^{(N),k}-t_{i_N}^k\right)\cdot\frac{\partial y_{i_N}^{(N),k}}{\partial w_{j_{n-1}j_n}^{(n)}}=$$

$$=\sum_{i_N=1}^{m_N}\left(y_{i_N}^{(N),k}-t_{i_N}^k\right)\cdot F_N'\left(S_{i_N}^{(N),k}\right)\cdot\frac{\partial S_{i_N}^{(N),k}}{\partial w_{j_{n-1}j_n}^{(n)}}=$$

$$=\sum_{i_N=1}^{m_N}\left(y_{i_N}^{(N),k}-t_{i_N}^k\right)\cdot F_N'\left(S_{i_N}^{(N),k}\right)\cdot\sum_{i_{N-1}=1}^{m_{N-1}}w_{i_{N-1}i_N}^{(N)}\cdot\frac{\partial y_{i_{N-1}}^{(N-1),k}}{\partial w_{j_{n-1}j_n}^{(n)}}=$$

$$=\sum_{i_N=1}^{m_N}\left(y_{i_N}^{(N),k}-t_{i_N}^k\right)\cdot F_N'\left(S_{i_N}^{(N),k}\right)\times$$

$$\times\sum_{i_{N-1}=1}^{m_{N-1}}w_{i_{N-1}i_N}^{(N)}\cdot F_{N-1}'\left(S_{i_{N-1}}^{(N-1),k}\right)\cdot\frac{\partial S_{i_{N-1}}^{(N-1),k}}{\partial w_{j_{n-1}j_n}^{(n)}}=\dots=$$

$$=\sum_{i_N=1}^{m_N}\left(y_{i_N}^{(N),k}-t_{i_N}^k\right)\cdot F_N'\left(S_{i_N}^{(N),k}\right)\times$$

$$\times\sum_{i_{N-1}=1}^{m_{N-1}}w_{i_{N-1}i_N}^{(N)}\cdot F_{N-1}'\left(S_{i_{N-1}}^{(N-1),k}\right)\cdot\dots\times$$

$$\times\sum_{i_n=1}^{m_n}w_{i_ni_{n-1}}^{(n+1)}\cdot F_n'\left(S_{i_n}^{(n),k}\right)\cdot y_{j_{n-1}}^{(n-1),k}\cdot\delta_{j_n}^{i_n}=$$

$$=\varepsilon^k\cdot MF_N\cdot W^{(N)}\cdot MF_{N-1}\cdot\ldots\cdot W^{(n+1)}\cdot MF_n\times$$

$$\times M_{j_nj_{n-1}}^{(n)}\cdot Y^{(n-1),k}=C^{(n)}\cdot M_{j_nj_{n-1}}^{(n)}\cdot Y^{(n-1),k},$$

where

$$C^{(n)}=C^{(n+1)}\cdot W^{(n+1)}\cdot MF_n\ ,\quad C^{(N)}=\varepsilon^k\cdot MF_N.$$

Because of $\frac{\partial E_s}{\partial z}=\frac{\partial\left(\frac{1}{L}\sum_{k=1}^{L}E_s^{(k)}\right)}{\partial z}=\frac{1}{L}\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial z}$, we can write formulas for synaptic weights changes

$$w_{j_{n-1}j_n}^{(n)}(t+1)=w_{j_{n-1}j_n}^{(n)}(t)-\alpha^{(n)}\frac{\partial E_s}{\partial w_{j_{n-1}j_n}^{(n)}},\qquad j_{n-1}=\overline{1,m_{n-1}},$$

$j_n=\overline{1,m_n}$ like:

$$w_{j_{n-1}j_n}^{(n)}(t+1)=w_{j_{n-1}j_n}^{(n)}(t)-\alpha^{(n)}\cdot\frac{1}{L}\cdot\sum_{k=1}^{L}C^{(n)}\cdot M_{j_nj_{n-1}}^{(n)}\cdot Y^{(n-1),k}$$

In similar way we can receive formulas for network thresholds changes.

Based on the above theorem we can use the next formulas for the two-layer feedforward neural network:

$$w_{j_1j_2}^{(2)}(t+1)=w_{j_1j_2}^{(2)}(t)-\alpha^{(2)}\cdot\frac{1}{L}\cdot G_{j_1j_2}^{(2)},$$

$$T_{j_2}^{(2)}(t+1)=T_{j_2}^{(2)}(t)-\alpha^{(2)}\cdot\frac{1}{L}\cdot G_{(m_1+1)j_2}^{(2)},$$

$$w_{j_0j_1}^{(1)}(t+1)=w_{j_0j_1}^{(1)}(t)-\alpha^{(1)}\cdot\frac{1}{L}\cdot G_{j_0j_1}^{(1)},$$

$$T_{j_1}^{(1)}(t+1)=T_{j_1}^{(1)}(t)-\alpha^{(1)}\cdot\frac{1}{L}\cdot G_{(m_0+1)j_1}^{(1)},$$

where

$$G_{j_1j_2}^{(2)}=\sum_{k=1}^{L}C^{(2),k}\cdot K_{j_1j_2}^{(2),k}\ ,\ G_{j_0j_1}^{(1)}=\sum_{k=1}^{L}C^{(1),k}\cdot K_{j_0j_1}^{(1),k},$$

and

$$C^{(2),k}=\varepsilon_2^k\cdot MF_2'\ ,\ K_{j_1j_2}^{(2),k}=M_{j_2j_1}^{(2)}\cdot Y^{(1),k},$$

$$C^{(1),k}=C^{(2),k}\cdot W^{(2)}\cdot MF_1'=C^{(2),k}\cdot MW^{(1)},$$

$$K_{j_0j_1}^{(1),k}=M_{j_1j_0}^{(1)}\cdot Y^{(0),k}.$$

3. TWO-LAYER NEURAL NETWORKS' TRAINIG METHODS

For training of two-layer neural network we can take parameters $\alpha^{(1)},\alpha^{(2)}$ like constant, or like adaptive. In case of adaptive training step let us see three different cases:

1. Layerwise training. We change synaptic connection in only one layer. And then again, after giving of all training sample on to the changed neural networks, we modify weights and threshold in residuary layer.
2. Two-parameter training. Such method consist of finding different parameters $\alpha^{(1)},\alpha^{(2)}$, which combination minimize mean-square error.
3. Generalized method of fastest descent. By using of this method we finding such adaptive step $\alpha=\alpha^{(1)}=\alpha^{(2)}$, which minimize mean-square error in the antigradient direction.

We proofed next theorems for finding a adaptive steps for every of this methodic. In its formulating we use the next notations [6]:

$$\left(S^{(2)}\right)_{l_1 l_2}^{j_1 j_2}=\sum_{k=1}^{L}\left(\left(K_{l_1 l_2}^{(2),k}\right)^T\cdot\left(\left(MF_2'\right)^2+DE^k\cdot MF_2''\right)\cdot K_{j_1 j_2}^{(2),k}\right)$$

$$\left(S^{(1)}\right)_{l_0 l_1}^{j_0 j_1}=\sum_{k=1}^{L}\left(\left(K_{l_0 l_1}^{(1),k}\right)^T\cdot\left(\left(MW^{(1)}\right)^T\cdot\left[\left(MF_2'\right)^2+DE^k\cdot MF_2''\right]\cdot MW^{(1)}+\right.\right.$$
$$\left.\left.+DE^k\cdot\left(MF_2'\cdot W^{(2)}\right)\cdot MF_1''\right)\cdot K_{j_0 j_1}^{(1),k}\right)$$

$$\left(S^{(1,2)}\right)_{l_1 l_2}^{j_0 j_1}=\sum_{k=1}^{L}\left(\left(K_{l_1 l_2}^{(2),k}\right)^T\cdot\left(\left(MF_2'\right)^2+DE^k\cdot MF_2''\right)MW^{(1)}\cdot K_{j_0 j_1}^{(1),k}\right)$$

Theorem 1. For layerwise training the adaptive step is defined accordance to

$$\alpha_m^{(2)}=\frac{\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1 j_2}^{(2)}\right)^2}{\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1 l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1 l_2}^{j_1 j_2}\cdot G_{j_1 j_2}^{(2)}},$$

$$\alpha_m^{(1)}=\frac{\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0 j_1}^{(1)}\right)^2}{\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0 l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0 l_1}^{j_0 j_1}\cdot G_{j_0 j_1}^{(1)}}.$$

Proof. After modification of second layer synaptic connections the network error is changed accordance to the formula:

$$E_s(t+1)=\frac{1}{L}\cdot\sum_{k=1}^{L}E_s^{(k)}(t+1)=\frac{1}{L}\sum_{k=1}^{L}E_s^{(k)}(t)+$$
$$+\frac{1}{L}\cdot\left(\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial w_{j_1 j_2}^{(2)}}\right)\cdot\left(w_{j_1 j_2}^{(2)}(t+1)-w_{j_1 j_2}^{(2)}(t)\right)+\right.$$
$$\left.+\sum_{j_2=1}^{m_2}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial T_{j_2}^{(2)}}\right)\cdot\left(T_{j_2}^{(2)}(t+1)-T_{j_2}^{(2)}(t)\right)\right)+$$
$$+\frac{1}{2L}\cdot\sum_{k=1}^{L}\left(\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\sum_{l_1=1}^{m_1}\sum_{l_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial w_{j_1 j_2}^{(2)}\partial w_{l_1 l_2}^{(2)}}\cdot\left(w_{j_1 j_2}^{(2)}(t+1)-w_{j_1 j_2}^{(2)}(t)\right)\times\right.$$
$$\times\left(w_{l_1 l_2}^{(2)}(t+1)-w_{l_1 l_2}^{(2)}(t)\right)+$$
$$+\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\sum_{l_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial w_{j_1 j_2}^{(2)}\partial T_{l_2}^{(2)}}\cdot\left(w_{j_1 j_2}^{(2)}(t+1)-w_{j_1 j_2}^{(2)}(t)\right)\times$$
$$\times\left(T_{l_2}^{(2)}(t+1)-T_{l_2}^{(2)}(t)\right)+$$
$$+\sum_{l_1=1}^{m_1}\sum_{l_2=1}^{m_2}\sum_{j_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial T_{j_2}^{(2)}\partial w_{l_1 l_2}^{(2)}}\cdot\left(T_{j_2}^{(2)}(t+1)-T_{j_2}^{(2)}(t)\right)\times$$
$$\times\left(w_{l_1 l_2}^{(2)}(t+1)-w_{l_1 l_2}^{(2)}(t)\right)+$$
$$+\sum_{l_2=1}^{m_2}\sum_{j_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial T_{l_2}^{(2)}\partial T_{j_2}^{(2)}}\cdot\left(T_{l_2}^{(2)}(t+1)-T_{l_2}^{(2)}(t)\right)\times$$
$$\left.\times\left(T_{j_2}^{(2)}(t+1)-T_{j_2}^{(2)}(t)\right)\right)=$$
$$=E_s(t)-\alpha^{(2)}\cdot\frac{1}{L^2}\cdot\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1 j_2}^{(2)}\right)^2+$$
$$+\left(\alpha^{(2)}\right)^2\cdot\frac{1}{2L^3}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1 l_2}^{(2)}\cdot\sum_{k=1}^{L}\left(\left(K_{l_1 l_2}^{(2),k}\right)^T\cdot\left(\left(MF_2'\right)^2+\right.\right.\right.$$
$$\left.\left.\left.+DE^k\cdot MF_2''\right)\cdot K_{j_1 j_2}^{(2),k}\right)G_{j_1 j_2}^{(2)}\right)$$

Lets find a such value of $\alpha^{(2)}$, which minimizes network error. For that purposes we must solve the next equation:

$$\frac{\partial E_s(t+1)}{\partial\alpha^{(2)}}=-\frac{1}{L^2}\cdot\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1 j_2}^{(2)}\right)^2+$$
$$+\alpha^{(2)}\cdot\frac{1}{L^3}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1 l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1 l_2}^{j_1 j_2}\cdot G_{j_1 j_2}^{(2)}\right)=0,$$

where

$$\left(S^{(2)}\right)_{l_1 l_2}^{j_1 j_2}=\sum_{k=1}^{L}\left(\left(K_{l_1 l_2}^{(2),k}\right)^T\cdot\left(\left(MF_2'\right)^2+DE^k\cdot MF_2''\right)\cdot K_{j_1 j_2}^{(2),k}\right)$$

We receive the next value of $\alpha^{(2)}$:

$$\alpha^{(2)}=L\cdot\frac{\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1 j_2}^{(2)}\right)^2}{\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1 l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1 l_2}^{j_1 j_2}\cdot G_{j_1 j_2}^{(2)}}.$$

So

$$\alpha_m^{(2)}=\alpha^{(2)}\cdot\frac{1}{L}=\frac{\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1 j_2}^{(2)}\right)^2}{\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1 l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1 l_2}^{j_1 j_2}\cdot G_{j_1 j_2}^{(2)}}.$$

After changes of second layer synaptic connections the network error is changed accordance to the formula:

$$E_s(t+1)=\frac{1}{L}\cdot\sum_{k=1}^{L}E_s^{(k)}(t+1)=\frac{1}{L}\sum_{k=1}^{L}E_s^{(k)}(t)+$$

$$+\frac{1}{L}\cdot\left(\sum_{j_0=1}^{m_0}\sum_{j_1=1}^{m_1}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial w_{j_0j_1}^{(1)}}\right)\cdot\left(w_{j_0j_1}^{(1)}(t+1)-w_{j_0j_1}^{(1)}(t)\right)+\right.$$

$$\left.+\sum_{j_1=1}^{m_1}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial T_{j_1}^{(1)}}\right)\cdot\left(T_{j_1}^{(1)}(t+1)-T_{j_1}^{(1)}(t)\right)\right)=$$

$$+\frac{1}{2L}\cdot\sum_{k=1}^{L}\left(\sum_{j_0=1}^{m_0}\sum_{j_1=1}^{m_1}\sum_{l_0=1}^{m_0}\sum_{l_1=1}^{m_1}\frac{\partial^2 E_s^{(k)}}{\partial w_{j_0j_1}^{(1)}\partial w_{l_0l_1}^{(1)}}\cdot\left(w_{j_0j_1}^{(1)}(t+1)-w_{j_0j_1}^{(1)}(t)\right)\times\right.$$

$$\times\left(w_{l_0l_1}^{(1)}(t+1)-w_{l_0l_1}^{(1)}(t)\right)+$$

$$+\sum_{j_0=1}^{m_0}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1}\frac{\partial^2 E_s^{(k)}}{\partial w_{j_0j_1}^{(1)}\partial T_{l_1}^{(1)}}\cdot\left(w_{j_0j_1}^{(1)}(t+1)-w_{j_0j_1}^{(1)}(t)\right)\times$$

$$\times\left(T_{l_1}^{(1)}(t+1)-T_{l_1}^{(1)}(t)\right)+$$

$$+\sum_{l_0=1}^{m_0}\sum_{l_1=1}^{m_1}\sum_{j_1=1}^{m_1}\frac{\partial^2 E_s^{(k)}}{\partial T_{j_1}^{(1)}\partial w_{l_0l_1}^{(1)}}\cdot\left(T_{j_1}^{(1)}(t+1)-T_{j_1}^{(1)}(t)\right)\times$$

$$\times\left(w_{l_0l_1}^{(1)}(t+1)-w_{l_0l_1}^{(1)}(t)\right)+$$

$$+\sum_{l_1=1}^{m_1}\sum_{j_1=1}^{m_1}\frac{\partial^2 E_s^{(k)}}{\partial T_{l_1}^{(1)}\partial T_{j_1}^{(1)}}\cdot\left(T_{l_1}^{(1)}(t+1)-T_{l_1}^{(1)}(t)\right)\times$$

$$\left.\times\left(T_{j_1}^{(1)}(t+1)-T_{j_1}^{(1)}(t)\right)\right)=$$

$$=E_s(t)-\alpha^{(1)}\cdot\frac{1}{L^2}\cdot\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2+$$

$$+\left(\alpha^{(1)}\right)^2\cdot\frac{1}{2L^3}\cdot\left(\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}G_{j_0j_1}^{(1)}\right),$$

where

$$\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}=\sum_{k=1}^{L}\left(\left(K_{l_0l_1}^{(1),k}\right)^T\cdot\left(\left(MW^{(1)}\right)^T\cdot\left[\left(MF_2'\right)^2+\right.\right.\right.$$

$$\left.\left.\left.+DE^k\cdot MF_2''\right]\cdot MW^{(1)}+DE^k\cdot\left(MF_2'\cdot W^{(2)}\right)\cdot MF_1''\right)\cdot K_{j_0j_1}^{(1),k}\right)$$

Lets find a such value of $\alpha^{(1)}$, which minimizes network error. For that purposes we must solve the next equation:

$$\frac{\partial E_s(t+1)}{\partial\alpha^{(1)}}=-\frac{1}{L^2}\cdot\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2+$$

$$+\alpha^{(1)}\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}G_{j_0j_1}^{(1)}\right)=0$$

We receive the next value of $\alpha^{(1)}$:

$$\alpha^{(1)}=L\cdot\frac{\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2}{\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}G_{j_0j_1}^{(1)}}.$$

So

$$\alpha_m^{(1)}=\alpha^{(1)}\cdot\frac{1}{L}=\frac{\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2}{\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}G_{j_0j_1}^{(1)}}.$$

Theorem 2. For two-parameter training the adaptive step is defined accordance to

$$\alpha_m^{(1)}=\frac{D_1C-D_2B}{AC-B^2},\ \alpha_m^{(2)}=\frac{D_2A-D_1B}{AC-B^2}.$$

where

$$A=\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}G_{j_0j_1}^{(1)},$$

$$B=\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)},$$

$$C=\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2}G_{j_1j_2}^{(2)},$$

$$D_1=\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2,\ D_2=\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1j_2}^{(2)}\right)^2.$$

Proof. After synaptic connection modification the network error is changed in the next way:

$$E_s(t+1)=\frac{1}{L}\cdot\sum_{k=1}^{L}E_s^{(k)}(t+1)=\frac{1}{L}\sum_{k=1}^{L}E_s^{(k)}(t)+$$

$$+\frac{1}{L}\cdot\left(\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial w_{j_1j_2}^{(2)}}\right)\cdot\left(w_{j_1j_2}^{(2)}(t+1)-w_{j_1j_2}^{(2)}(t)\right)+\right.$$

$$\left.+\sum_{j_2=1}^{m_2}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial T_{j_2}^{(2)}}\right)\cdot\left(T_{j_2}^{(2)}(t+1)-T_{j_2}^{(2)}(t)\right)\right)+$$

$$+\frac{1}{L}\cdot\left(\sum_{j_0=1}^{m_0}\sum_{j_1=1}^{m_1}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial w_{j_0j_1}^{(1)}}\right)\cdot\left(w_{j_0j_1}^{(1)}(t+1)-w_{j_0j_1}^{(1)}(t)\right)+\right.$$

$$\left.+\sum_{j_1=1}^{m_1}\left(\sum_{k=1}^{L}\frac{\partial E_s^{(k)}}{\partial T_{j_1}^{(1)}}\right)\cdot\left(T_{j_1}^{(1)}(t+1)-T_{j_1}^{(1)}(t)\right)\right)+$$

$$+\frac{1}{2L}\cdot\sum_{k=1}^{L}\left(\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\sum_{l_1=1}^{m_1}\sum_{l_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial w_{j_1j_2}^{(2)}\partial w_{l_1l_2}^{(2)}}\cdot\left(w_{j_1j_2}^{(2)}(t+1)-w_{j_1j_2}^{(2)}(t)\right)\times\right.$$

$$\times\left(w_{l_1l_2}^{(2)}(t+1)-w_{l_1l_2}^{(2)}(t)\right)+$$

$$+\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\sum_{l_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial w_{j_1j_2}^{(2)}\partial T_{l_2}^{(2)}}\cdot\left(w_{j_1j_2}^{(2)}(t+1)-w_{j_1j_2}^{(2)}(t)\right)\times$$

$$\times\left(T_{l_2}^{(2)}(t+1)-T_{l_2}^{(2)}(t)\right)+$$

$$+\sum_{l_1=1}^{m_1}\sum_{l_2=1}^{m_2}\sum_{j_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial T_{j_2}^{(2)}\partial w_{l_1l_2}^{(2)}}\cdot\left(T_{j_2}^{(2)}(t+1)-T_{j_2}^{(2)}(t)\right)\times$$

$$\times\left(w_{l_1l_2}^{(2)}(t+1)-w_{l_1l_2}^{(2)}(t)\right)+$$

$$+\sum_{l_2=1}^{m_2}\sum_{j_2=1}^{m_2}\frac{\partial^2 E_s^{(k)}}{\partial T_{l_2}^{(2)}\partial T_{j_2}^{(2)}}\cdot\left(T_{l_2}^{(2)}(t+1)-T_{l_2}^{(2)}(t)\right)\times$$

$$\times\left(T_{j_2}^{(2)}(t+1)-T_{j_2}^{(2)}(t)\right)\Big)+$$

$$+\frac{1}{2L}\cdot\sum_{k=1}^{L}\Big(\sum_{j_0=1}^{m_0}\sum_{j_1=1}^{m_1}\sum_{l_0=1}^{m_0}\sum_{l_1=1}^{m_1}\frac{\partial^2 E_s^{(k)}}{\partial w_{j_0 j_1}^{(1)}\partial w_{l_0 l_1}^{(1)}}\cdot\left(w_{j_0j_1}^{(1)}(t+1)-w_{j_0j_1}^{(1)}(t)\right)\times$$

$$\times\left(w_{l_0 l_1}^{(1)}(t+1)-w_{l_0l_1}^{(1)}(t)\right)+$$

$$+\sum_{j_0=1}^{m_0}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1}\frac{\partial^2E_s^{(k)}}{\partial w_{j_0j_1}^{(1)}\partial T_{l_1}^{(1)}}\cdot\left(w_{j_0j_1}^{(1)}(t+1)-w_{j_0j_1}^{(1)}(t)\right)\times$$

$$\times\left(T_{l_1}^{(1)}(t+1)-T_{l_1}^{(1)}(t)\right)+$$

$$+\sum_{l_0=1}^{m_0}\sum_{l_1=1}^{m_1}\sum_{j_1=1}^{m_1}\frac{\partial^2E_s^{(k)}}{\partial T_{j_1}^{(1)}\partial w_{l_0l_1}^{(1)}}\cdot\left(T_{j_1}^{(1)}(t+1)-T_{j_1}^{(1)}(t)\right)\times$$

$$\times\left(w_{l_0l_1}^{(1)}(t+1)-w_{l_0l_1}^{(1)}(t)\right)+$$

$$+\sum_{l_1=1}^{m_1}\sum_{j_1=1}^{m_1}\frac{\partial^2E_s^{(k)}}{\partial T_{l_1}^{(1)}\partial T_{j_1}^{(1)}}\cdot\left(T_{l_1}^{(1)}(t+1)-T_{l_1}^{(1)}(t)\right)\times$$

$$\times\left(T_{j_1}^{(1)}(t+1)-T_{j_1}^{(1)}(t)\right)\Big)+$$

$$+2\cdot\frac{1}{2L}\cdot\sum_{k=1}^{L}\Big(\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\sum_{l_0=1}^{m_0}\sum_{l_1=1}^{m_1}\frac{\partial^2E_s^{(k)}}{\partial w_{j_1j_2}^{(2)}\partial w_{l_0l_1}^{(1)}}\cdot\left(w_{j_1j_2}^{(2)}(t+1)-w_{j_1j_2}^{(2)}(t)\right)\times$$

$$\times\left(w_{l_0l_1}^{(1)}(t+1)-w_{l_0l_1}^{(1)}(t)\right)+$$

$$+\sum_{j_1=1}^{m_1}\sum_{j_2=1}^{m_2}\sum_{l_1=1}^{m_1}\frac{\partial^2E_s^{(k)}}{\partial w_{j_1j_2}^{(2)}\partial T_{l_1}^{(1)}}\cdot\left(w_{j_1j_2}^{(2)}(t+1)-w_{j_1j_2}^{(2)}(t)\right)\times$$

$$\times\left(T_{l_1}^{(1)}(t+1)-T_{l_1}^{(1)}(t)\right)+$$

$$+\sum_{l_0=1}^{m_0}\sum_{l_1=1}^{m_1}\sum_{j_2=1}^{m_2}\frac{\partial^2E_s^{(k)}}{\partial T_{j_2}^{(2)}\partial w_{l_0l_1}^{(1)}}\cdot\left(T_{j_2}^{(2)}(t+1)-T_{j_2}^{(2)}(t)\right)\times$$

$$\times\left(w_{l_0l_1}^{(1)}(t+1)-w_{l_0l_1}^{(1)}(t)\right)+$$

$$+\sum_{l_2=1}^{m_2}\sum_{j_1=1}^{m_1}\frac{\partial^2E_s^{(k)}}{\partial T_{l_2}^{(2)}\partial T_{j_1}^{(1)}}\cdot\left(T_{l_2}^{(2)}(t+1)-T_{l_2}^{(2)}(t)\right)\times$$

$$\times\left(T_{j_1}^{(1)}(t+1)-T_{j_1}^{(1)}(t)\right)\Big)=$$

$$=E_s(t)-\alpha^{(2)}\cdot\frac{1}{L^2}\cdot\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1j_2}^{(2)}\right)^2-$$

$$-\alpha^{(1)}\cdot\frac{1}{L^2}\cdot\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2+$$

$$+\left(\alpha^{(2)}\right)^2\cdot\frac{1}{2L^3}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2}\cdot G_{j_1j_2}^{(2)}\right)+$$

$$+\left(\alpha^{(1)}\right)^2\cdot\frac{1}{2L^3}\cdot\left(\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right)+$$

$$+\alpha^{(1)}\cdot\alpha^{(2)}\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right).$$

Lets find parameters $\alpha^{(1)}$ and $\alpha^{(2)}$, which minimize criterion function. For that purposes, we must find critical points by solving the system of equations, where partial derivatives are equal to zero. Such critical point will by a point of minima for our function of network error.

$$\frac{\partial E_s(t+1)}{\partial\alpha^{(1)}}=-\frac{1}{L^2}\cdot\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2+$$

$$+\alpha^{(1)}\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right)+$$

$$+\alpha^{(2)}\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right),$$

$$\frac{\partial E_s(t+1)}{\partial\alpha^{(2)}}=-\frac{1}{L^2}\cdot\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1j_2}^{(2)}\right)^2+$$

$$+\alpha^{(2)}\cdot\frac{1}{L^3}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2}\cdot G_{j_1j_2}^{(2)}\right)+$$

$$+\alpha^{(1)}\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right).$$

We must solve the next system of linear equation:

$$\alpha^{(1)}\cdot\left(\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1}G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right)+$$

$$+\alpha^{(2)}\cdot\left(\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right)=$$

$$=L\cdot\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2,$$

$$\alpha^{(1)}\cdot\left(\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right)+$$

$$+\alpha^{(2)}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2}G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2}\cdot G_{j_1j_2}^{(2)}\right)=$$

$$L\cdot\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1j_2}^{(2)}\right)^2,$$

or in our notation:

$$\begin{cases}\alpha^{(1)}\cdot A+\alpha^{(2)}\cdot B=L\cdot D_1\\ \alpha^{(1)}\cdot B+\alpha^{(2)}\cdot C=L\cdot D_2\end{cases}.$$

By using Kramer method we receive

$$\alpha^{(1)}=\frac{\begin{vmatrix}L\cdot D_1 & B\\ L\cdot D_2 & C\end{vmatrix}}{\begin{vmatrix}A & B\\ B & C\end{vmatrix}}=L\cdot\frac{D_1C-D_2B}{AC-B^2}$$

$$\alpha^{(2)}=\frac{\begin{vmatrix}A & L\cdot D_1\\ B & L\cdot D_2\end{vmatrix}}{\begin{vmatrix}A & B\\ B & C\end{vmatrix}}=L\cdot\frac{D_2A-D_1B}{AC-B^2}$$

And finally,

$$\alpha_m^{(1)}=\frac{1}{L}\cdot\alpha^{(1)}=\frac{D_1C-D_2B}{AC-B^2}$$

$$\alpha_m^{(2)} = \frac{1}{L}\cdot\alpha^{(2)} = \frac{D_2A - D_1B}{AC - B^2}.$$

Theorem 3. For generalized method of fastest descent the adaptive step is defined accordance to

$$\alpha_m = \frac{D_1 + D_2}{A + 2B + C},$$

where

$$A = \sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1} G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1} G_{j_0j_1}^{(1)},$$

$$B = \sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2} G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)},$$

$$C = \sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2} G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2} G_{j_1j_2}^{(2)},$$

$$D_1 = \sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2,\quad D_2 = \sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1j_2}^{(2)}\right)^2.$$

Proof. Extending error function in to the Taylor series as was described above, but remembering that $\alpha^{(1)} = \alpha^{(2)} = \alpha$, we receive:

$$E_s(t+1) = \frac{1}{L}\cdot\sum_{k=1}^{L} E_s^{(k)}(t+1) =$$

$$= E_s(t) - \alpha\cdot\frac{1}{L^2}\cdot\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1j_2}^{(2)}\right)^2 - \alpha\cdot\frac{1}{L^2}\cdot\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2 +$$

$$+\alpha^2\cdot\frac{1}{2L^3}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2} G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2} G_{j_1j_2}^{(2)}\right) +$$

$$+\alpha^2\cdot\frac{1}{2L^3}\cdot\left(\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1} G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1} G_{j_0j_1}^{(1)}\right) +$$

$$+\alpha^2\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2} G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right).$$

Let find such value of α, which minimize error function. For that purpose we must solve the equation:

$$\frac{\partial E_s(t+1)}{\partial\alpha} = -\frac{1}{L^2}\cdot\sum_{j_1=1}^{m_1+1}\sum_{j_2=1}^{m_2}\left(G_{j_1j_2}^{(2)}\right)^2 - \frac{1}{L^2}\cdot\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\left(G_{j_0j_1}^{(1)}\right)^2 +$$

$$+\alpha\cdot\frac{1}{L^3}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2} G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2} G_{j_1j_2}^{(2)}\right) +$$

$$+\alpha\cdot\frac{1}{L^3}\cdot\left(\sum_{j_1,l_1=1}^{m_1+1}\sum_{j_2,l_2=1}^{m_2} G_{l_1l_2}^{(2)}\cdot\left(S^{(2)}\right)_{l_1l_2}^{j_1j_2} G_{j_1j_2}^{(2)}\right) +$$

$$+\alpha\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0,l_0=1}^{m_0+1}\sum_{j_1,l_1=1}^{m_1} G_{l_0l_1}^{(1)}\cdot\left(S^{(1)}\right)_{l_0l_1}^{j_0j_1} G_{j_0j_1}^{(1)}\right) +$$

$$+2\alpha\cdot\frac{1}{L^3}\cdot\left(\sum_{j_0=1}^{m_0+1}\sum_{j_1=1}^{m_1}\sum_{l_1=1}^{m_1+1}\sum_{l_2=1}^{m_2} G_{l_1l_2}^{(2)}\cdot\left(S^{(1,2)}\right)_{l_1l_2}^{j_0j_1}\cdot G_{j_0j_1}^{(1)}\right) = 0$$

So adaptive training step we can calculate accordance to the formula:

$$\alpha = L\cdot\frac{D_1 + D_2}{A + 2B + C},$$

And finally

$$\alpha_m = \alpha\cdot\frac{1}{L} = \frac{D_1 + D_2}{A + 2B + C}.$$

4. CONCLUSION

In practical experiments we can see that two-parameter training has a advantage before another proposed methods in a time of convergence.

5. REFERENCES

[1] K. Hornik, M. Stinchcombe, H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, Vol. 2 (1989). pp. 359-366.

[2] G. Cybenco. Approximation by superposition of sigmoidal function. *Mathematics of Control, Signals and Systems*, Vol. 2 (1989). pp. 303-314.

[3] K.-I. Funahashi. On the approximate realization of continuous mapping by neural networks. *Neural Networks*, Vol. 2 (1989). pp. 183-192.

[4] V. Golovko, Y. Savitsky, N. Maniakov. Modeling Nonlinear Dynamic using Multilayer Neural Networks. *Proceedings of the Workshop Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications. (IDAACS'2001)*, Foros, Ukraine, 1-4 July 2001, pp.197-202.

[5] I. Gladkij, V. Golovko, L. Makhnist. Neural network training with use of method of fastest descent. *Vestnik BGTU*, Vol. 5 (11) (2001). pp. 56-61. (in Russian).

[6] N. Maniakov, L. Makhnist. Matrix algorithmithation of training of multilayer neural networks with use of gradient descents methods. *Vestnik BGTU*, Vol.5 (17) (2002). (in Russian).