Coordination algorithm in hierarchical structure of the learning process of Artificial Neural Network

Stanislaw Placzek and Bijaya Adhikari
stanislaw.placzek@wp.pl, bijaya.adhikari1991@gmail.com
Vistula University, Warsaw

Abstract

Hierarchical structure is an accepted default way to define ANN structure. This structure can be described using different methods, mathematical tools, software and/or hardware realization. In this article, we are proposing ANN decomposition into hidden and output sub networks. To build this kind of learning algorithm, information is exchanged between sub networks level and coordinator level in every iteration. Learning coefficients are tuned in every iteration. The main coordination task is to choose the coordination parameters in order to minimize both the global target function and all local target functions. In each iteration their values should decrease in asymptotic way to achieve the minimum. In this article, learning algorithms using forecasting of sub networks connectedness is studied.

1 Introduction

Many ANN structures are in practice. The most popular among them is the one with Forward Connections having complete or semi-complete set of weight coefficients. For special needs, ANNs with Forward Cross Connections and Back Connections are used. The full structure of ANN is depicted on Fig.1. To describe the structure, independent of the ANN complexity, partition of layers are used and these partitions are the input layer, one or more hidden layers, and the output layer. Input layer connects ANN with external world (environment) and performs initial processing, calibration or filtering of input data. The hidden layers are used for data processing.

In most common structures, hidden layers include more neurons than input layer and they use non-linear activation function. The output layer, which sums all signals from hidden layers, uses two types of activation functions: linear activation function for approximation tasks and nonlinear sigmoid or tanth activation functions for classification tasks. In this paper, to avoid confusion regarding the number of layers, only the hidden layers and the output layer are included. The concept of layers in ANN structures reflects the silent assumption that ANN structures are hierarchical. Taking this into account as a very important feature of ANN, new description could be created to characterize networks.

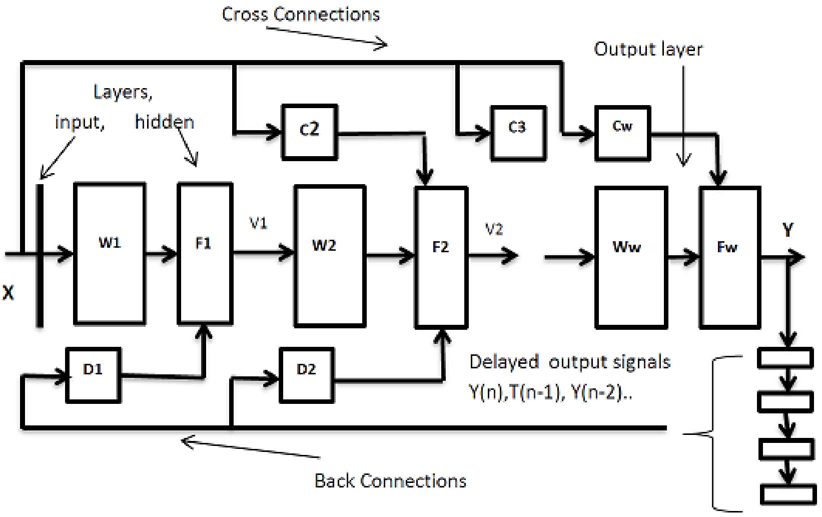


Figure 1: Scheme of the full ANN structure.

1.1 Abstract description of ANN

To analyze ANN structure, verbal description is used so as to help everybody understand how ANN is built. For more detailed analysis, mathematical description using algebraic and/or differential equations is required. Based on these descriptions, ANNs are then implemented by a computer program. So, to achieve complete description of ANN, concepts and models from different fields of science and technology have to be used.

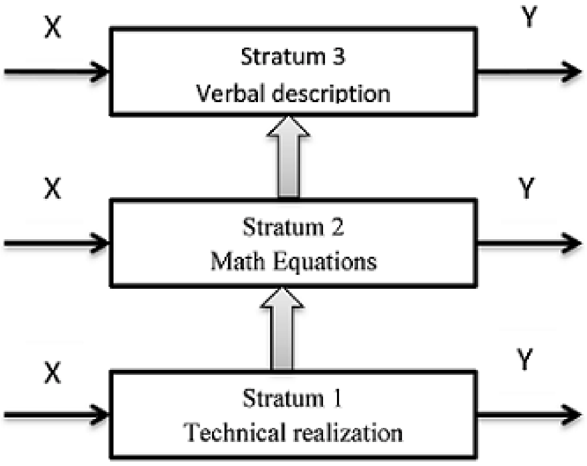


Figure 2: ANN stratification description.

Every model uses its own set of variables and terminology in different abstract level. To describe and understand how a particular ANN works, some hierarchical set of abstract concepts are used. To separate these concepts from the layer description, a new name is used [15] – stratification of ANN into abstract stratum (Fig.2.). For such a hierarchical description, each level should be as independent as possible. Using this definition we can state that:

- the selection of strata, in terms of which an ANN is described, depends on the scientist, their target and needs;
- the concepts in which every stratum is described should be as independent as possible;
- one can comprehensively understand how an ANN works, moving down the strata hierarchy;
- a stratified description implies reduction in information, moving up the strata hierarchy.

The input set X and the output set Y can be represented as Cartesian products. It is assumed that two families of sets are given:

$$X = X_1 \times X_2 \times .... \times X_{n_s} \tag{1}$$

$$Y = Y_1 \times Y_2 \times ... \times Y_{n_s} \tag{2}$$

Where:
n_s -the number of strata in which one describes an ANN structure.
If concepts in which every stratum is described are fully independent, the ANN stratification can be described as:

$$ANN_i : X_i \rightarrow Y_i \tag{3}$$

Where: $i = 1, 2...n_s$

1.2 Calculation complexity or decision taking.

For multi-layered ANN hidden layers and output layer can be sectioned off. Every layer has own output vector that is an input vector of the next layer, vi $i = 1, 2, ...n$. Both hidden layers and output layer can be described as sub - networks. "n" defines the total number of sub networks. ANN logic decomposition depends on layers separated by establishment of extra output vectors vi, where $i = 1, 2, ...n$. Now the network consists of the set of sub- networks, for each of which local target function is defined by $\Phi = (\Phi 1, \Phi 2, ...\Phi n)$.

Similar to ANN structure decomposition, learning algorithm using error back propagation can be decomposed as well. (Fig.3.). We can sort out:

- The first level task in which the minimum of the local target functions Φi $i = 1, 2, ...n$ is searched.
- The second level task which has to coordinate the all first level tasks.

In learning algorithms constructed this way, there is a set of optimization tasks on the first level. The goal of these optimization tasks is to find the minimum value of target function Φ. Unfortunately these are non-linear tasks without constrains. In practice, standard procedures to solve these problems

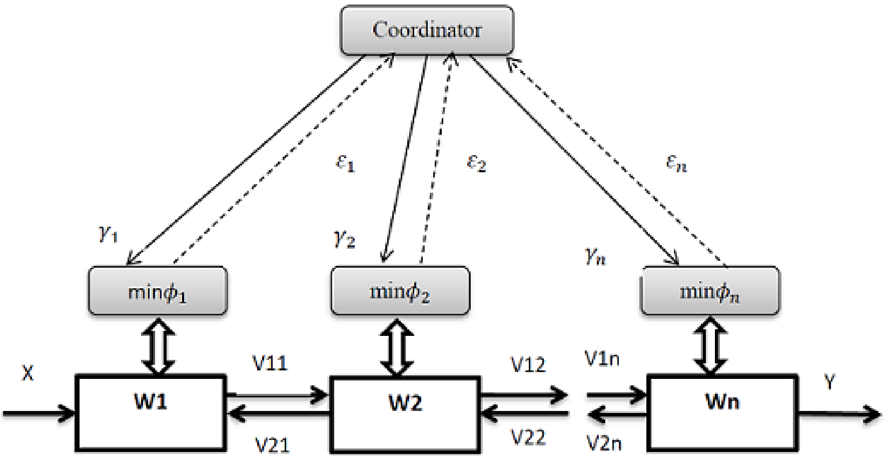


Figure 3: ANN decomposition.

exist. But in two level learning algorithm structure, coordinator is not responsible for solving the global task. Coordinator is only obliged to calculate the value of coordination parameters $\gamma = (\gamma_1, \gamma_2, \dots \gamma_n)$ for every task on the first level. The first level tasks have to use the coordination parameters value searching for the solution. It is an iterative process. Coordinator, in every iteration cycle, receives new values of feedback parameters $\epsilon = (\epsilon_1, \epsilon_2 \dots \epsilon_n)$ from the first level tasks. Using this information coordinator has to make new decisions – calculate the new coordination parameters value. These procedures could be relatively complicated and in the most situations they happen to be non–gradient procedures. In the hierarchical learning algorithm, target functions can be defined as:

- Global target function Ψ,
- Set of local target functions for every sub network Φi where $i = 1, 2, ...n$,
- Coordinator target function Ω.

According to [15][2], solution of the primary task depends on the minimum global target function Ψ. The first level tasks should be built in a way that when all the first level tasks are solved, the final solution must be achieved – the minimum of the global target function. This kind of stratified structure is known as level hierarchy [15].

To summarize we conclude:

- Complexity of the problem increases from the first level to the second. Coordinator needs more time to solve its own tasks.
- Coordination tasks could be non–parametric procedures. To study dynamics of changing target functions value Φ, coordinator should have the ability to change (or changing) learning parameters in the first level tasks. As stressed above all the first level task are non–linear and have to be solved using iterative procedures.

- For different tasks, characteristic of ANN learning processes could be different . Coordinator studying feedback information from the first level tasks should have the ability to change all parameters in the both coordinator and the first level procedures.

2 Decomposition and coordination of ANN learning algorithm

Let us take two layered ANN with one hidden layer and output layer using full internal forward connections into account. Let us assume that this network does not have Cross Forward and Back Connections. These kind of networks can be used for both approximation and classification tasks. According to concept introduced above this ANN can be described by using two strata.

2.1 Verbal description of Structure. Stratum 2.

Aforementioned ANN with full forward connection contains one hidden layer. In this layer connections between input vector X and output vector $V1$ are represented by matrix $W1$. All matrix coefficients are defined. Connections in the output layer are defined by matrix $W2$. Matrices connect input vector $V1$ and output vector Y. In this matrix all weight coefficient are defined. Number of input neurons is defined by vector X which has dimensionality of N_0. In the same way, number of neurons in the output layer is defined by vector Y which has dimensionality of N_2.Number of neurons in the hidden layers, N_1, depends on complexity of problem. Usually $N_1 > N_0$, so data is not compressed in the first layer. Based on the description introduced above, ANN could be set up as hierarchical level structure (Fig.4).In the first level, two local target functions, $\Phi 1$ for the first sub-network and $\Phi 2$ for the second sub-network, are defined. On the second level, coordinator is established. Its main goal is to coordinate all the first level tasks and to achieve the minimum of the global target function Ψ. For the coordinator, two functions G and H are defined which transforms coordination signals $(V21, V12)$ and feedback signals $(V1, V2)$. At the same time, coordinator should have the ability to change value of learning coefficients α_1 and γ_2 by using transformation functions $h_1(\Phi 1, \Phi 2)$ and $h_2(\Phi 1, \Phi 2)$ (Fig.4.).

2.2 Mathematical description of structure of ANN. Stratum 1

In the decomposed ANN structure, we can defined the next target functions:

- Global target function Ψ. For all epoch:

$$\Psi(W1, W2, X, Y) = \sum_{k=1}^{N_2}\sum_{p=1}^{N_p} \Psi_k^p = \frac{1}{2}\sum_{k=1}^{N_2}\sum_{p=1}^{N_p}(y_k^p - z_k^p)^2 \tag{4}$$

  Where:
  $Y[1 : N_2, 1 : N_p]$ - ANN output value,
  $Z[1 : N_2, 1 : N_p]$ - teaching data,
  N_2 - number of output neurons,

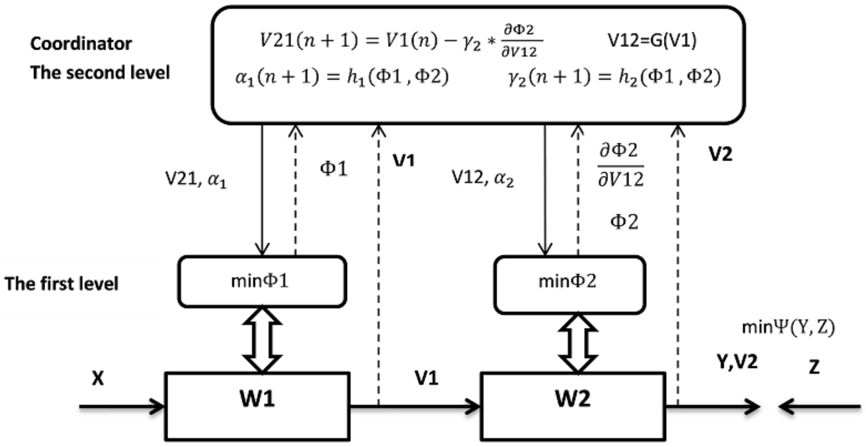

Coordinator
The second level
$V21(n+1) = V1(n) - \gamma_2 * \frac{\partial \Phi 2}{\partial V12}$
V12=G(V1)
$\alpha_1(n+1) = h_1(\Phi 1, \Phi 2)$
$\gamma_2(n+1) = h_2(\Phi 1, \Phi 2)$
V21, α_1
Φ1
V1
V12, α_2
$\frac{\partial \Phi 2}{\partial V12}$
Φ2
V2
The first level
minΦ1
minΦ2
minΨ(Y, Z)
X
W1
V1
W2
Y,V2
Z

Figure 4: Coordination scheme

N_p- dimensionality of training set.
Ψ_k^p- global target function for ”k” output of the second sub-network and p$^{\text{th}}$ element of training set.

- Local target functions for all epoch:

$$\epsilon_1 = \Phi 1(W1, X, V21) = \sum_{i=1}^{N_1} \sum_{p=1}^{N_p} \phi 1_i^p = \frac{1}{2} \sum_{i=1}^{N_1} \sum_{p=1}^{N_p} (v1_i^p - v21_i^p)^2 \tag{5}$$

where:
$V1, V2[1:N_1, 1:N_p]$ - coordination matrices,
N_1 - number of hidden neurons,
N_0 - number of input neurons.
$\Phi 1_i^p$ - local target function for ”i” output of the first sub-network and p$^{\text{th}}$ element of training set.

$$v1_i^p = f(u_i^p) \tag{6}$$

$$u_i^p = \sum_{j=0}^{N_0} W1_{ij} \cdot x_j^p \tag{7}$$

$$\epsilon 2 = \Phi 2(W2, V12, V2) = \sum_{i=1}^{N_2} \sum_{p=1}^{N_p} \Phi 2_k^p = \frac{1}{2} \sum_{i=1}^{N_2} \sum_{p=1}^{N_p} (v2_k^p - z_k^p)^2 \tag{8}$$

$$v2_k^p = f(e_k^p) \tag{9}$$

$$e_k^p = \sum_{i=0}^{N_2} W2_{ki} \cdot v12_i^p \tag{10}$$

Where:
f - sigmoid function,
$i = 1, 2...N_1$ $k = 1, 2, ..N_2$,
$\Phi 2^p_k$- local target function for "k" output of the second sub-network and pth element of training set.

On the first level two minimization task $\Phi 1$ and $\Phi 2$ have to be solved. These target functions have additive structures. Both could be divided into N_1 and N_2 sub-tasks respectively. This can be used to build programming procedures using appropriate programming language. So, we can formulate N_1 sub-tasks

$$min\Phi 1 = min \sum_{i=1}^{N_1} \phi 1_i = \sum_{i=1}^{N_1} \sum_{p=1}^{N_p} (f[\sum_{j=0}^{N_0} W1_{ij} \cdot x^p_j] - v21^p_i)^2 \tag{11}$$

$$\frac{\partial \Phi 1}{\partial W1_{ij}} = \sum_{p=1}^{N_p} (v1^p_i - v21^p_i) \cdot \frac{\partial f}{\partial u_i} \cdot x^p_j \tag{12}$$

$$W1_{ij}(n+1) = W1_{ij} - \alpha 1 \cdot \frac{\partial \Phi 1}{\partial W1_{ij}} \tag{13}$$

For $i = 1, 2, ...N_1 \quad j = 0, 1, 2...N_0$
$\Phi 1_i$-local target function for "i" output of the first sub-network and for whole training set.

In the same way can be formulated N_2 sub-tasks

$$min\Phi 2 = min \sum_{k=1}^{N_2} \phi 2_k = \sum_{k=1}^{N_2} \sum_{p=1}^{N_p} (f[\sum_{i=0}^{N_1} W2_{ki} \cdot v12^p_i] - z^p_k)^2 \tag{14}$$

$$\frac{\partial \Phi 2}{\partial W2_{ki}} = \sum_{p=1}^{N_p} (v2^p_k - z^p_k) \cdot \frac{\partial f}{\partial e_k} \cdot v12^p_i \tag{15}$$

$$\frac{\partial \Phi 2}{\partial v12_i} = \sum_{k=1}^{N_2} \sum_{p=1}^{N_p} (v2^p_k - z^p_k) \cdot \frac{\partial f}{\partial e_k} \cdot W2_{ki} \tag{16}$$

$$W2_{ki}(n+1) = W2_{ki}(n) - \alpha 2 \cdot \frac{\partial \Phi 2}{\partial W2_{ki}} \tag{17}$$

For $\quad k = 1, 2, ...N_2 \quad i = 0, 1, 2, ...N - 1$
$\Phi 2_k$-local target function for "k" output of the second sub-network and for whole training set.

2.3 Coordination description

The primary ANN structure was decoupled into the "N" sub-network. Every sub-network has its own target function – the minimum of the sum of the square error – and it is not fully independent. They are connected by interfaces in both directions: forward and backward. These interactions of sub-processes are the prime reason for the existence of conflict known as the intra-level conflict. Solving its own task, a sub-process cannot ignore the optimization decision to be taken by other sub-processes. Only the coordinator can resolve the conflict.

The coordinator should have the ability to influence the interaction between processes. This can be done using different methods. In [15], three principles are defined:

- Interaction Prediction. The coordination input may involve a prediction of the interface input.
- Interaction Decoupling. Each first-level sub-system is introduced into the solution of its own task and can treat the interface input as an additional decision variable to be free. This means that sub-systems are completely decoupled.
- Interaction Estimation. The coordinator specifies the ranges of interface inputs over which they may vary.

To solve the coordination problem, the Interaction Decoupling principle was selected. The coordinator can be described as :

$$\Omega(\Phi 1, \Phi 2, V1, V2, V12, V21) \tag{18}$$

Which relates to the forward and backward interfaces. If Ω is a vector function, then:

$$V12 = \omega 1(V1) \tag{19}$$

$$V21 = \omega 1(V2, \Phi 2) \tag{20}$$

For a simple coordination structure, one can define:

$$V12 = \lambda 2 \cdot V1 \tag{21}$$

$$V21(n+1) = V21(n) - \gamma 2 \cdot \frac{\partial \Phi 2}{\partial V12} \tag{22}$$

The coordinator and the first-level sub-processes work simultaneously. The information is exchanged between the first and the second levels. The first level tasks calculate control parameters and send them to coordinator. Additionally, in every iteration, coordinator analyzes the local target functions $\phi 1_i(n)$ and $\phi 2_k(n)$. This information is necessary to calculate the new vector value $V21$. At the same time, coordinator should have the ability to interfere in learning process by selecting new value of learning parameters $\alpha_1, \alpha_2, \gamma_2$. Coordinator can calculate the value of target function by itself using data sent to it by the first level. We should stress that values of the target function change dramatically during the learning process. We observed that values of $\phi 1_i(n)$ and $\phi 2_k(n)$ changed significantly during several hundred iterations. At the same, during learning process the values of the target functions can increase to a large value and then decrease drastically. This process explains that ANN, at the beginning of the learning process, has to attune the weight coefficients of the W1 matrix. In the next step both $\Phi 1$ and $\Phi 2$ target functions change their value in an asymptotic way to achieve their minimum. This means that weight coefficients for both $W1$ and $W2$ matrices are near the stable values and only small corrections are pursued. So, coordinator should study not only the target functions but also their dynamic changing process.

3 Example

In this example the main dynamic characteristics of the learning process are shown. The stress is made on the characteristic of the first level local target functions Φ1 and Φ2. The structure of ANN is simple and can be described as ANN(3-5-1). This mean that ANN includes 3 input neurons, 5 neurons in hidden layer, and 1 output neuron. Sigmoid activation functions are implemented in both hidden and output layers. Three arguments of XOR function is fed as input data. So, every epoch includes 8 vectors. Changing different learning parameters as $\alpha_1, \alpha_2, \gamma_2, \beta_1, \beta_2$ dynamic characteristics have been studied.

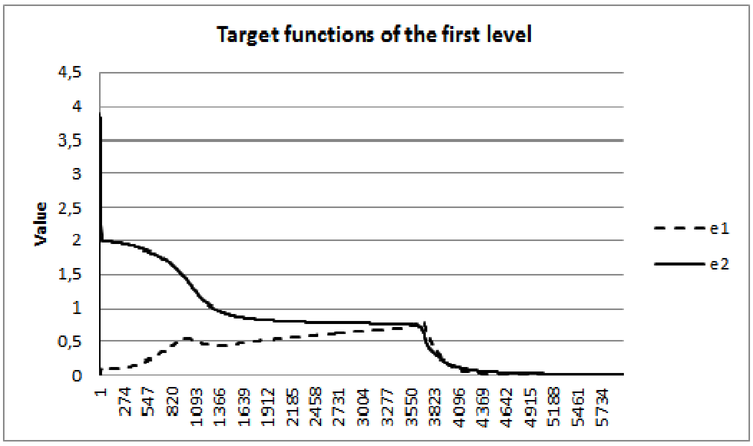


Figure 5: Target functions value for $\alpha 1 = 0.3,\ \ \alpha 2 = 0.05,\ \ \gamma 2 = 0.5$

In the second part of the test, the simple adaptive coordination algorithm was used. Fig.5. shows how the two target functions Φ1 and Φ2 changed their value during learning process (iterations' number). The quality of dynamic processes is different for these two target functions. The function Φ2 represents the second local target function (output layer). This process is smooth. This means that during the learning process the value of Φ2 decreases in a uniform way to the minimum value. Midway through the process, its value decreases very slowly. This is correlated with the first target function Φ1 (hidden layer). From start to 3700 iterations target function Φ1increased its value. Two local maximum in 1000 iterations and 3700 iterations are observed. After that, both Φ1 and Φ2 functions decrease their value and in the asymptotic way to achieve the minimum.

As we stressed in previous sections, hidden layer can be divide into 5 sub-networks. Fig.6. shows the outputs of the three sub-networks $(\phi 1_1, \phi 1_3, \phi 1_4)$. The quality of dynamic characteristics are the same, but maximum values are different.

In the next figure (Fig.7), we can see that the learning process depends on γ_2 parameter. This parameter is calculated by coordinator and has impact on the forecast of the vector $V21$value. For γ_2 =0.1, which is too small, learning process isn't smooth. Small oscillations are observed. But if γ_2=0.5, it is too big and the peak value is more than 5 times greater than for γ_2=0.1. So, coordinator should calculate $\gamma 2$ using its own adaptive algorithm which should receive target

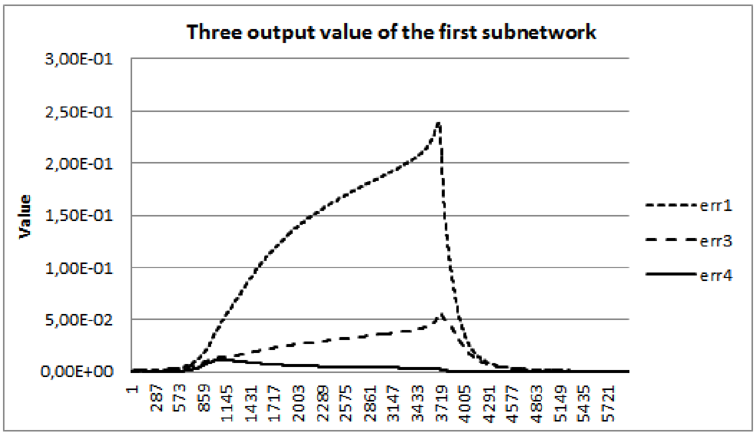


Figure 6: Value of ϕ_1, ϕ_3, ϕ_4 depending on iteration number

functions Φ1 and Φ2 from the first level and analyze them.

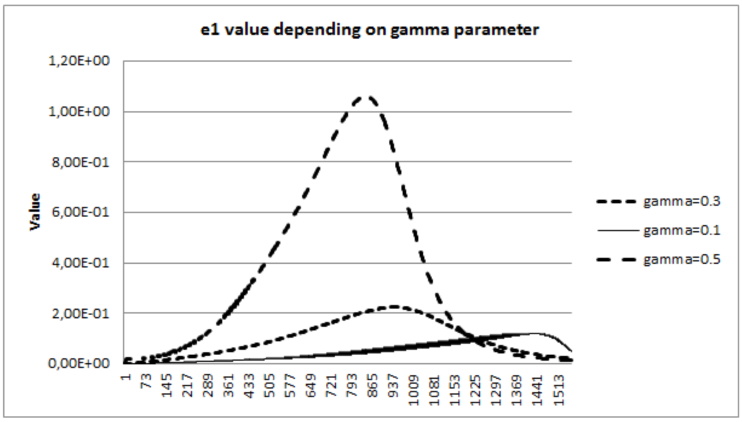
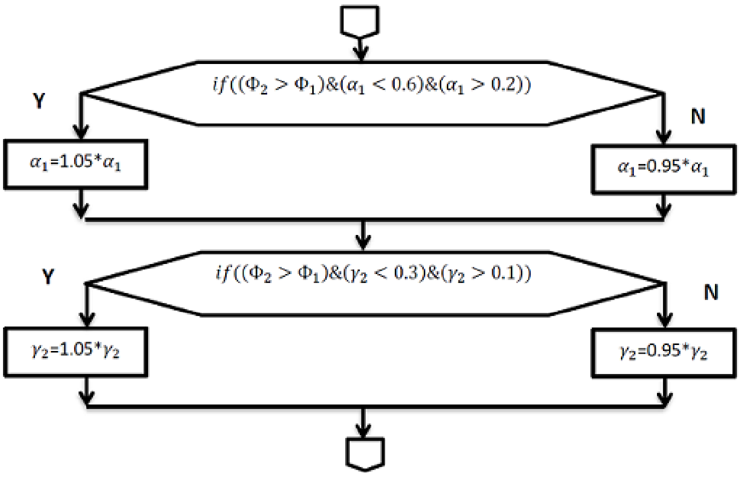


Figure 7: Dynamic characteristic of the learning process for the first layer depending on gamma

To study impact of the coordinator on the quality of learning process, adaptive algorithm changes two parameters α_1 – learning rate for the hidden layer, and γ_2 - learning rate for the vector $V21$. Vector $V21$ forecasts the hidden layer's output (Fig.8). When Φ2 is greater than Φ1 learning rates increases. In this example, their values were increased in very small steps of 0.05. Learning rates α_1 and γ_2 shouldn't be extremely large or small.

Fig. 9 corresponds with Fig. 5 and 6. As stated before, the learning process is dynamic. The sub-networks achieve their stable state (the minimum target functions) at the same time, but the quality of sub-processes is different. In the beginning of the learning process, from 1 to 3700 iterations, the matrix $W1$ achieved its final result ($\Delta W1 = 0$).

However, the matrix $W2$ is far from the final value. In iteration number 3700, $\Delta W2$ achieved its minimum and after that it sharply increased its value



$if((\Phi_2 > \Phi_1)\&(\alpha_1 < 0.6)\&(\alpha_1 > 0.2))$
Y
N
α_1=1.05*α_1
α_1=0.95*α_1
$if((\Phi_2 > \Phi_1)\&(\gamma_2 < 0.3)\&(\gamma_2 > 0.1))$
Y
N
γ_2=1.05*γ_2
γ_2=0.95*γ_2

Figure 8: Coordination algorithm

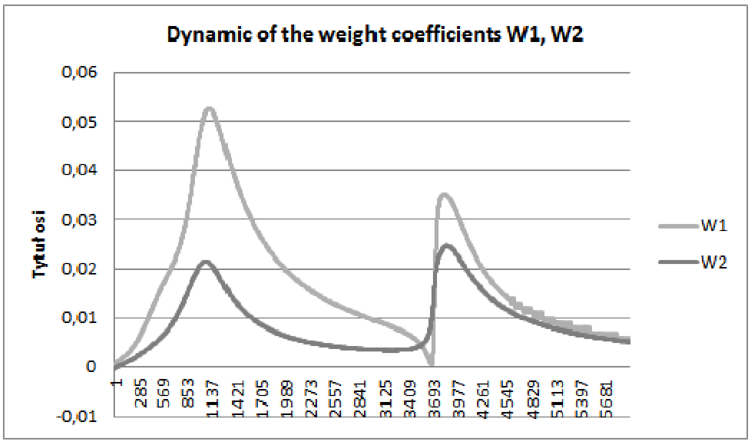


Figure 9: . Dynamic characteristic of the $\Delta W1 = W1(n+1) - W1(n)$ and $\Delta W2 = W2(n=1) - W2(n)$ depending on iteration number

to the maximum. This process entails the first sub-network, which achieves the maximum value as well. After that, the sub-processes achieved a stable value in the asymptotic way. This concluded the learning process.

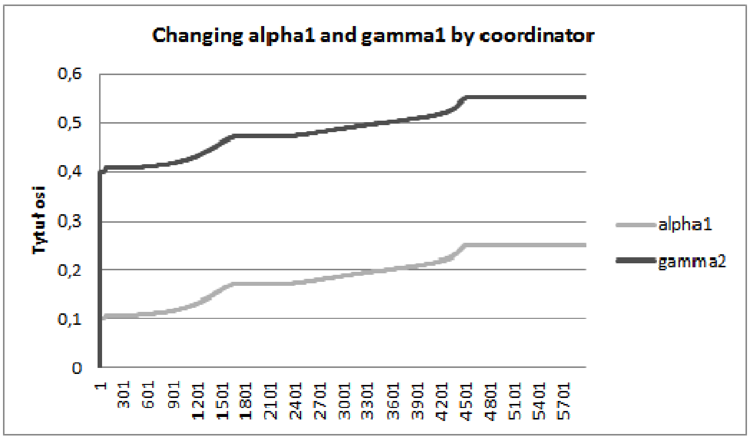


Figure 10: Value of learning coefficients $\alpha 1$, $\gamma 2$ changing by coordinator

Fig.10 shows how the values of two learning rates are changed by coordinator. Both learning coefficients $\alpha 1$ and $\gamma 2$ increase their value. This has a positive impact on the convergence rate. The second part of the learning process (after 4000 iterations) is smooth and asymptotically tends to zero.

In this learning area, both learning coefficients, for the first sub-network $\alpha 1$ and for the coordinator $\gamma 2$, should increase their value.Fig. 11 shows the final learning result. The difference between the teacher's value and the ANN's value is small, less than 1%.

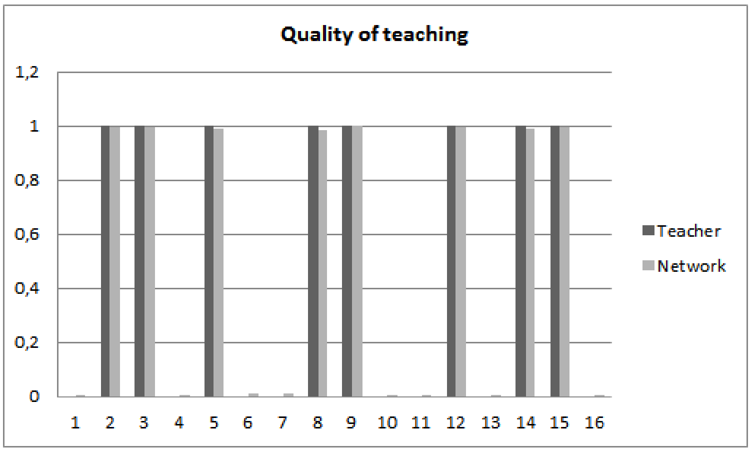
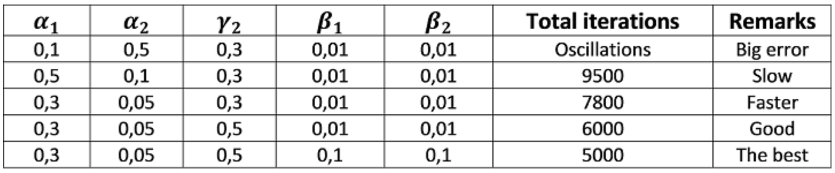


Figure 11: The final learning result

4 Conclusion

In [15], few of coordination principles are defined for big scale hierarchical structures. In this article, the following principle is used - the forecast of the connections between sub-networks. In the hierarchical structure of ANN, coordinator should forecast the value of the vector $V21$. This value should be the same as the real value of the hidden layer output $V1$. In this situation global target function Ψ should achieve its minimum value and then the learning process terminates.

If both of the first level local target functions $\Phi1$ and $\Phi2$ meet few conditions [2][15], then convergence is guaranteed. Unfortunately the global target function Ψ isn't concave and could have multiple local minima. Therefore, it is not possible to prove that algorithm is stable and convergent mathematically. But the first local target functions didn't include any constraints and that helps while building learning algorithm.



α_1	α_2	γ_2	β_1	β_2	Total iterations	Remarks
0,1	0,5	0,3	0,01	0,01	Oscillations	Big error
0,5	0,1	0,3	0,01	0,01	9500	Slow
0,3	0,05	0,3	0,01	0,01	7800	Faster
0,3	0,05	0,5	0,01	0,01	6000	Good
0,3	0,05	0,5	0,1	0,1	5000	The best

Figure 12: Impact of the different learning rates to the quality of the learning process

In the learning processes shown in (Fig.10). all rates were constant. Coordinator calculates new $V21$ value using γ_2. Fig.5 shows that value of target function $\Phi2$ did not change between 2000 and 3700 iterations. This is due to the fact that ANN in the first order has to stabilize the $W1$ matrix weight coefficients. This process depends on $V21$ vector. When all the $W1$ weights coefficients are stable, matrix $W2$ then stabilizes its weight coefficient. In this ANN, the first layer played the most important role. The impact of sub-networks on the final value of the first layer's target function $\Phi1$ is different for different structures. There are ANN structures in which its impact is very small. This can be explained by the hidden layer structure. The hidden layer includes structural neurons redundancy. Finally, the coordination algorithm is analyzed. Learning rates α_1 and γ_2 didn't achieve their maximum value. Possibly, the value of the learning rate should be calculated using not only the relation between $\Phi1$ and $\Phi2$, but also using their dynamic characteristics as the first difference
$\Delta\Phi1(n) = \Phi1(n) - \Phi1(n-1)$ and $\Delta\Phi2(n) = \Phi2(n) - \Phi2(n-1)$.
This implies that coordinator should implement the PID controller algorithm.

$$\gamma2(n+1) = \gamma2(n) + \lambda1 \cdot \Phi1(n) + \lambda2 \cdot (\Phi1(n) - \Phi1(n-1)) \tag{23}$$

These two problems will be studied in the future works.

References

[1] Ch. M. Bishop, Pattern Recognition and Machine Learning , Springer Science + Business Media, LLC, 2006

[2] W. Findeisen, J. Szymanowski, A. Wierzbicki, Teoria i metody obliczeniowe optymalizacji. Państwowe Wydawnictwo Naukowe, Warszawa 1977.

[3] D. J. Montana, L. Davis Training Feed Forward Neural Networks Using Genetic Algorithms. IJCAI Detroit, Michigan 1989.

[4] S. Osowski, Sieci Neuronowe do Przetwarzania Informacji. Oficyna Wydawnicza Politechniki Warszawskiej, Warsaw 2006.

[5] S. Osowski, Sieci neuronowe w ujeciu algorytmicznym . WNT,Warszawa 1996.

[6] Toshinori Munakate, Fundamentals of the New Artificial Intelligence. Second Edition, Springer 2008.

[7] C. Fyle, Artificial Neural Networks and Information Theory , Department of Computing and information Systems, The University of Paisley, 2000.

[8] A. Marciniak, J. Korbicz, J. Kus, Wstepne przetwarzanie danych, Sieci Neuronowe tom 6, Akademicka Oficyna Wydawnicza EXIT, Warsaw 2000.

[9] Z. Mikrut, R. Tadeusiewicz, Sieci neuronowe w przetwarzaniu i rozpoznawaniu obrazow, Sieci Neuronowe tom 6, Akademicka Oficyna Wydawnicza EXIT 2000.

[10] J. R. Rabunal, J Dorado, Artificial Neural Networks in Real-Life Applications, Idea Group Publishing 2006.

[11] S.Placzek, B. Adhikari, Analysis of Multilayer Neural Networks with Direct and Cross-Forward Connection , CS&P Conference in the University of Warsaw, Warsaw 2013

[12] A. Marciniak, J. Korbicz, Neuronowe sieci modularne, Sieci Neuronowe tom 6, Akademicka Oficyna Wydawnicza EXIT 2000.

[13] Zeng-Guang Hou.Madan M.Gupta, Peter N. Nikiforuk, Min Tan, and Long Cheng, A Recurrent Neural Network for Hierarchical Control of Interconnected Dynamic Systems, IEEE Transactions on Neural Networks, Vol. 18, No. 2, March 2007.

[14] L. Rutkowski, Metody i techniki sztucznej inteligencji, Wydawnictwo Naukowe PWN, Warsaw 2006.

[15] M. D. Mesarocic, D. Macko, and Y. Takahara, Theory of hierarchical multilevel systems, Academic Press, New York and London, 1970.