

# A two-level on-line learning algorithm of Artificial Neural Network with forward connections

Stanislaw Placzek  
 IT Faculty  
 Vistula University  
 Warsaw, Poland

**Abstract**—An Artificial Neural Network with cross-connection is one of the most popular network structures. The structure contains: an input layer, at least one hidden layer and an output layer. Analysing and describing an ANN structure, one usually finds that the first parameter is the number of ANN's layers. A hierarchical structure is a default and accepted way of describing the network. Using this assumption, the network structure can be described from a different point of view. A set of concepts and models can be used to describe the complexity of ANN's structure in addition to using a two-level learning algorithm. Implementing the hierarchical structure to the learning algorithm, an ANN structure is divided into sub-networks. Every sub-network is responsible for finding the optimal value of its weight coefficients using a local target function to minimise the learning error. The second coordination level of the learning algorithm is responsible for coordinating the local solutions and finding the minimum of the global target function. In the article a special emphasis is placed on the coordinator's role in the learning algorithm and its target function. In each iteration the coordinator has to send coordination parameters into the first level of sub-networks. Using the input  $X$  and the teaching  $Z$  vectors, the local procedures are working and finding their weight coefficients. At the same step the feedback information is calculated and sent to the coordinator. The process is being repeated until the minimum of local target functions is achieved. As an example, a two-level learning algorithm is used to implement an ANN in the underwriting process for classifying the category of health in a life insurance company.

**Index Terms**—Keyword1; neural network, learning algorithm, hierarchical structure, decomposition, coordination

## I. INTRODUCTION

In practice many ANN structures are used but the most popular are the ANNs with forward connections that have a complete or semi-complete set of weight coefficients. The structure of an ANN is depicted in (Fig.1). Neurons in both the hidden and the output layers use sigmoid or tanh activation functions. In the output layer the linear activation function is usually used for approximation tasks. In the most common structures hidden layers include more neurons than input layers, so input information is not compressed in the hidden layers. In this paper two assumptions are accepted:

- To define an ANN structure only the hidden layers and output layer are included. A network described as ANN (10-15-8) includes 10 neurons in the input layer, 15 neurons in one hidden layer and 8 in the output one. It is a two-layer ANN.

- To implement a two-level learning algorithm, an ANN with one hidden layer is used.

The concept layer is used in the primary sense.

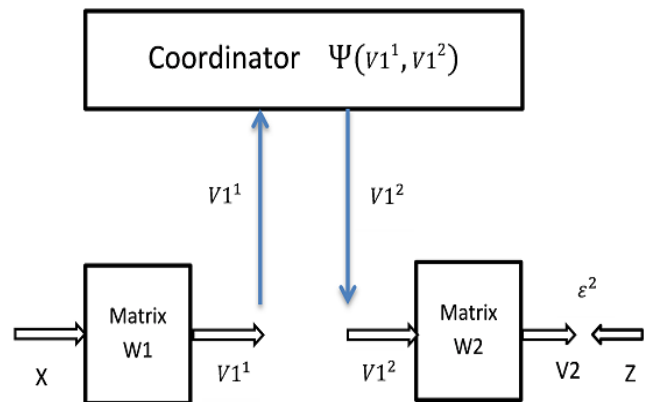


Fig. 1: Scheme of the ANN with forward connections

Using (Fig.1) symbols a set of forward and back formulas can be written.

For forward

$$X \rightarrow U1 \rightarrow V1^1 \rightarrow V1^2 \rightarrow U2 \rightarrow V2 \rightarrow \epsilon^2 \leftarrow Z \quad (1)$$

For the first layer (the hidden layer)

$$u1_i = \sum_{j=0}^{n_0} W1_{ij} \cdot x_j \quad (2)$$

$$v1_i^1 = f_1(u1_i) \quad (3)$$

The coordinator is described by the  $\Psi$  function

$$\Psi(V1^1, V1^2) \quad (4)$$

For the second layer (the output layer)

$$u2_k = \sum_{i=0}^{n_1} W2_{ki} \cdot v1_i^2 \quad (5)$$

$$v2_k = f_2(u2_k) \quad (6)$$

The target function

$$\epsilon^2 = \frac{1}{2} \cdot \sum_{k=1}^{n_2} (v2_k - z_k)^2 \quad (7)$$

Where:

$j = 0, 1, ..n_0$  - number of input neurons,  
 $i = 0, 1, ..n_1$  - number of hidden neurons,  
 $k = 1, 2, ..n_2$  - number of output neurons.

Using a standard backpropagation notation, derivatives with respect to the weight coefficients are achieved

$$\frac{\partial \epsilon^2}{\partial w2_{ki}} = (v2_k - z_k) \cdot f'_2(u2_k) \cdot v1_2 \quad (8)$$

$$\frac{\partial \epsilon^2}{\partial w1_{ij}} = \sum_{k=1}^{n_2} (v2_k - z_k) \cdot f'_2(u2_k) \cdot w2_{ki} \cdot f'_1(u1_i) \cdot x_j \quad (9)$$

Equation (4) is known as coordination function.

## II. TYPES OF HIERARCHICAL MODELS

Using concepts described in [1], an ANN will be treated as a complex system in an internal hierarchical structure. Three terms are introduced in relation to an ANN:

- The layer of both an ANN and a learning algorithm description or abstraction,
- The layer of algorithm complexity,
- The layer of algorithm structure.

To distinguish between these concepts, the following three terms: a stratum, a level, and an echelon, are used respectively. The term layer is used as a common term referring to any of the aforementioned concepts. For future use of the formal description of different concepts of the hierarchical structures, we describe an ANN as a relation between sets X and Y.

$$ANN : X \rightarrow Y \quad (10)$$

Where:

ANN is a function

X an input set,  $X = (x_0, x_1, x_2, \dots, x_{n_0})$

Y an output set or response of the ANN,  $Y = (y_1, y_2, \dots, y_{n_2})$ .

### A. ANN layers of description or abstraction

To treat an ANN as a complex system and to describe it in a complete and detailed way, a different approach should be used. There arises the dilemma of the simplicity of description and the complete understanding of an ANN's behavior [1]. At the first stage, a verbal description is used to help understand how an ANN is built. For a more detailed analysis, mathematical descriptions using algebra and/or differential equations are required. Finally, math formulas have to be implemented in a computer program or an electronic device. Therefore, to achieve a complex description of an ANN, a family of concepts and models from different fields of science and technology have to be used. Every model uses its own set of

variables, laws, principles and terminology by means of which an ANN is described. For such a hierarchical description the functioning on any level should be as independent as possible. To separate this concept of hierarchy from others, a new name is used, a stratified ANN or a stratified description [1]. The layer of abstraction will be referred as stratum (Fig. 2).

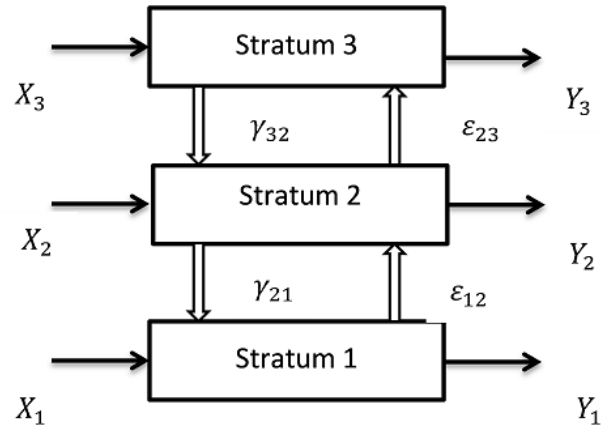


Fig. 2: The ANN stratification description

Using this definition we can state that [1]:

- 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 is working, moving down from the hierarchy of strata.
- a stratified description implies a reduction in information sent up the hierarchy by the reduction of information.

The input set X and the output set Y are both representable as Cartesian products. It is assumed that there are given two families of sets :

$$X = X_1 \times X_2 \times \dots \times X_{n_s} \quad (11)$$

$$Y = Y_1 \times Y_2 \times \dots \times Y_{n_s} \quad (12)$$

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:

$$ANN_i : X_i \rightarrow Y_i \quad (13)$$

Where:  $i = 1, 2, \dots, n_s$

### B. Organisational hierarchy

For a multi-layer ANN a lot of hidden layers and one output layer are sectioned off. The smaller part will be described as a sub-network. Every sub-network has its own output vector that is, at the same time, an input vector of the succeeding one

$V^{ij}, i = 1, 2, \dots, n-1, j = 2, 3, \dots, n$ , where  $n$  is the number of sub-networks. Because of the specific organisation of an ANN's hierarchy there are a lot of sub-networks on the first level, for each of which local target functions are defined:

$$\Phi = (\phi_1, \phi_2 \dots \phi_n) \quad (14)$$

These sets of local tasks have to be coordinated to achieve the global solution. The coordinator, as an independent task, will have its own target function  $\Psi$ . Taking everything into account, this concept is the base on which one may build the new scheme of the ANN learning algorithm structure (Fig. 3). It is the hierarchical organisational structure. To distinguish this concept of hierarchy structure from others, the echelon is used.

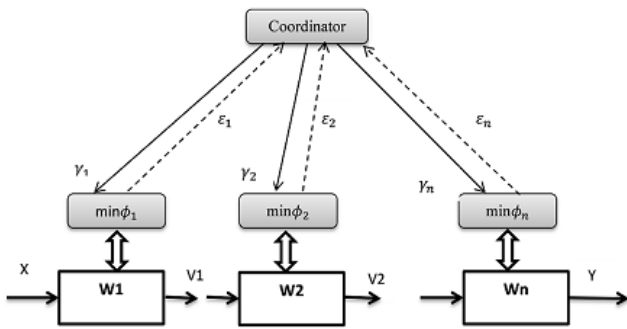


Fig. 3: The hierarchical structure of an ANN learning algorithm

The two-level ANN learning algorithm can be described as a set of procedures. The procedures on the first level are responsible for solving their local tasks and calculating the part of matrix weight coefficients. The second-level procedure has to coordinate all the local procedures (tasks) using its own local target function. There is the vertical interaction between the procedures and two types of information are sent. One is a downward transmission of control signals:

$$\Gamma = (\gamma_1, \gamma_2 \dots \gamma_n) \quad (15)$$

The second is upward from the first level to the second. It is a feedback signal that informs the coordinator about the behaviour of the first-level tasks:

$$E = (\epsilon_1, \epsilon_2 \dots \epsilon_n) \quad (16)$$

Consequently, in all the structures, three different task are defined:

- the global target function

$$\Phi : X \rightarrow Y \quad (17)$$

- a set of the first-level tasks (the first level task)

$$\phi_1 : X \times \gamma_1 \rightarrow V_1 \quad (18)$$

$$\phi_i : V_i \times \gamma_i \rightarrow V_{i+1} \quad (19)$$

$$\phi_n : V_n \times \gamma_n \rightarrow Y \quad (20)$$

Where:

$$i = 1, 2, \dots, n-1$$

- a coordinator task  $\Psi$

$$\Psi : E \rightarrow \Gamma \quad (21)$$

Where:  $\Gamma = (\gamma_1, \gamma_2 \dots \gamma_n), E = (\epsilon_1, \epsilon_2 \dots \epsilon_n)$

To build the two-level learning algorithm two assumptions have been made:

- There is no explicit relation between the procedures on the first level for direct communication. The procedures are using only input and output vectors and a coordinator signal.
- There is no direct relation between the global target function  $\Phi$  and the coordinator task  $\Psi$ .

### C. Levels of calculation complexity

The standard ANN learning algorithm is a non-linear minimisation task without constraints. To solve this task, iteration procedures are used. Using the most popular back propagation algorithm, one has to choose a lot of control parameters. From a theoretical point of view one can have only general suggestions and recommendations regarding the choice of real parameters, for example learning parameters. The algorithm is time-consuming and its convergence is not fast. Dividing the primary algorithm for all ANNs into the sub-network tasks, the local target functions are simpler and can be used in different procedures. Additionally, a new procedure is needed: the coordination procedure. In practice, however, the coordinator does not have the ability to find all the parameters needed for the first-level procedures. To solve this problem, a multi-level decision hierarchy is proposed [1]. Solving the problems in the iteration algorithm on both the first and the second level, one can observe certain dynamic processes. These processes are non-linear and use a lot of control parameters. During the learning process these parameters are stable and do not change. Practice proves that this solution is not optimal. To control the way learning parameters change, an additional level could be used the adaptation level (Fig. 4).

Thus, one can build three levels at a minimum:

- The local optimisation procedures: the algorithm is defined directly as a minimisation task without constraints.
- The coordination procedure: this algorithm could be defined directly as a minimization of the target function as well. Constraints could exist or not.
- The adaptation procedure: the task or procedure on this level should specify the value of learning parameters not only for the coordinator level, but also on the first level. To solve this task, a procedure should achieve dynamic characteristic of the learning process on all the levels.

As a conclusion one can state that the complexity of the problem increases from the first level to the next one. The coordination and adaptation procedures need more time to solve their own procedure.

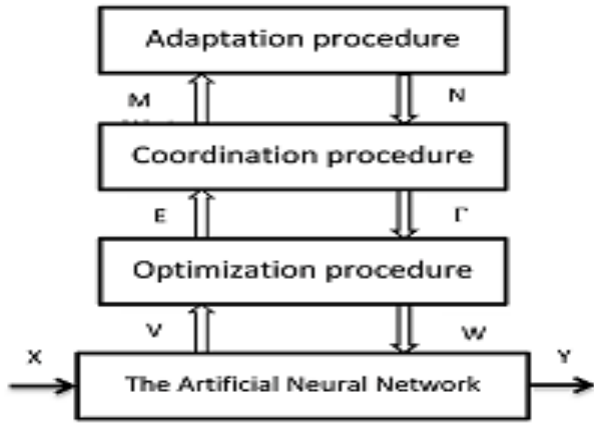


Fig. 4: Functional multi-level hierarchy for an ANN learning algorithm

### III. TWO-LEVEL ALGORITHM STRUCTURE

The two-layer ANN with an input layer, one hidden layer and an output layer can be used for further considerations. This simple structure is very popular and by using it one can solve a lot of practice tasks. Since this network is used to solve different classification tasks, sigmoid activation functions are used. To decompose the standard learning algorithm structure into a sub-network task, the coordination target function has to be built. The two-level learning algorithm structure for the ANN with one hidden layer is shown in (Fig. 5).

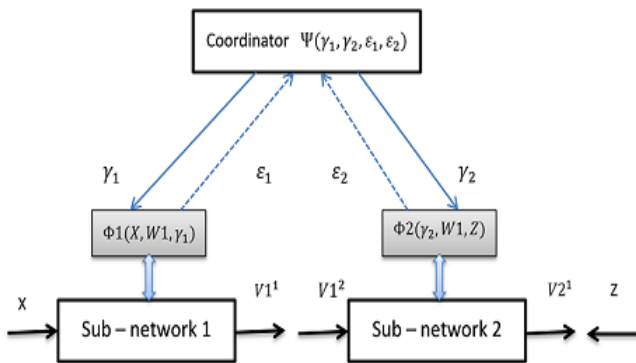


Fig. 5: Scheme of two-level learning algorithm structure

According to [2][8] the following set of formulas can be written.

1) For the first sub-network: The local target function  $\Phi 1$  is defined as error-mean-square

$$\Phi 1(W 1, X, \gamma 1)=\frac{1}{2} \sum_{i=1}^{n_1}\left(v 1_i-\gamma 1_i\right)^2 \quad (22)$$

Other relations

$$u 1_i=\sum_{j=0}^{n_1} W 1_{i j} \cdot X_j \quad (23)$$

$$v 1_i=f_1\left(u 1_i\right) \quad (24)$$

Where:

$\gamma 1$ — a target value given by the coordinator

Total derivatives with respect to the weight coefficients of matrix  $W 1$ .

$$\frac{\partial \Phi 1}{\partial W 1_{i j}}=\left(v 1_i-\gamma 1_i\right) \cdot f_1^{\prime}\left(u 1_i\right) \cdot X_j \quad (25)$$

A sigmoid derivative function

$$f_1^{\prime}\left(u 1_i\right)=a \cdot v 1_i \cdot\left(1-v 1_i\right) \quad (26)$$

The new value of weight coefficients

$$W 1_{i j}(n+1)=W 1_{i j}(n)-\alpha_1 \cdot \frac{\partial \Phi 1}{\partial W 1_{i j}}+\beta_1 \cdot\left[W 1_{i j}(n)-W 1_{i j}(n-1)\right] \quad (27)$$

Where:

$\alpha_1$  -the learning coefficient

$\beta_1$  - the regularisation coefficient

The feedback information sent by the first sub-network to the coordinator

$$\epsilon 1_i=v 1_i \quad (29)$$

Where:

$i=1,2, \dots, n_1$

2) For the second sub-network: The local target function  $\Phi 2$  is also defined as error-mean-square.

$$\Phi 2\left(W 2, \gamma 2, Z\right)=\frac{1}{2} \sum_{k=1}^{n_2}\left(v 2_k-z_k\right)^2 \quad (30)$$

Other relations

$$v 2_k=f_2\left(u 2_k\right) \quad (31)$$

$$u 2_k=\sum_{i=0}^{n_1} W 2_{k i} \cdot \gamma 2_i \quad (32)$$

Where:

$\gamma 2_i$  the input value given by the coordinator.

Total derivatives with respect to the weight coefficients of matrix  $W 2$ .

$$\frac{\partial \Phi 2}{\partial W 2_{k i}}=\left(v 2_k-z_k\right) \cdot f_2^{\prime}\left(u 2_k\right) \cdot \gamma 2_i \quad (33)$$

And the sigmoid derivative

$$f_2^{\prime}=a \cdot v 2_k \cdot\left(1-v 2_k\right) \quad (34)$$

The new value of the weight coefficients

$$W 2_{k i}(n+1)=W 2_{k i}(n)-\alpha_2 \cdot \frac{\partial \Phi 2}{\partial W 2_{k i}}+\beta_2 \cdot\left[W 2_{k i}(n)-W 2_{k i}(n-1)\right] \quad (35)$$

$$+ \beta_2 \cdot\left[W 2_{k i}(n)-W 2_{k i}(n-1)\right] \quad (36)$$

As we state in (30), the local target  $\Phi_2$  is the function of  $W_2$  weight coefficient and  $\gamma_2$  parameters given by the coordinator. The total derivatives with respect to the coordination parameters  $\gamma_2$  are:

$$\frac{\partial \Phi_2}{\partial \gamma_2} = \sum_{k=1}^{n_1} (v_{2k} - z_k) \cdot f_2'(u_{2k}) \cdot W_{2ki} \quad (37)$$

New feedback information sent by the second sub-network to the coordinator:

$$\epsilon_{2i}(n+1) = \gamma_{2i}(n) - \alpha_3 \cdot \frac{\partial \Phi_2}{\partial \gamma_{2i}} \quad (38)$$

3) *For the coordinator:* In a two-level learning algorithm, the coordinator plays the main role. It is now time to decide what kind of coordination principle will be chosen. This principle specifies various strategies for the coordinator and determines the structure of the coordinator. In [1] three ways were introduced in which the interaction could be performed.

- 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.

In this article the Interaction Prediction is used. The coordinator predicts the interface between sub-networks. This means that the output of the first sub-network  $V_1^1$  and the input of the second sub-network  $V_1^2$  are predicted. The signal  $\gamma_1$  predicts the output signal of the first sub-network  $V_1^1$ . The first sub-network uses this signal as a teachers value and it is a part of the target function  $\Phi_1$  of the first sub-network according to formula (22). The coordinator predicts signal  $\gamma_2$  as well. The second sub-network uses this signal as the input value  $V_1^2$ . Using this assumption, it gives the ability to define the local target function  $\Phi_2$  of the second sub-network (30). Consequently, two local target functions  $\Phi_1$  and  $\Phi_2$  can be defined. As stated above, the coordinator needs the feedback information from the first-level sub-networks checking if the predicted signals  $\gamma_1, \gamma_2$  were true. If not, the coordinator using its own target function should find the new value of the coordination signal  $\gamma_1, \gamma_2$ . The first sub-network using the formula (24) calculates the new value of its output signal which, at the same time, is the feedback signal  $\epsilon_1$  to the coordinator (29). The second sub-network is trying to minimise the local target function  $\Phi_2$  and calculate the new optimal value of input signal  $\epsilon_2$  (38), which is sending to the coordinator. Therefore, the coordinator has full information and is ready to calculate and predict the new value of the coordination input signal  $\gamma_1, \gamma_2$ . Taking into account that,

$$\gamma_1 = \gamma_2 = \gamma \quad (39)$$

the coordinator target function is defined as:

$$\Psi = \frac{1}{2} \cdot \sum_{i=1}^{n_1} (\epsilon_{1i} - \gamma_i)^2 + \frac{1}{2} \cdot \sum_{i=1}^{n_1} (\gamma_i - \epsilon_{2i})^2 \quad (40)$$

Using gradient algorithm one can calculate using gradient algorithm one can calculate

$$\frac{\partial \Psi}{\partial \gamma_i} = (\gamma_i - \epsilon_{1i}) + (\gamma_i - \epsilon_{2i}) \quad (41)$$

$$\frac{\partial \Psi}{\partial \gamma_i} = 2 \cdot \gamma_i - (\epsilon_{1i} + \epsilon_{2i}) \quad (42)$$

The new value of the coordinator signal  $\gamma_1, \gamma_2$

$$\gamma_{1i}(n+1) = \gamma_{2i}(n+1) = \gamma_i(n) - \lambda_1 \cdot \frac{\partial \Psi}{\partial \gamma_i} \quad (43)$$

$$\gamma_{1i}(n+1) = \gamma_{2i}(n+1) = \gamma_i \cdot (1 - 2 \cdot \lambda_1) + \lambda_1 \cdot (\epsilon_{1i} + \epsilon_{2i}) \quad (44)$$

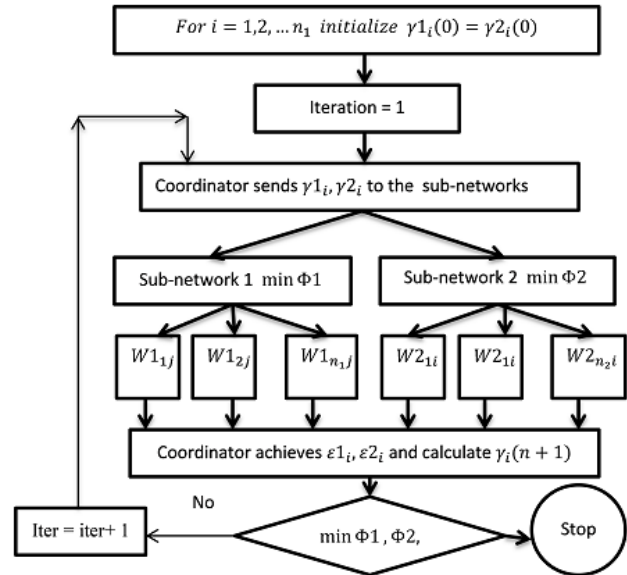


Fig. 6: Functional multi-level hierarchy for an ANN learning algorithm

#### IV. EXAMPLE

In a life insurance company the underwriting process has been playing the main role in risk control and premium calculation. The ANNs could be used to help the insurance agents to classify the insurance applicant and calculate the first level of premium. Therefore, a special short questionnaire was prepared which only includes 10 main questions. The data were used to teach the ANN work as an insurance specialist, known as the underwriter. All data were divided into three subsets:

- The first set of both the input data X and the output data Z included 250 records of data. This set is known as the learning set. As an example, a small part of the input data is shown in the (Fig.7). The learning epoch includes 250

vectors that are sending into the ANN input one by one. When this sequence of the process is finished, the next iteration begins.

- The second set is used to verify the quality of the learning process. This set contains 150 records. It is known as verification set.
- Finally, the third set, known as the testing set, contains only 100 records. It helps the decision-making specialist to decide whether the ANN achieves good quality and if it is ready for use.

Nr	X1	X2	X3	X4	X5	X6	X7	X8	X9	X10
	Sex	Years old	Weight in kg	Growth in cm	Smoking	Father's long life	Mather's long life	Children number	Living condition	Living place
1	1	48	70	190	0	68	67	0	1	2
2	1	62	71	182	0	82	91	1	0	0
3	0	64	83	180	5	89	100	2	1	2
4	1	40	107	177	10	74	66	2	1	2
5	1	58	120	183	20	90	77	3	1	1
6	1	56	80	174	0	87	75	1	1	2
7	1	60	108	186	0	88	83	3	1	0
8	1	47	91	182	0	78	79	1	1	3
9	1	52	71	186	0	71	81	0	1	3
10	1	62	77	177	0	96	81	2	0	1

Fig. 7: An input data example

To achieve this, the two-level learning algorithm has been used to teach the ANN. The structure of the ANN includes only one hidden layer: 10 input neurons as the dimensionality of the vector X, 15 neurons in the hidden layer and 8 neurons in the output layer. This structure can be shortly described as the ANN (10-15-8). Two sub-networks were introduced in accordance with the algorithm description. The first sub-network includes the hidden layer and its local target function  $\Phi_1$ . The second sub-network includes the output layer with its local target function  $\Phi_2$ . The coordinator has its own local target function  $\Psi$  and coordinates local tasks to achieve the minimum of the global target function  $\Phi$  (the whole ANN). The main goal of this example is to study the dynamic characteristic of the ANN learning process, especially the relations between all the target functions: two local target functions  $\Phi_1$  and  $\Phi_1$ , the coordinator target function  $\Psi$  and the global target function  $\Phi$ . All values have to achieve the minimum value according to the relation, as shown in formula (22)(30)(40).

In Fig. 8. the dynamic characteristic of the learning process of the first sub-network is shown. In the beginning phase, the target function  $\Phi_1$  decreases its value from 1.2 to less than 0.1 during 4,000 iterations. After that, it very slowly decreases its value to the target value 0.001.

To study the dynamic characteristic of the second sub-network (Fig. 9.), one can say that in the beginning phase of the learning process, errors occur less frequently and achieve value 0.1 after only 2,000 iterations. The differences between the sub-networks can be explained by the dimensionality of  $W_1$  and  $W_2$  matrices. The matrix  $W_1$  includes 165 weight coefficients while  $W_2$  includes only 128 weight coefficients.

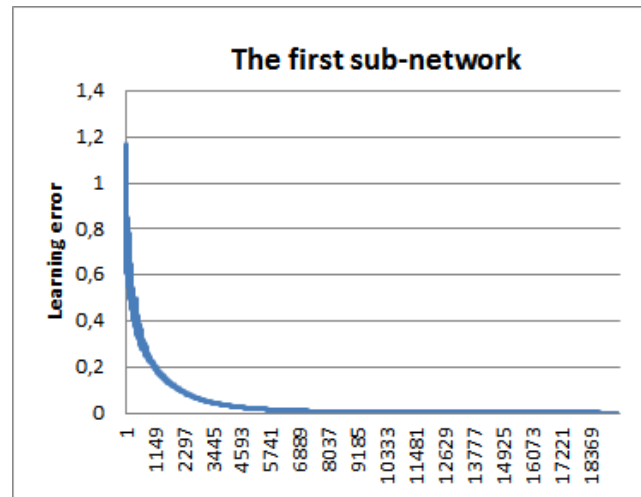


Fig. 8: The target function  $\Phi_1$  of the first sub-network depending on the iteration number

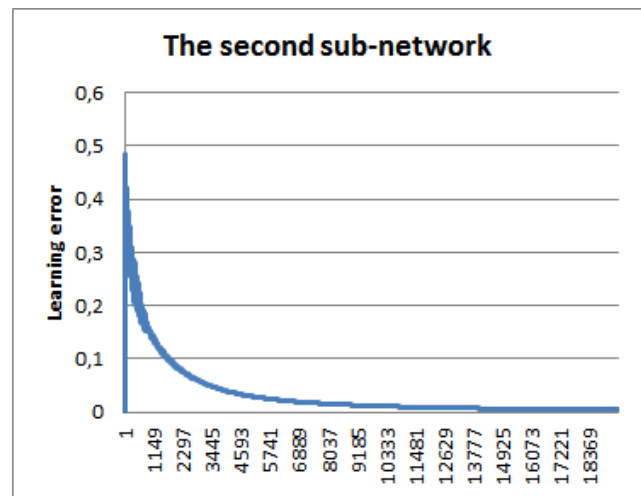


Fig. 9: The target function  $\Phi_2$  of the second sub-network depending on the iteration number

Therefore, in the second sub-network the learning process achieves the minimum value faster.

Finally, the coordinator target function is shown in Fig. 10. The quality of the dynamic processes is the same for both  $\Phi_1$  and  $\Phi_2$ . The starting error is the greatest and achieves its value 0.1 after 6,000 operations, which can be explained by the relations between the sub-networks. When the learning process started, the sub-networks were not connected (decoupled). This means that every sub-network has to change both the weight coefficient and the input– output vectors using the coordinator gamma signal. The coordinator calculates the optimal  $\gamma$  value using feedback information  $\epsilon_1$ ,  $\epsilon_2$  from both sub-networks. This is the iteration process and it has two stages. During the first stage all errors decrease their value dramatically, rather quickly achieving value less than 0.1. After that, the process stabilises and achieves the final value after a

considerably long time (number of iterations). This part of the algorithm is not optimal and the coordinator should change the strategy to calculate a new  $\gamma$  value.

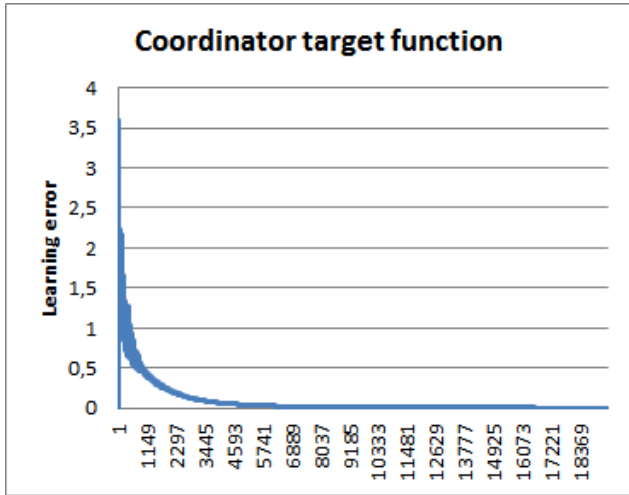


Fig. 10: The coordinator target function value  $\Psi$  depending on the iteration number

It should be emphasised that in the first stage all the dynamic processes are having vibrations. This can be explained by the fact that when trying to increase the learning process, learning coefficients  $\alpha_1$ ,  $\alpha_2$  and  $\alpha_3$  have a rather big value, namely 0.8. When learning coefficients are decreased, dynamic processes are more stable, without the vibrations. In Fig.11. a part of the iteration process is shown.

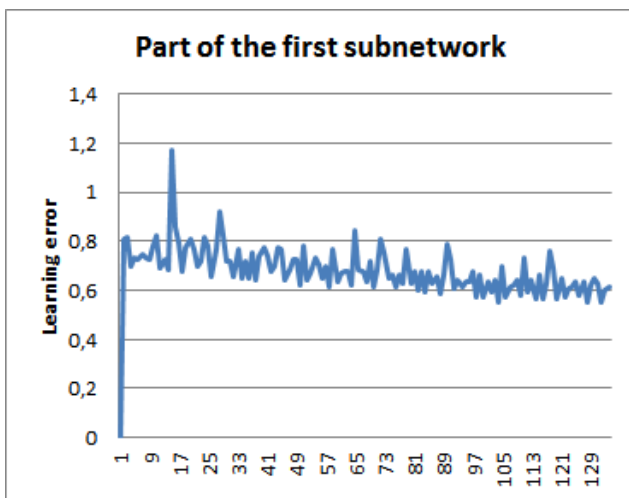


Fig. 11: The part of the learnig process including vibrations

When the ANN achieves the final learning result, the verification sets are used and the differences between the teachers data and the ANN's calculations are collocated (Fig.12). It can be seen that not all the output vectors are the same. In most cases, the ANN calculates less than the teacher (an insurance

specialist). The chart shows a number of categories for a couple of insurance candidates. When a category is higher, the insurance premium is greater as well.

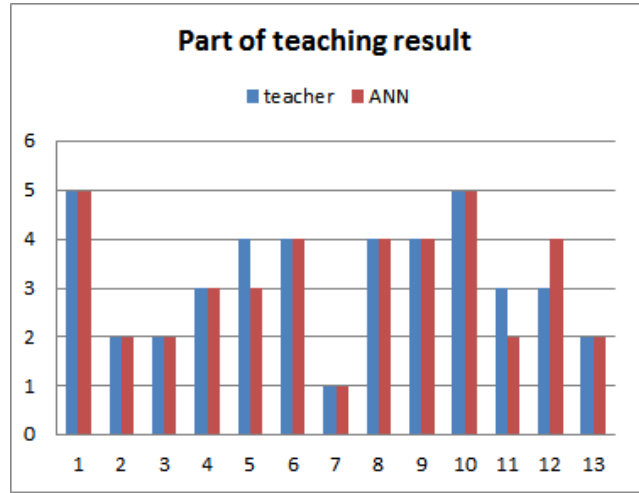


Fig. 12: Result of the ANN's learning

Therefore, one can state that the ANN is more conservative than a life insurance company specialist. For a few candidates it calculates the higher category and they would pay higher premium. Finally, the dynamic characteristic of the global target function is shown (Fig. 13). The characteristic is closely related to all the above. The maximum value is higher and the process needs more time to achieve value less than 0.1, namely after about 8,000 iterations (the sum of the local values).

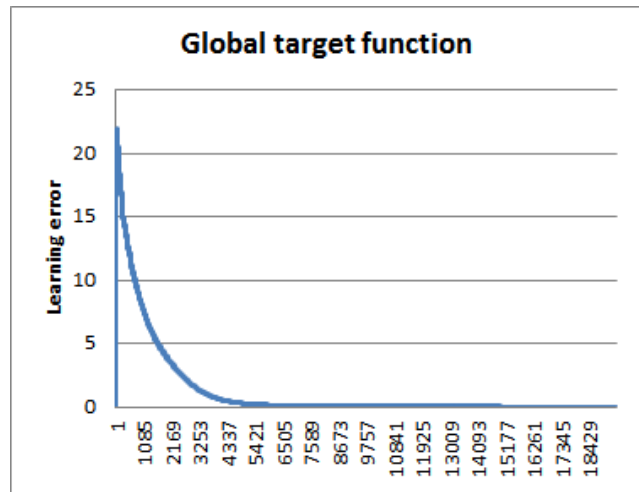


Fig. 13: Learning the global target function  $\Phi$  for the ANN

Characteristics of the feedback signals  $\epsilon_1(n)$  end  $\epsilon_2(n)$  are depicted on Fig.14.

## V. CONCLUSION

In [1] for the big systems with hierarchical structure, three coordination principles are defined. For the ANN learning process interaction prediction was used. Each sub-network is

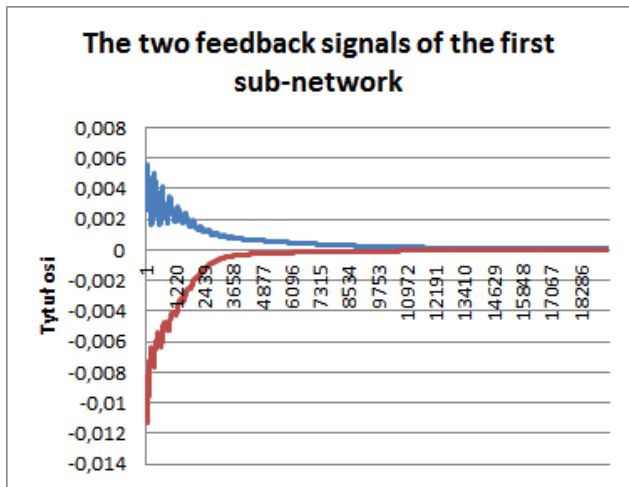


Fig. 14: Example of dynamic characteristic of  $\epsilon_{1_1}$  and  $\epsilon_{1_2}$

responsible for finding the minimum value of its own target function to treat the interface inputs as additional variables. The  $\gamma$  signal plays this role. For the first sub-network,  $\gamma_1$  is used as the teacher data and the sub-network should change its own weight coefficients in such a way that the final sub-networks output should be as close to the teacher value as possible (in a square error sense, of course). For the second sub-network, gamma works as the input vector. This vector and the teacher vector are used to train the sub-network. The coordinator is responsible for finding the optimal value of the  $\gamma_1$  signal, using its own target function  $\Psi$ . The underwriting process was used to show that this learning algorithm structure is able of finding the minimum of the global target function  $\Phi$  for quite a complicated problem and the ANN is ready to work. The weight coefficient for both  $W_1$  and  $W_2$  matrices were memorised and the program was sent to insurance agents to use. As it has been emphasised, during the second stage, the learning algorithm works far away from the optimal value. Convergence errors decrease their values very slowly and (Fig.15.) reaffirmed the quality of the dynamic characteristics:

$$\Delta W_1(n) = W_1(n) - W_1(n - 1) \quad (45)$$

and

$$\Delta W_2(n) = W_2(n) - W_2(n - 1) \quad (46)$$

Where:

$n$  – number of iterations

Therefore coordinator should use a more complicated coordination algorithm that has to include not only a PD algorithm structure but a PID algorithm as well. This work will be continued. Analysing the learning result shown in Fig. 12. one can see that the ANN should find a discrete category value (from the set 1 to 8). From time to time, the network solutions are different than the specialist's decisions. Shifting the solution into fuzzy sets could solve this conflict. What follows is the suggestion that the second sub-network should

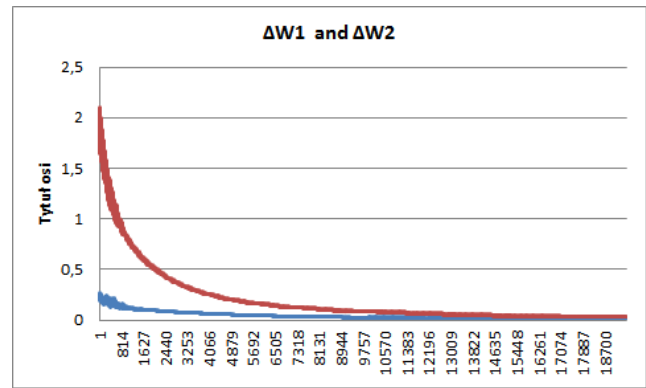


Fig. 15: Example of dynamic characteristic of  $\Delta W_1$  and  $\Delta W_2$  matrixes

include in its structure the logic module based on the fuzzy solutions.

#### ACKNOWLEDGMENT

The authors would like to thank...

#### REFERENCES

- [1] M. D. Mesarovic, D. Macko, and Y. Takahara, Theory of hierarchical multilevel systems, Academic Press, New York and London, 1970.
- [2] Ch. M. Bishop, Pattern Recognition and Machine Learning, Springer Science + Business Media, LLC 2006.
- [3] W. Findeisen, J. Szymanowski, A. Wierzbiicki, Teoria i metody obliczeniowe optymalizacji. Pastwowe Wydawnictwo Naukowe, Warsaw 1977.
- [4] 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.
- [5] Joarder Kamruzzaman, Rezaul Begg, Artificial Neural Network in Finance and Manufacturing, Idea Group Publishing, Hershey, Pennsylvania 2006.
- [6] O. B. Lapunow, On possibility of circuit synthesis of diverse elements, Mathematical Institute of B.A. Steklova 1958.
- [7] Toshinori Munakate, Fundamentals of the New Artificial Intelligence, Second Edition, Springer 2008.
- [8] S. Osowski, Sieci neuronowe do przetwarzania informacji, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 2006.
- [9] S. Placzek, B. Adhikari, "Analysis of Multilayer Neural Network with Direct Connection Cross-forward Connection", CS&P Conference 2013, The University of Warsaw, Warsaw 2013.
- [10] L. Rutkowski, Metody i techniki sztucznej inteligencji, Wydawnictwo Naukowe PWN, Warszawa 2006.