Research Journal of Applied Sciences 10 (12): 815-818, 2015
ISSN: 1815-932X


# Application of Grammars of Kitano Graph Generation for Coding of Structure of Feedforward Artificial Neural Network

M.V. Akinin, N.V. Akinina. A.N. Kolesnikov, A.V. Sokolova and I.S. Soldatenko
Tver State University, Zhelyabova Street, 33, Tver, Russia

**Abstract:** In this study, we have considered one of developments of grammars of Kitano graph generation theory as an approach to describing structure of feedforward Artificial Neural Network (ANN) within the context of application of Genetic algorithms for search of its optimal structure. We have described in details applied Genetic algorithm of search of optimal feedforward ANN structure. Results of the experiments have been provided within the framework of which solution for complex realization task was given, namely the task of dividing classes "Softwood forest" and "Mixed forest".

**Key words:** Artificial neural network, feedforward neural network, grammars of Kitano graph generation, Genetic algorithm, classification, tree-like classifier

## INTRODUCTION

One of the most popular neural network concepts is feedforward ANN. Such ANN may be presented as a function y = ann(v) a function that produces corresponding response y for each input vector.

Feedforward ANN consists of several neurons which are connected with each other with synapses by which signals circulating in the network are transmitted. Each neuron also has some inhibiting threshold level that defines the level of neuron's reaction on corresponding firing. The term "direct distribution" means that there are no cycles in ANN-V vector, presented to ANN input, having systematically come through all neurons in the network one and only one time, appear in the output of the network, forming its Y response. Each neuron in the network transforms its input vector in some way in accordance with its threshold value and its activation function, transmitting its output through corresponding synapses to other neurons.

The majority of feedforward ANN refer to class of neural networks, studied with teacher. For teaching such ANN, learning set of $V_{teach}$ vectors is used for which:

$$d_{teach}\sum_{i=1}^{n}(X_i-\bar{X})^2$$

multitude of ideal responses is known. For testing quality of ANN learning, training multitude of $V_{test}$ is used for which ideal responses $d_{test}$ are also known.

Special case of feedforward ANN are multilayer perceptrons in which neurons are united in fully connected layers outputs of previous layer neurons are given in full volume to inputs of next neurons. This ANN structure is excessive; however, it is simple within the context of its description.

Generally, selection of feedforward ANN structure is a difficult task. There is a range of approaches aimed at its solution, among which the most developed one is an approach with application of Genetic algorithm (Aksenov and Novoseltsev, 2006) for the search of optimal structure of neural network in terms of root mean square composed function of E ($V_{test}$) error (Eq. 1):

$$E(\text{ann}, V_{test}) = \frac{\sum_i \left(\text{ann}\left(v_{test,\,i}\right)-d_{test,i}\right)^2}{V_{test}};\ i=\overline{1, V_{test}} \quad (1)$$

## CODING AND DECODING OF ANN STRUCTURE

An important problem that is being decided in the process of searching of optimal ANN structure with the help of genetic algorithm is a problem of coding of ANN structure in the view of continuous sequence of symbols which represent the sequence of chromosomes that compose the sole gene of each species' genome.

Kitano (1990) offered the method of coding ANN structure which was called grammars of Kitano graph generation. In this study, we study one of the developments of grammars of Kitano graph generation which lies in introducing of additional parameter W which defined the profundity of analysis of initial grammar rule and which allows determining networks of arbitrary size with the help of grammars that contain one and the same number of rules.

**Corresponding Author:** M.V. Akinin, Tver State University, Zhelyabova Street, 33, Tver, Russia

Feedforward ANN structure may be represented in the view of directed graphs without cycles. Each arc in this graph corresponds to synapsis and has its own weight which is determined in the process of network training. Each graph's peak corresponds to one neuron; each peak is connected with threshold value of corresponding neuron, being adjusted in the process of network training.

Arcs that are presented in graph may be determined with the help of incident matrixes G(2) of ANN, in which the point codes the presence of edge, directed from the peak with the number that equals to the number of line to the peak with the number which equals to number of column where zero is the absence of such edge, respectively:

$$G=\begin{pmatrix} g_{11} & g_{12} & \dots & g_{1j} & \dots & g_{1H_G} \\ g_{21} & g_{22} & \dots & g_{2j} & \dots & g_{2H_G} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ g_{i1} & g_{i2} & \dots & g_{ij} & \dots & g_{iH_G} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ g_{H_G1} & g_{H_G2} & \dots & g_{H_Gj} & \dots & g_{H_GH_G} \end{pmatrix}; \quad g_{ij}\in\{0,1\};\ i=\overline{1,H_G};\ j=\overline{1,H_G} \tag{2}$$

G matrix may be coded with the help of some rules of production grammar A(3):

$$A=\{T, T_P, T_N, F_T, F_P, F_N, S\}$$
$$T=\{0, 1, \omega\};\ T_P=\{T_{P\ i}\};\ i=\overline{1,16};\ T_N=\{T_{N\ j}\};\ j=\overline{1,H_N};\ S\in T_N$$
$$F_T=\left\{1\to\begin{pmatrix}1&1\\1&1\end{pmatrix};\ 0\to\begin{pmatrix}0&0\\0&0\end{pmatrix};\ \omega\to 0\right\}$$
$$F_P=\left\{\begin{matrix} F_{P\ 1}\to\begin{pmatrix}0&0\\0&0\end{pmatrix} & F_{P\ 2}\to\begin{pmatrix}0&0\\0&1\end{pmatrix} & F_{P\ 3}\to\begin{pmatrix}0&0\\1&0\end{pmatrix} & F_{P\ 4}\to\begin{pmatrix}0&0\\1&1\end{pmatrix} \\ F_{P\ 5}\to\begin{pmatrix}0&1\\0&0\end{pmatrix} & F_{P\ 6}\to\begin{pmatrix}0&1\\0&1\end{pmatrix} & F_{P\ 7}\to\begin{pmatrix}0&1\\1&0\end{pmatrix} & F_{P\ 8}\to\begin{pmatrix}0&1\\1&1\end{pmatrix} \\ F_{P\ 9}\to\begin{pmatrix}1&0\\0&0\end{pmatrix} & F_{P\ 10}\to\begin{pmatrix}1&0\\0&1\end{pmatrix} & F_{P\ 11}\to\begin{pmatrix}1&0\\1&0\end{pmatrix} & F_{P\ 12}\to\begin{pmatrix}1&0\\1&1\end{pmatrix} \\ F_{P\ 13}\to\begin{pmatrix}1&1\\0&0\end{pmatrix} & F_{P\ 14}\to\begin{pmatrix}1&1\\0&1\end{pmatrix} & F_{P\ 15}\to\begin{pmatrix}1&1\\1&0\end{pmatrix} & F_{P\ 16}\to\begin{pmatrix}1&1\\1&1\end{pmatrix} \end{matrix}\right\}$$
$$F_N=\{F_{N\ i}\};\ F_{N\ i}:T_{N\ i}\to\begin{pmatrix}a&ß\\\gamma&d\end{pmatrix};\ i=\overline{1,H_N};\ a, ß, \gamma, \delta\in(F_N\cup F_P) \tag{3}$$

Where:

- $T$ = Stands for the set of terminal symbols
- $T_p$ = Stands for the set of preterminal symbols
- $T_N$ = Stands for the set of non-terminal symbols
- $T_T$ = Stands for inference rules for terminal symbols
- $F_P$ = Stands for inference rules for preterminal symbols
- $F_N$ = Stands for inference rules for non-terminal symbols
- $S$ = Stands for starting non-terminal symbol

A grammar is called grammar of Kitano graph generation. One A Grammar corresponds to one species of population of genetic algorithm.

Elements T, $T_P$, $T_N$, $F_T$, $F_P$ are fixed for each A and consequently, species' gene codes only a set of rules $F_N$. Set of rules $F_N$ may be coded according to the scheme, presented at Fig. 1. Each rule $F_{N\ i}$ is reflected in four consecutive chromosomes $\alpha_i$, $\beta_i$, $\gamma_i$, $\delta_i$ which also follow $\delta_{i-1}$ chromosome in gene b of the species.

Algorithm of restoring matrix G from Grammar A consists of the following stages:

1. Initial matrix $G_1$ = S is created;
2. A step W = 1 is performedfrom the rules $F_T$, $F_P$ and$F_N$:

$$G_1=S\to G_2=\begin{pmatrix}a_S & ß_S\\ \gamma_S & d_S\end{pmatrix}$$

3. Step 2 is repeated for each of $\alpha_S$, $\beta_S$, $\gamma_S$, $\delta_S$ rules:

$$G_2=\begin{pmatrix}a_S & ß_S\\ \gamma_S & d_S\end{pmatrix}\to G_3$$

dimension $G_3$ consequently equals $2^{(3-1)}$ at $2^{(3-1)}=4$ at 4 rules

4. Step 3 is repeated (W-2) of times, until matrix $G_W$ is obtained
5. In matrix $G_W$ any symbol $g\in(T_P\cup T_N)$ is considered as a symbol f and is converted according to corresponding rule in 0
6. Matrix $G_W$ is converted intomatrix G according to Eq. 4 thus, it is
7. Guaranteed that in matrix G there will be no cycles

$$G=(g_{ij});\ G_W=(g_{W\ ij});\ g_{ij}=\begin{Bmatrix}0,\ i\geq j\\ g_{W\ ij},\ i<j\end{Bmatrix};\ i=\overline{1,H_G};\ j=\overline{1,H_G} \tag{4}$$

Parameter W of the algorithm of G incidence matrix recovery defined maximally possible number of neurons that may be in resultant ANN $2^{W-1}$ of neurons. Considering the fact that the number of inputs and outputs of ANN is limited, several neurons are adjoining to ANN that is being described by corresponding graph:

- Input neurons in sufficient amount are connected by synapses with graph's source such synapses are directed from input neurons to graph sources

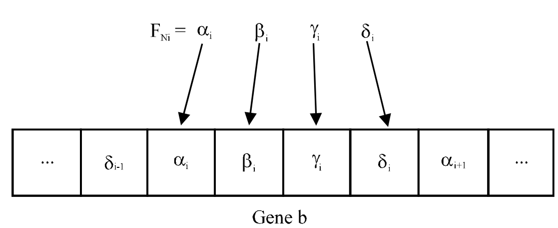


Fig. 1: Scheme of coding set of $F_N$ rules

- Output neurons in sufficient amount are connected by synapses with graph's sources such synapses are directed from graph sources to output neurons

Capacity $H_N$ of $T_N$ multitude, equally as W parameter, cause a significant influence at quality of work of genetic algorithm, since it is what defined gene's length (like $4H_N$ chromosomes) and correspondingly, the level of diversity of rules that are being coded by the gene. Choice of $H_N$ is performed heuristically small value of $H_N$ would lead to degeneration of population and its descent to local optimum while larger value of $H_N$, on the contrary would increase diversity of species, however, it would also increase the time of genetic algorithm's descent to possible global optimum.

In comparison with traditional version of Kitano grammars of graph generation, suggested version allows uniformly compactly describing neural networks with any maximal number of neurons which is obtained at the account of W parameter.

## GENETIC ALGORITHM

Genetic algorithm (Ulker and Ulker, 2012; Gladkov *et al*., 2006), used for the search of optimal ANN structure and for coding ANN structure of Kitano grammars of graph generation, consists of the following steps:

1. Multitude $V = \{v_i\}$; $i = \overline{1, Hv}$ of vectors is created from studied subject area and multitude $d = (d_i)$; $i = \overline{1, Hv}$ of ideal ANN responses to corresponding vectors from V; on the base of multitudes V and d at each iteration of genetic algorithm, training $V_{teach}$ and $d_{teach}$ will be created as well as test $V_{test}$, $d_{test}$ of multitude of vectors and responses
2. Initial population of species $B = \{b_i\}$; $i = \overline{1, H_B}$ is created randomly
3. Subsets $V_{taech}$, $V_{test}$ are randomly extracted from multitude v for multitudes $V_{teach}$ and $V_{test}$ corresponding subsets $d_{teach}$ and $d_{test}$ are extracted
4. For each species $b_i \in B$ corresponding grammars A $_i$is restored which produces neural network $ann_i$
5. For each species $b_i \in B$ fitness function $Q(_{bi})$ is calculated (Eq. 5):

$$Q(b_i) = \frac{1}{E(ann_i, V_{test}) + \varepsilon}; \ \varepsilon \in \mathbb{R}; \ \varepsilon < 10^{-8} \quad (5)$$

6. Species $b_i \in B$ are sorted in descending order of fitness function $Q(_{bi})$
7. Species with indices $I \in [1, H_{B\ test}]$ form the multitude of species $B_{cross} \subset B$; species with indices $I \in (H_{B\ best} + H_{B\ cross} + H_{B\ best} + H_{B\ cross} + H_{B\ mut}]$ form the multitude of species $B_{mut} \subset B$, at this the following statements must be performed (Eq. 6) (Rutkovskaya *et al*., 2013):

$$\begin{aligned} & H_{B\ best} \geq 2 \\ & H_{B\ cross} = \sigma H_{B\ mut}; \ \sigma \in \mathbb{R}; \ \sigma > 1 \\ & H_{B\ best} + H_{B\ cross} + H_{B\ mut} = H_B \\ & B_{best} \cap B_{cross} = \varnothing; \ B_{best} \cap B_{mut} = \varnothing; \ B_{cross} \cap B_{mut} = \varnothing \end{aligned} \quad (6)$$

8. Species $b_i \in B_{cross}$ are substituted byspecies $b_{cross\ kl}$ = cross $(b_k, b_l)$; $b_k$, $b_l \in B_{best}$ i.e., by the species which are results of crossing random species from multitude $B_{best}$ of the best species from population
9. Species $b_i \in B_{mut}$ are substituted by species $b_{mut\ k}$ = mutation$(b_k)$; $b_k \in B_{best}$, i.e. by the species which are results of random species' mutation from $B_{best}$ multitude of best species from population
10. If the best species from $b_1$ population was the best and corresponded to condition of $Q(b_1) > P$ stop at more than I previous algorithm iterations in a row then one should move to step 3, otherwise, the algorithm is stopped. Parameter P is a level that defines sufficient amount of ANN training

Upon the stop of ANN genetic algorithm, corresponding species $b_1$ from final population that is generated by $A_1$ is accepted as the best one. Genetic algorithm has the following parameters:

- $H_v$; $H_{v\ teach} = (V_{teach})$; $H_{v\ test} = (V_{test})$ define the dimensions of training and test vector samples; values of these parameters depend on subject area and difficulty of classification task that is being solved by desired ANN
- $H_{B\ best}$; $H_{B\ cross}$; $H_{B\ mut}$; σ define the number of species, preserved at switch to the next iteration of Genetic algorithm, substituted to results of crossing and mutation, respectively
- Cross, mutation are composed functions of crossing mutation, respectively
- P defined upper level of quality of training of the best ANN, sufficient for algorithm stop
- I defines the number of iterations that are sufficient for algorithm stop which should consequently repeat conditions for stopping for the best species it is obvious that the more I is the more is probability of descending algorithm to stable optimum, i.e., to ANN that gives equally quality result of training at random $V_{teach}$ and $V_{test}$

## AN EXPERIMENT

Offered algorithm of searching optimal feedforward ANN structure was tested in the course of building tree-like classifier, the structure of which is presented at

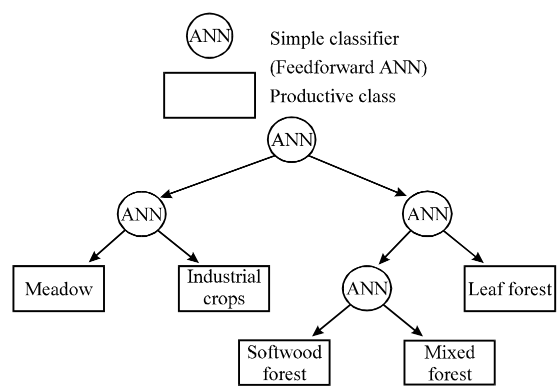


Fig. 2: Classifier structure

Fig. 2, used for building vegetation maps for Russian midlands according to Earth remote sensing data, obtained from CA 'Landsat 5'.

Genetic algorithm was used for selection of ANN structure which solved the task that was the most complex one within this complex, i.e., the task of dividing classes "Softwood forest" and "Mixed forest". Genetic algorithm was launched with the following parameters:

- W = 8 maximum amount of training neurons that could be in neural network, equaled $2^{W-1} = 2^7 = 128$ (without consideration of fixed number of input neurons which equaled dimensions of input vector v and one input neuron, since ANN performed binary classification)
- $H_N = 10$
- $H_v = 80$; $H_{v\ teach} = 30$; $H_{V\ test} = 50$
- $H_{B\ best} = 20$; $H_{B\ cross} = 20$; $H_{B\ mut} = 10$

$$P = 12.6 > \frac{50}{4}$$

- I = 5 which considering additional limitation $V_{teach} \cap V_{test} = \varnothing$, gave the probability $p_{test} \approx 0.9046$ of the fact that each vector $v \in V$ would be included into $V_{test}$ in the process of quality assessment of training the best species, thus, with a high probability the best species upon completion of algorithm would give ANN that provides acceptable result at each vector $v \in V$

## RESULTS

Two-point crossing was selected as a composed function of crossing cross. The size of gene areas, produced from each parent was selected identical and equaled 20 chromosomes. Substitution of random 20 chromosomes of the gene to random chromosomes (rules) from $(F_N \cup F_P)$ was selected as composed function of mutation.

Dependence of quality of ANN training, produced from the best species $b_1$ of population from the number of Genetic algorithm's iteration is presented at Fig. 3.

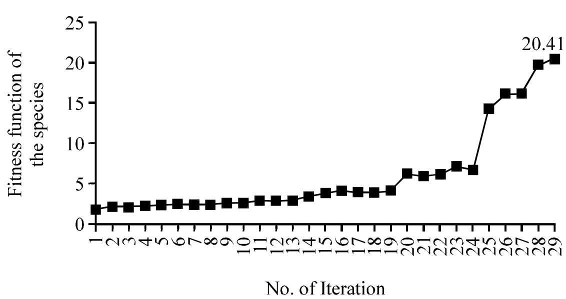


Fig. 3: Dependence of quality of ANN training from the number of Genetic algorithm's iteration

## CONCLUSION

Application of genetic algorithm for searching of optimal feedforward ANN structure and grammars of Kitano graph generation for coding of ANN structure within the course of using it is feasible in the tasks that require fine-tuning of ANN structure for achieving acceptable result.

The only disadvantage of defined application of genetic algorithm is its significant temporal complexity, since its application is not feasible for solving tasks for which consecutive selection of possible ANN structures gives sufficient result.

## ACKNOWLEDGEMENT

This study was published as part of the grant 2014-14-576-0047-097/RFMEFI57414X0084 (Ministry of Education and Science of the Russian Federation).

## REFERENCES

Gladkov, L.A., V.V. Kureichik, V.M. Kureichik, 2006. Genetic Algorithms. 2nd Edn., Fizmatlit, Moscow, Russian Federation, Pages: 320.

Kitano, H., 1990. Designing neural networks using genetic algorithms with graph generation system. Complex Syst. J., 4: 461-476.

Rutkovskaya, D., M. Pilinskiy, L. Rutkovskiy, 2013. Neural Networks. 2nd Edn., Hotline-Telecom, Moscow, Russian Federation, Pages: 384.

Ulker, E.D. and S. Ulker, 2012. Comparison study for clonal selection algorithm and genetic algorithm. Int. J. Comput. Sci. Inf. Technol., 4: 107-118.