Coordinator Synthesis for Hierarchical Structure of Artificial Neural Network

Stanislaw Placzek

Engineering Department
Vistula University
Stoklosy Street, 02-787 Warsaw, Poland
stanislaw.placzek@wp.pl

Abstract. Two important issues have to be dealt with when implementing the hierarchical structure [1] of the learning algorithm of an Artificial Neural Network (ANN). The first one concerns the selection of the general coordination principle. Three different principles are described. They vary with regard to the degree of freedom for first-level tasks. The second issue concerns the coordinator structure or coordination algorithm. The ANN learning process can be examined as a two-level optimization problem. Importantly all problems and sub-problems are unstructured minimization tasks . The article concentrates on the issue of the coordinator structure. Using the interaction prediction principle as the most suitable principle for two-level ANN structures, different coordinator target functions are defined. Using classification task examples, the main dynamic characteristics of the learning process quality are shown and analyzed.

Keywords: Artificial Neural Network (ANN), hierarchy, decomposition, coordination, coordination principle, coordinator structure

1 Computational task complexity

Large-scale multidimensional classification, interpolation and extrapolation are complex tasks that can require long calculation times. For these tasks one can make use of the ANN learning process using input and output data vectors with a defined network architecture. There is no theoretical solution to the architecture selection process, including the definition of the number of hidden layers and neuron distribution between layers. From a calculation point of view, the ANN learning process approaches a local or a global minimum asymptotically and is very time-consuming. A multi-layered ANN with one input layer, a set of hidden layers and an output layer can be sectioned off [1]. Every layer has its own input and output vectors. For a standard two-layer network both the hidden layer and the output layer can be described as sub-networks. These sub-networks form the first-level of the hierarchical structure. So the network consists of two sub-networks, and the local target function for each of them is defined $\Phi = (\Phi_1, \Phi_2)$.

Similarly to the ANN structure decomposition, learning algorithm using error backpropagation can also be decomposed. It can be decomposed into:

- The first-level task, searching for the minimum of the local target functions $\Phi = (\Phi_1, \Phi_2)$,
- The second-level task, coordinating the first-level tasks.

Unfortunately, the first level optimization tasks in such learning algorithm are non-linear. In practice, only standard procedures exist to solve these optimization problems . But in a two-level learning algorithm structure, the coordinator is not responsible for solving the global task. In [2], the most popular interaction prediction principle was implemented. According to this principle, the coordinator plays an active role in the current ANN learning process. The main interaction prediction principle is shown in Fig. 1. With each iteration, the coor-

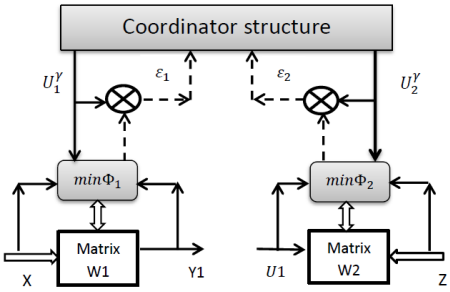

Fig. 1. Interaction Prediction Principle

dinator and all of the first-level sub-tasks interchange information. The first-level sub-tasks are optimal searching tasks. Usually, they look for the minimum of the Mean Squared Error (MSE). The coordination algorithm structure is primary related to the interaction prediction principle and two signals are used: primary discerning $U^\gamma = (U_1^\gamma, U_2^\gamma)$ and the feedback signal ascending $\epsilon = (\epsilon_1, \epsilon_2)$. The primary signals are known as coordination signals and are sent from the coordinator to all of the first-level sub-tasks. Thereby the coordinator assists in optimizing the first-level sub-tasks. The forecast coordination signals may not be precisely correct and the sub-tasks calculate their own value of the coordination signals. These signals are sent up into the coordinator. The coordinator then uses its own gradient method to calculate the new value of the coordination signals (improving upon its own previous estimate). The process can be continued until the coordinator and all the first-level sub-tasks have solved their own tasks.

1.1 Coordination aspects

Coordination as an operation process is related to three types of decision problems [3]. Finding a solution for:

– The global task as the primary task for the entire ANN structure,
– Local tasks as the decomposition's result of the global target function,
– Coordinator task that should be synthesized or find a solution procedure.

In effect , three different specific problems have to be solved:

1. Coordinator Synthesis. Decompose the global target function Φ into two sub-tasks with their own local target functions Φ_1, Φ_2; the upper-level coordinator task should be defined in such a way that all the sub-tasks are coordinated.
2. Modification Problem. In a two-level ANN learning algorithm; both the first-level tasks and the coordinator task are defined. Unfortunately, the ANN is not coordinated by the coordinator tasks. Before this problem can be dealt with, one should modify the first-level tasks in such a way that the coordinator coordinates the modified local target functions $\Phi = (\Phi_1, \Phi_2)$. This can be formalized using the following predicate formula:

$$(\exists\gamma)(\exists W)[\mathbf{P}(W, \Phi) and \mathbf{P}(U, \Psi(U^{\gamma}, \epsilon))] \tag{1}$$

   Where: The predicate $\mathbf{P}(W, \Phi)$ is true, if Φ is a problem (task) and W is one of its solution. $\Psi(U, \epsilon)$- coordinator target function. The first-level tasks are coordinated with the coordination task when the coordination task has a solution, all the first-level sub-tasks also have the solution for same coordination input U^{γ}.
3. Decomposition. Given only the global target task Φ for ANN - decompose this task Φ into the two sub-tasks Φ_1, Φ_2 and find a coordinator structure and a coordination procedure. This is formalized as:

$$(\exists\gamma)(\exists W)[\mathbf{P}[W = (W_1, W_2), (\Phi_1(U), \Phi_2(U) and \mathbf{P}(W, \Psi(X, Z, W))]] \tag{2}$$

   Where: X - input file, Z - learning file, $W = (W_1, W_2)$ - ANN's weight coefficients.

The first-level tasks are coordinated with a given global target task when the global task has a solution and as do some of the first-level tasks. The coordinator has to influence the first-level tasks in such a way that the resulting action guarantees the solution of the global target task.

1.2 Decomposition and coordination

Using Fig. 1, one can define the set of target functions:

– The global target function:

$$\Phi(W1, W2, Y, Z) = \frac{1}{2} \cdot \sum_{k=1}^{N_2} (y_k - z_k)^2 \tag{3}$$

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

– The local target function Φ_1

$$\Phi_1(W1, X, U^\gamma) = \frac{1}{2}\sum_{i=1}^{N_1} f(\sum_{j=0}^{N_0} W1_{ij} \cdot x_j) - u_i^\gamma)^2 \tag{4}$$

Where: $U^\gamma[1 : N_1]$ - the coordination matrix as an input variable, N_1 - the number of hidden neurons, N_0 - the number of input neurons, $f(*)$ - a sigmoid function.

– The local target function Φ_2.

$$\Phi_2(W2, Z, U^\gamma) = \frac{1}{2}\sum_{k=1}^{N_2} f(\sum_{i=0}^{N_1} W2_{ki} \cdot u_i^\gamma) - z_k)^2 \tag{5}$$

Where: $U^\gamma[1 : N_1]$ - the coordination matrix as an input variable, N_2 - the number of output neurons, $f(*)$ - a sigmoid function

Using (4), one can calculate the feedback signal $\epsilon 1_i$ and the new value of matrix W1.

$$\epsilon 1_i = f(\sum_{j=0}^{N^0} W1_{ij}\dot{x}_j) \tag{6}$$

$$\frac{\partial \Phi 1}{\partial W1_{ij}} = (v1_i - u_i^\gamma) \cdot f' \cdot x_j \tag{7}$$

$$W1_{ij}(n+1) = W1_{ij}(n) - \alpha 1 \cdot \frac{\partial \Phi 1}{\partial W1_{ij}} \tag{8}$$

For the second sub-network using (5), one can calculate the new value of $\epsilon 2_i$ and the new value of matrix $W2_{ki}$.

$$\frac{\partial \Phi 2}{\partial W1_{ki}} = (v2_k - z_k) \cdot f' \cdot u_k^\gamma \tag{9}$$

$$W2_{ki}(n+1) = W2_{ki}(n) - \alpha 2 \cdot \frac{\partial \Phi 2}{\partial W1_{ki}} \tag{10}$$

$$\frac{\partial \Phi 2}{\partial u_i^\gamma} = \sum_{k=1}^{N_1}(v2_k - z_k) \cdot f' \cdot W2_{ki} \tag{11}$$

$$\epsilon 2_i(n+1) = u_i^\gamma(n) - \alpha 3 \cdot \frac{\partial \Phi 2}{\partial u_i^\gamma} \tag{12}$$

Where: $\alpha 1, \alpha 2, \alpha 3$ - learning coefficients, $u_i^\gamma = u1_i^\gamma = u2_i^\gamma$ - coordination signals, $\epsilon 1_i, \epsilon 2_i$ - feedback signals.

To summarize the first-level includes two sub-networks and two optimization tasks. The first sub-network calculates the new coefficient matrix $W1_{ik}(n + 1)$ and the feedback signal $\epsilon 1_i$ value by taking the parameter $u1_i^\gamma$ from the coordinator and using the optimization procedure. The feedback signal is sent

into the coordinator. For the second sub-network, the coordination signal $u2_i^{\gamma}$ sets the optimization procedure in motion and calculates the new coefficient matrix $W2_{ki}(n+1)$ and the feedback signal $\epsilon 2_i$ value. The feedback signal is also sent into the coordinator. After that, the coordinator procedure has to calculate the new coordinator signal $u1_i^{\gamma}(n+1)$ *and* $u2_i^{\gamma}(n+1)$. Thus, design of the coordinator structure is the main problem in a two-level learning algorithm.

2 Coordinator structure

In a two-level learning algorithm, the coordinator plays the main role. Therefore, the choice of the coordinator principle is paramount. The principle should specify strategies for the coordinator and determines the structure of the coordinator. Three different approaches to how the interaction could be performed were introduced [3].For an ANN learning algorithm, the Interaction Prediction Principle is the most suitable. According to it, the coordinator predicts the interface inputs. Success in coordinating all the first-level tasks depends on the accuracy of the prediction of the interface inputs or the effect of the prediction inputs. Therefore, to obtain responses from the first-level tasks, the coordinator could measure:

1. The interface inputs value. This principle will be known as the Interface Interaction Balance.
2. The target function value. This principle will be known as the Performance Prediction Principle.

2.1 Interface Interaction Balance

The coordination input may involve a prediction of the interface between the first and the second sub-networks. For the first sub-network, the coordinator signal U_1^{γ} is sent into the target function Φ_1 as a teacher data parameter. For the second sub-network, the coordinator signal U_2^{γ} is treated as an input variable. Thanks to it, both tasks are fully specified and algorithms can find the minimum value of their target functions Φ_1, Φ_2 . The Interface Interaction Balance idea is shown in Fig. 2. The coordinator target function Ψ could be a linear or a nonlinear

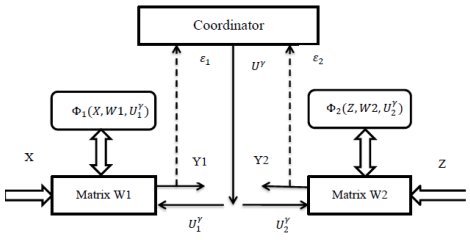

Fig. 2. Coordination algorithm structure for Interface Prediction Principle

function.

$$\Psi(U^\gamma, \epsilon^\gamma) = \Psi(U^\gamma, U^\gamma - Y) \tag{13}$$

Equation (13) uses the linear relation between two feedback signals $\epsilon 1, \epsilon 2$ and the prediction interface value U^γ. Where: U^γ- prediction interface value Y - real interface value

$$\Psi = \frac{1}{2}\sum_{i=1}^{N_1}(u_i^\gamma - \epsilon 1_i^\gamma)^2 + \frac{1}{2}\sum_{i=1}^{N_1}(u_i^\gamma - \epsilon 2_i^\gamma)^2 \tag{14}$$

Where: $U^\gamma = [u_1^\gamma, u_2^\gamma ... u_{N_1}^\gamma]$

Using formula (14), the first derivative is calculated

$$\frac{\partial \Psi}{\partial u_i^\gamma} = (u_i^\gamma - \epsilon 1_i^\gamma) + (u_i^\gamma - \epsilon 2_i^\gamma) \tag{15}$$

$$u_i^\gamma(n+1) = u_i^\gamma(n) - \lambda 1 \cdot \frac{\partial \Psi}{\partial u_i^\gamma} \tag{16}$$

Where: $\lambda 1$ - learning coefficient,
The coordinator and the first-level sub-networks work in an iterative scheme. When the coordinator signal $U^\gamma(n)$ is applied and the first-level optimization tasks find their own solution, a new coordination signal can be calculated (16). Using the Interface Interaction Balance, the two vectors signals are measured: the predicted interface input U^γ and the real interface value Y_1, Y_2 (Fig.2). In a real situation this requirement could be difficult to implement.It is usually possible to measure the first-level tasks and to send into the coordinator the target function (performance) Φ_1, Φ_2 value and their derivatives.

2.2 Linear Performance Balance Principle

The target functions values and their derivatives are of a more generalized form.The coordination scheme that uses this idea is shown in Fig. 3. The coor-

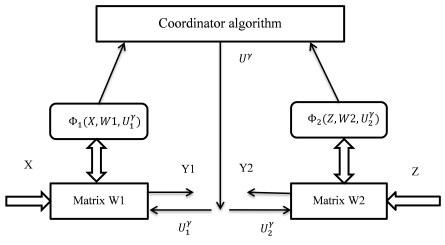

Fig. 3. Coordination algorithm structure for Performance Prediction Principle

dinator can receive both the target function value and the derivative value from

the first-level sub-systems . The coordinator function can be defined as a linear or a nonlinear function of the two parameters $\Phi 1, \Phi 2$

$$\Psi = \Psi(\Phi 1, \Phi 2) \tag{17}$$

Then the local target functions can be defined by :

$$\Phi 1(Y1, W1, U^{\gamma}) = \frac{1}{2}\sum_{i=1}^{N_1}(y1_i - u_i^{\gamma})^2 \tag{18}$$

$$\Phi 2(Y2, W2, Z) = \frac{1}{2}\sum_{k=1}^{N_2}(y2_k - z_k)^2 \tag{19}$$

In this subsection, we examine the coordinator function (17) as linear relation:

$$\Psi = \Phi 1 + \Phi 2 \tag{20}$$

Using (18) and (19) the first partial derivatives are calculated:

$$\frac{\partial \Phi 1}{\partial u_i^{\gamma}} = (y1_i - u_i^{\gamma}) \tag{21}$$

$$\frac{\partial \Phi 2}{\partial u_i^{\gamma}} = \sum_{k=1}^{N_2}(y2_k - z_k) \cdot f' \cdot W2_{ki} \tag{22}$$

Using the same gradient algorithm for the coordinator as above, the new coordination signal $U^{\gamma}(n+1)$ is calculated using the formula

$$u_i^{\gamma}(n+1) = u_i^{\gamma}(n) - \lambda 1 \cdot \frac{\partial \Psi}{\partial u_i^{\gamma}} \tag{23}$$

The global target function Ψ is a nonlinear function that should take into account the sigmoid activation functions for both the first and the second sub-network. Because of this, one can say that the coordination function described by formula (20) is simplistic and does not consider the non - linearity of the coordinator structure.

2.3 Nonlinear Performance Balance Principle

In the article, non - linearity will be demonstrated by two coordinator target functions

$$\Psi_m = \Phi 1 + \Phi 2 + c \cdot \Phi 1 \cdot \Phi 2 \tag{24}$$

$$\Psi_p = \Phi 1 + \Phi 2 + c \cdot (\Phi 1^2 + \Phi 2^2) \tag{25}$$

Where:
Ψ_m - indicates non-linear multiplication part,
Ψ_p - indicates non-linear sum of power part.

parameter "c" describes the impact of the non-linear part on the coordinator algorithm.

The structure of target functions Φ_1, Φ_2 are the same as in formula (18) and (19), respectively. The final formulas for derivatives have more complicated structures

$$\frac{\partial \Psi_m}{\partial u_i^\gamma} = \frac{\partial \Phi 1}{\partial u_i^\gamma} + \frac{\partial \Phi 2}{\partial u_i^\gamma} + c \cdot \left[\frac{\partial \Phi 1}{\partial u_i^\gamma} \cdot \Phi 2 + \frac{\partial \Phi 2}{\partial u_i^\gamma} \cdot \Phi 1 \right] \tag{26}$$

$$\frac{\partial \Psi_p}{\partial u_i^\gamma} = \frac{\partial \Phi 1}{\partial u_i^\gamma} + \frac{\partial \Phi 2}{\partial u_i^\gamma} + 2 \cdot c \cdot \left[\frac{\partial \Phi 1}{\partial u_i^\gamma} \cdot \Phi 1 + \frac{\partial \Phi 2}{\partial u_i^\gamma} \cdot \Phi 2 \right] \tag{27}$$

In practice, calculation developers are obliged to find the optimal "c" value. This parameter will have a significant impact on the quality and stability of the coordinator learning process.

3 Classification task example

The main dynamic characteristics of the learning process can be shown using the following example. Emphasis is put on the characteristics of the first-level local target functions, Φ_1, Φ_2, and the second level, coordinator target function Ψ . Optimal ANN learning characteristics depend on two connected tasks. Firstly, a network structure that includes a number of hidden layers needs to be created.Secondly, neurons need to be distributed between layers. A single hidden layer is chosen in this example. The structure of the ANN is simple and can be described as ANN $(N_0 - N_1 - N_2)$. In literature, one can only find suggestions regarding optimal numbers of neurons using the Vapnik - Cervonenkis dimension [4][5]. However, the hidden layer structure could also be determined using Kolmogorov's theorem [4][6]. For an ANN with one hidden layer, a sigmoid activation function can be chosen for classification tasks and N_0 input neurons,

$$VCdim = N_0 + 1 \tag{28}$$

Therefore, we can use this measure to define the number of neurons in the hidden layers

$$N_1 = VCdim \tag{29}$$

For a continuous function with N_0 input vector dimension and N_2 output vector, the number of neurons according to Kolmogorov's theorem can be calculated as

$$N_1 = 2 \cdot N_0 + 1 \tag{30}$$

In practice, the number of neurons in the hidden layer will be chosen according to the formula

$$N_0 + 1 \leq N_1 \leq 2 \cdot N_0 + 1 \tag{31}$$

Sigmoid activation functions are implemented in both the hidden and output layers. For the classification task using 6 - dimension input vectors $N_0 = 6$ and $N_2 = 1$, the number of hidden neurons is calculated according to formula

$$7 \leq N_1 \leq 13 \tag{32}$$

3.1 Example for Interface Interaction Balance

The quality of the leaning process depends on the key learning parameters for the first-level as well as the coordinator level. Fig. 4 shows the dynamic characteristics of the learning process.

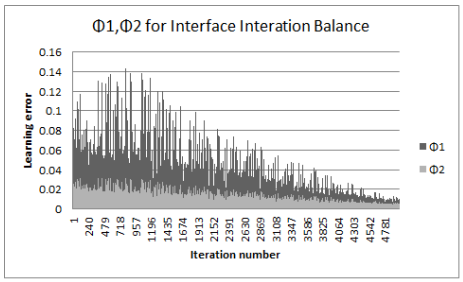

Fig. 4. The first sub-networks learning error. $\lambda_1 = 0.5$

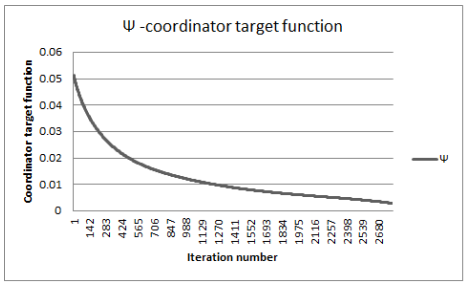

Fig. 5. The coordinator learning error. $\lambda = 0.2$

The main learning parameters, $\alpha 1 = 0.3, \alpha 2 = 0.3$, guarantee that every sub-task can find its own minimum target function value. The coordinator has its own algorithm described by equation (23). If parameter λ_1 is too large, the learning process is not stable, especially at the end of the iterative process.

The first sub-network includes 6 input neurons and 13 output neurons. Matrix W1 includes 7*13 = 91 neurons, but matrix W2 only 14*1=14 neurons. At each stage the first sub-network calculates a lower target function $\Phi 1$ as the second sub-network. In the middle and the final part of the iterative process, the characteristics are the same. This means that the sub-networks achieved only small corrections to their matrix coefficients and the standard gradient algorithm is not efficient (Fig.6).

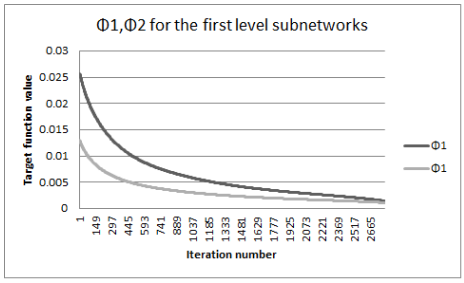

Fig. 6. Learning error with regards to the iteration number. The learning parameter $\lambda_1 = 0.2$

3.2 Linear Performance Balance Principle

As stated above, the performance prediction principle is more general that interface prediction. The coordinator gets information not from all of the elements of the output vectors, Y_1, Y_2, as in the previous algorithm, but only some general information, such as the function value Φ_1, Φ_2 and their derivatives.

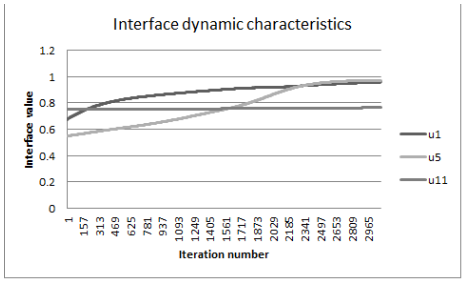

Fig. 7. Dynamic characteristics of the hidden interface value

In Fig. 6, the quality of the learning process is shown . Error functions decrease their value quite fast and the oscillation does not exist. On Fig.7. dynamic characteristics of interface value u_1, u_5, u_{11} are shown. Part of them change their value dynamically.

3.3 Nonlinear Performance Balance Principle

The sub-network transfer function is nonlinear. The coordinator structure should consider this and use a more complicated target function structure. According to formulas (24)(26), the function Ψ includes a nonlinear part: the multiplication or sum of the second power target functions. In Fig. 8 the quality of the learning process is shown. A flex point for all of the sub-networks characterizes this

learning process. After that, the learning process achieves the minimum value in an asymptotic way. Learning process quality is different for $\lambda_1 - 0.2$. It is stable and smooth (Fig.9). The coordinator algorithm requires two parameters: the learning coefficient λ_1 and the parameter c. The learning process quality depends on the "c"-value. Fig. 10 shows this characteristic, including the iteration number for the flex point and the total iteration number.

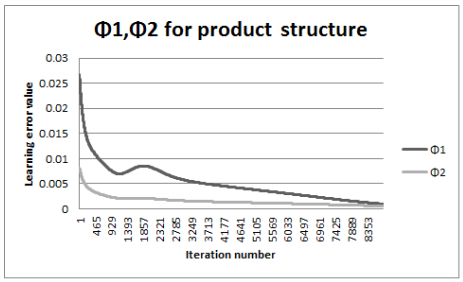

Fig. 8. The iteration number for nonlinear coordination function. $\lambda_1 = 0.3$

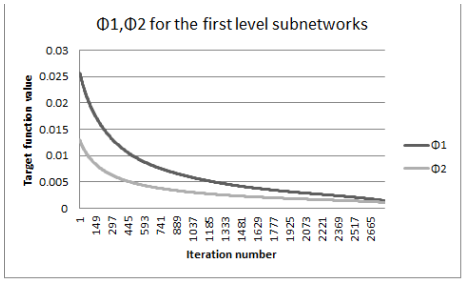

Fig. 9. The iteration number for nonlinear coordination function. $\lambda_1 = 0.2$

4 Conclusion

In this article, only a single principle was examined. The interaction balance principle can be realized by measuring two different feedback signals: the sub-network output signals Y_1, Y_2 or the sub-network performance value Φ_1, Φ_2. In the first case, the learning process is very flexible for the coordinator learning parameter λ. For $\lambda = 0.3$, the last learning process stage includes oscillation, which delays the process of convergence. For $\lambda = 0.2$ the characteristics are better. Learning processes are not simple and depend not only on an ANN structure

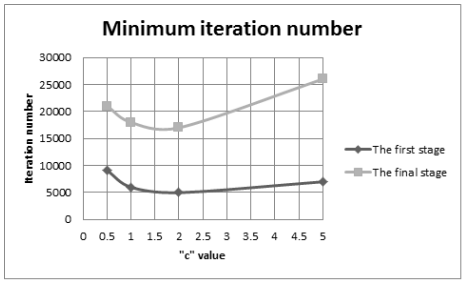

Fig. 10. Optimal parameters for Nonlinear Performance Balance Principle

but on the input data structure as well. To measure sub-network performance the coordinator can receive more general information. This has a positive impact on all the dynamic characteristics (Fig.6). In particular the oscillation seen in the middle of the learning process are smaller. From the coordinator point of view, every sub-network can be seen as a black box with the input signal U^{γ} and output Y_1 or Y_2. Response functions have to be nonlinear because the global target function is also nonlinear. The coordinator, when using a stable coordinator algorithm with constant learning parameters is not responsible for finding the optimal interaction vector U^{γ} during the entire learning process (from beginning to end). This has a negative impact on the quality and speed of convergence. Using the hierarchical principle, an additional level can be added: the adaptation level. The adaptation level, using its own identification algorithm, can predict learning parameters as λ and c. This theme should be studied in future. Finally, the nonlinear coordinator algorithm was examined. In that case all algorithms are very flexible with regarded to parameter c", which decides about the impact of the nonlinear part of the coordinator algorithm for convergence. With a small value, learning time is very long, but for a large value, oscillations are seen. The characteristic of $n = f(c)$, where: n- iteration number is shown in Fig. 10.

References

1. Placzek Stanislaw. A two-level on-line learning algorithm of Artificial Neural Network with forward connections IJARAI, vol.3, no. 12, 2014
2. Placzek Stanislaw. Decomposition and the principle of interaction prediction in hierarchical structure of learning algorithm of ANN Poznan University of Technology, Academic Journal. Electrical Engineering, no 84, Poznan 2015
3. Mesarovic M.D., Macka D., Takahara Y. Theory of hierarchical multilevel systems, Academic Press, New York and London 1970
4. Haykin S. Neural network, a comprehensive foundation. Macmillan College Publishing Company, New York 1994.
5. Vapnik V. Statistical Learning Theory Wiley, New York 1998
6. Osowski S. Sieci neuronowe w ujeciu algorytmicznym WNT Warszawa 1996