UDC 004.8 (045)

[1]O. I. Chumachenko,
[2]I. V. Kryvenko

NEURAL NETWORKS MODULE LEARNING

[1]Technical Cybernetic Department National Technical University of Ukraine
"Kyiv polytechnic institute", Kyiv, Ukraine
[2]Aviation Computer-Integrated Complexes Department, National Aviation University, Kyiv, Ukraine
E-mails: [1]chumachenko@tk.kpi.ua, [2]prostoigor1995@bigmir.net

Abstract—*Currently, there exists a huge number of neural networks of different classes, each with its own advantages and disadvantage. However, there aren't a lot of focus on hybrid neural networks, based on the combination of knowm topologies of neural networks. Modular organization principle seems to be very promising, however principles of its module creation isn't known and needs further research. The present study, therefore, proposes some methods of hybrid neural network module creation and their learning algorithms.*

Index terms—Artificial intelligence; connectionist models; bidirectional associative memory.

I. INTRODUCTION

Main element of artificial intelligence system is an artificial neural network – a mathematical model and its hardware or software execution, based on the principle if organizing and functioning of biological neural networks (nerve cells of a living organism). This concept originated in the study of the processes, occurring in the brain and during tries to simulate there processes. Like human beings, artificial neural networks can discriminate, identify, and categorize perceptual patterns. Hybrid neural system is a system, which uses more than one method of simulating human intellectual activity. So, in this paper we propose a modular approach to the organization of hybrid neural networks, according to which the elements of created topology are modules, whose structure includes neural networks of different topologies: a self-organizing network, the basic neural network (it's topology is chosen, basing on the required task) and bidirectional associative memory (Kosco's neural network). It's structural scheme is presented on Fig. 1.

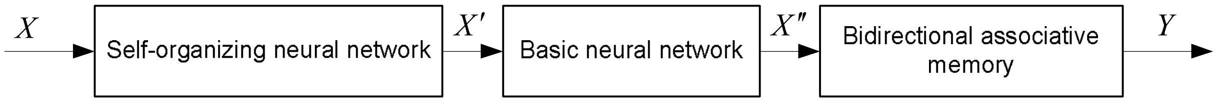

Fig. 1. Structural scheme of neural network module

In the [1], learning algorithm of hybrid neural network module is proposed, which includes following learning methods:

– Learning self-organizing neural network by the reference sample X and bidirectional associative memory by output reference sample Y.

– On the basis of tuned self-organizing neural network and bidirectional associative memory we determine reference output sample X.' by reference input sample X (reference input sample base neural network) and the input reference sample X." by the output reference sample Y (reference sample of the base neural network).

– Learning basic neural network by the reference input and output samples X.' and X.".

Self-organizing neural network learning isn't very challenging. In this work bidirectional associative memory learning, included in the module is described. This single-layer feedback neural network is based on two ideas: the adaptive resonance theory of Stephen Grosberg and auto-associative Hopfield memory.

Bidirectional associative memory (BAM) is a heteroassociative: input vector follows with a set of neurons and the corresponding output vector is generated as a set of different neurons. As a Hopfield network, BAM is able to generalize, emitting correct responses, despite the distorted input signals. In addition, adaptive versions of BAM can be realizes, which can distinguish a master image from a set of noisy signals. This features are closely reminiscent of the process of human thinking and they allow artificial neural networks to make a step towards the real human brain simulation.

This work shows the possibilities of BAM network learning with the introduction of a recency parameter, which gives the network huge gains in performance, compared to regular networks at a price of lowered memory storage limit. Also, with this inclusion the network requires less iterations.

II. Task Statement

Bidirectional associative memory structural scheme is shown in Fig. 2 where $x(0)$ and $y(0)$ represent input and output signals, W and V are the weight coefficients and t is the current iteration number [2].

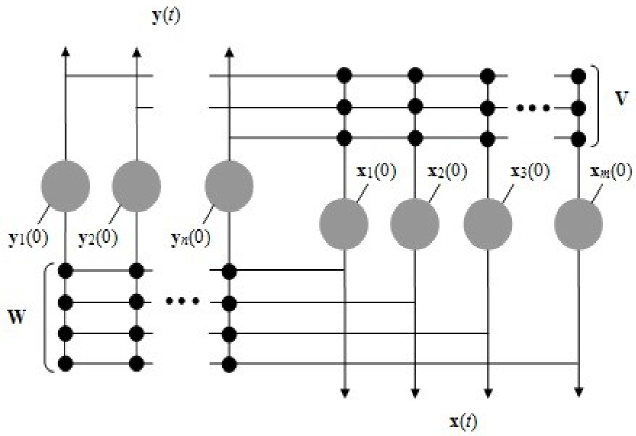

Fig. 2. Bidirectional associative memory structural scheme

This network type was developed by Chartier & Boukadoum and its unique feature is a different matrix set for each neuron layer, so it is possible for it to correlate different types of input signal patterns, such as bipolar or real-valued ones. The structure is a set of two Hopfield neural networks joined beginning-to-end style, which grants the information flow to both sides without interference, so it can be learned in both ways: from both input and output signal sets [3]

The task of this work is to learn network with a set of reference output signals and to determine weight coefficients V to determine a set of reference input signals and to determine network weight coefficients W backward, by the set of reference input signals.

III. Activation Function

The activation function is based on the classic Verhulst equation extended to a cubic form with a saturating limit at ± 1:

$$\forall i,\dots,N,\ y_{i[t+1]} = f\left(a_{i(t)}\right),$$

$$= \begin{cases} 1, & \text{if} \quad a_{i(t)} > 1, \\ -1, & \text{if} \quad a_{i(t)} < -1, \\ (\delta+1)a_{i(t)} - \delta a_{i(t)}^3, & \text{else} \end{cases}$$

and

$$\forall i,\dots,M,\ x_{i[t+1]} = f\left(b_{i(t)}\right),$$

$$= \begin{cases} 1, & \text{if} \quad b_{i(t)} > 1, \\ -1, & \text{if} \quad b_{i(t)} < -1, \\ (\delta+1)b_{i(t)} - \delta b_{i(t)}^3, & \text{else,} \end{cases}$$

where N and M are the number of units in each layer, i is the unit index, δ is a general transmission parameter and a and b are the activations. Figure 3 illustrates the shape of the transmission function for $\delta = 0.2$.

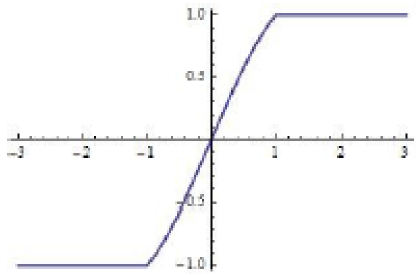

Fig. 3. Transmission function for $\delta = 0.2$

There are no asymptotic behaviors in this type of activation function, meaning no false roots. Also, this function has the advantage of exhibiting grey-level attractor behavior [4].

IV. Learning Rule

There exists a possibility to modify the learning rule, using a Hebbian/anti-Hebbian approach:

$$\begin{aligned} W(k+1) &= W(k) + \eta\left(y(0) - y(t)\right)\left(x(0) + x(t)\right)^{\mathrm{T}}; \\ V(k+1) &= V(k) + \eta\left(x(0) - x(t)\right)\left(y(0) + y(t)\right)^{\mathrm{T}}, \end{aligned} \tag{1}$$

where η is the learning parameter controlling for the speed of convergence and k is the learning trial number. Connection weights are initiated at 0 and $x(0)$ and $y(0)$ are the initial inputs to be associated. The network has converged when $x(0) = x(t)$ or $y(0) = y(t)$. Thus, each weight matrix converges when the feedback is equal to the initial inputs. In the BAM, the network convergence is guaranteed if the learning parameter η is set according the following condition:

$$\eta < \frac{1}{2(1-2\delta)\max[M,N]}, \qquad \delta \neq \frac{1}{2}, \tag{2}$$

where M and N are respectively the dimensionality of the input and its association. The η parameter was set to a lower value than the threshold found in (2) for every simulation performed. The learning rule (1) acts much like a long-term memory where the learning convergence is longer, but exhibits an increased storage capacity and has a better-defined attractor.

V. Learning Rule Modification

In order to lower the time to learn associations, the memory capacity has to be decreased [2]. One way to accomplish this is by introducing a recency parameter $(0 \le \beta \le 1)$. This parameter removes from the memory associations that are not reinforced enough. The resulting learning rule after modification is given by:

$$W(k+1) = \beta W(k) + \eta\big(y(0) - y(t)\big)\big(x(0) + x(t)\big)^{\mathrm{T}},$$
$$V(k+1) = \beta V(k) + \eta\big(x(0) - x(t)\big)\big(y(0) + y(t)\big)^{\mathrm{T}}, \quad (3)$$

If $\beta = 1$ then the learning is accomplished in the same fashion as in equation (1). This learning rule can be simplified to the following hebbian/anti-hebbian equation in the case of auto association where $y(0) = x(0)$:

$$W(k+1) = \beta W(k) + \eta\big(x(0)x(0)^{\mathrm{T}} - x(t)x(t)^{\mathrm{T}}\big)^{\mathrm{T}},$$
$$V(k+1) = \beta V(k) + \eta\big(y(0)y(0)^{\mathrm{T}} - y(t)y(t)^{\mathrm{T}}\big)^{\mathrm{T}}. \quad (4)$$

VI. Methodology

Learning was carried out according to the following procedure:

1) Random selection of a pair of patterns ($x(0)$ and $y(0)$).

2) Computation of $x(t)$ and $y(t)$ according to the transmission function (1).

3) Computation of the weight matrices update according to (3).

4) Repetition of steps 1) to 3) until all of the pairs have been presented.

5) Repetition of steps 1) to 4) for an a-priori set number of epochs.

The transmission parameter (δ) was set to 0.2 throughout the simulations and the number of iterations to perform by the network before the weight matrices were updated was set to = 1. The network was tested on an auto-association and hetero-association task that consisted of 26 stimuli placed on 7x7 grids (Fig. 4). The auto-association task was an association of uppercase stimuli only, whereas the hetero-association consisted of the association between uppercase and lower case stimuli. The recency parameter (β) was set to 0.99 and 0.995 for the rapid setting and at 1.0 for the standard long-term setting. In the rapid setting, instead of presenting all the patterns at once, the network was limited to only one subset at a given time. In other words, rather than learning all stimuli in one epoch, the network limited itself to grouped associations of a maximum of 5 associations.

Following the learning phase, the network was tested on a recall task according to the following procedure:

1) Selection of an input pattern $x(0)$.

2) Computation of $y(1)$ according to the transmission function (1).

3) Comparison with the target value $y(0)$

4) Repetition of steps 1) to 3) until all of the patterns have been presented.

In this situation a given pattern iterated until a steady state. Recall performance was recorded for the level of flipped pixels varying from 0 to 24 (0 to ≈ 50%). The network was tested on grouped associations of 2, 3, 4 and 5 patterns. The network was tested 200 times for every condition and the average performance was computed [2].

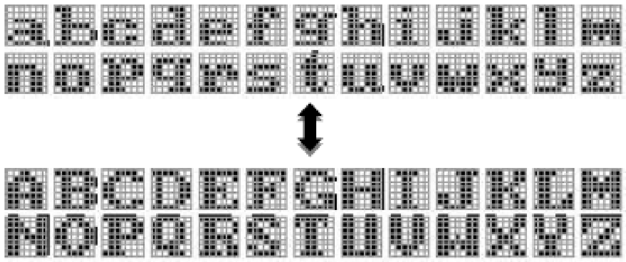

Fig. 4. Patterns used for the simulation

VII. Results

Figure 5 presents an example of the first 10 patterns recalled in a noiseless (0 flipped pixel) situation for both auto-association and hetero-association tasks. The orange dashed lines represent the demarcation between previously learned associations and the associations that have just been learned. The model was compared to the results of Hopfield's model (1982) as well as Kosko's (1988). For both networks, contrary to the BAM, there are no memory traces between the past and current association. In other words, the connection weights are reset to zero between the learning of a given group. The connection weights had to be set to zero since both Hopfield and Kosko's model cannot perform the task otherwise as they suffer from memory overload. It is as if we are comparing the performance of a single BAM with several independent Hopfield or Kosko models. Although this situation is different, it was included for comparison purposes using optimal conditions for Hopfield and Kosko.

The results (Fig. 5) for the auto-association of the short-term memory show that previously learned associations tend to be erased as new associations are made, particularly when the correlation is very high between two patterns (for example, the stimulus E and F). When patterns are presented in groups of two, the shortterm network makes no mistakes in associating the patterns presented within the step; this also holds for conditions where patterns are presented in groups of 5. The Hopfield network shows perfect performance when the input patterns are learned in groups of two. However, when presented in groups of 5, the network makes several mistakes even in the absence of noise. These results are even more disastrous for hetero-associations, where the network can barely recall any associations. Hence, Kosko's network is not able to learn any of the associations grouped in pairs, whereas the short-term BAM is able to learn all associations whether they are presented in groups of 2 or 5. Results for the standard BAM were not shown because it could learn and recall perfectly in all of the previous situations.

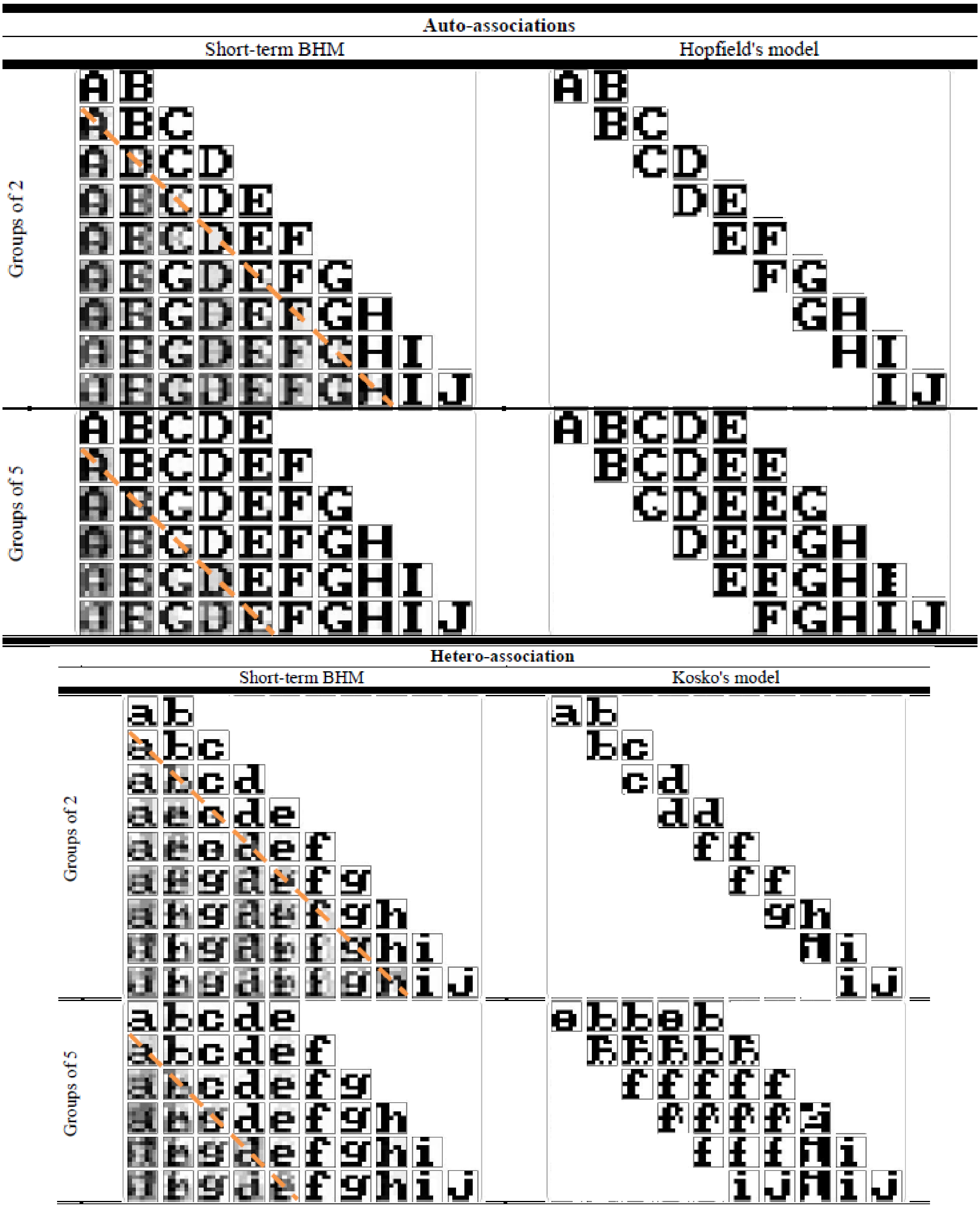

Fig. 5. Association recall for auto and hetero-association learning

VIII. Conclusions

In this paper we have shown the structure of BAM, the algorithms to learn the network both by input and output nodes or to modify the learning rule to lessen the required learning time.

References

[1] D. Yu. Koval, G. A. Sipakov, and D. D. Shevchuk, "Using ANFIS and NEFCLASS neural networks in classification problems". *Electronics and Control Systems*, no. 1(43), Kyiv, NAU, pp. 93–98, 2015.

[2] C. Tremblay, N. Berberian, and S. Chartier, Didirectional Associative Memory for Short-term Memory Learning, COGSCI, 2014, pp.1–6.

[3] S. Chartier, and M. Boukadoum, "Encoding static and temporal patterns with a bidirectional heteroassociative memory". *Journal of Applied Mathematics*, pp. 1–34, 2011.

[4] S. Chartier, and M. Boukadoum, "A bidirectional heteroassociative memory for binary and grey-level patterns". *IEEE Transactions on Neural Networks*, vol. 17(2), 2006, pp. 385–396.

Received January 28, 2016

Chumachenko Olena. Candidate of engineering. Associate Professor.
Technical Cybernetic Department, National Technical University of Ukraine "Kyiv Polytechnic Institute", Kyiv, Ukraine.
Education: Georgian Politechnic Institute, Tbilisi, Georgia, (1980).
Research area: system analysis, artificial neuron nets.
Publications: more than 50 papers.
E-mail: chumachenko@tk.kpi.ua

Kryvenko Igor. Bachelor.
Aviation Computer-Integrated Complexes Department, National Aviation University, Kyiv, Ukraine.
Education: National Aviation University, Kyiv, Ukraine, (2016).
E-mail: prostoigor1995@bigmir.net

О. І. Чумаченко, І. В. Кривенко. Навчання модулів нейронної мережі

Розглянуто систему гібридних нейронних мереж. Запропоновано методи навчання їх модулів.

Ключові слова: штучний інтелект; зв'язкові моделі; двонаправлена асоціативна пам'ять.

Чумаченко Олена Іллівна. Кандидат технічних наук. Доцент.
Кафедра технічної кібернетики, Національний технічний університет України «Київський політехнічний інститут», Київ, Україна.
Освіта: Грузинський політехнічний інститут, Тбілісі, Грузія, (1980).
Напрям наукової діяльності: системний аналіз, штучні нейронні мережі.
Кількість публікацій: більше 50 наукових робіт.
E-mail: chumachenko@tk.kpi.ua

Кривенко Ігор Вікторович. Бакалавр.
Кафедра авіаційних комп'ютерно-інтегрованих комплексів, Національний авіаційний університет, Київ, Україна.
Освіта: Національний авіаційний університет, Київ, Україна, (2016).
E-mail: prostoigor1995@bigmir.net

Е. И. Чумаченко, И. В. Кривенко. Обучение модулей нейронной сети

Рассмотрена система гибридных нейронных сетей. Предложены методы обучения их модулей.

Ключевые слова: искусственный интеллект; связевые модели; двунаправленная ассоциативная память.

Чумаченко Елена Ильинична. Кандидат технических наук. Доцент.
Кафедра технической кибернетики, Национальный технический университет Украины «Киевский политехнический институт», Киев, Украина.
Образование: Грузинский политехнический институт, Тбилиси, Грузия, (1980).
Направление научной деятельности: системный анализ, искусственные нейронные сети.
Количество публикаций: более 50 научных работ.
E-mail: chumachenko@tk.kpi.ua

Кривенко Игорь Викторович. Бакалавр.
Кафедра авиационных компьютерно-интегрированных комплексов, Национальный авиационный университет, Киев, Украина.
Образование: Національний авіаційний університет, Київ, Україна, (2016).
E-mail: prostoigor1995@bigmir.net