ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 6, Issue 6 , June 2019

Construction of a Neural Network Using an Approach to a Genetic Algorithm.

Yusupbekov N.R., Mukhitdinov D.P.,Sattarov O.U.,Boybutayev S.B.

Professor, Tashkent State Technical University, Tashkent, Uzbekistan,
Senior lecturer, Navoi State Mining Institute, Navoi, Uzbekistan

ABSTRACT: A new approach to the formalization of the management process of production of phosphoric acid based on neural network control systems, which not only implements adaptive control methods, but also offers its algorithmic approaches to a number of problems whose solution is difficult, is substantiated and implemented. An improved scheme for searching the architecture of neural networks using genetic algorithms (route coding method) is proposed.

KEY WORDS: mathematical model, neural network, neural network algorithms, neural network architecture, PID controller, neuro-emulator.

I.INTRODUCTION

An approach to modeling a multiply connected technological process. Modern research in the field of the theory and practice of chemical technology is aimed at creating the scientific foundations of energy and resource saving processes. The analysis of processes for the extraction of phosphoric acid (EPA) by the methods of mathematical modeling is of significant theoretical and practical importance. To find the optimal technological mode and build an automated control system, they traditionally require an experiment, often costly and time consuming. Using the same methods of mathematical modeling allows, without the cost of real equipment, to obtain interesting information about the technological process in a convenient form, and also to calculate the optimal parameters for the process.

By combining in its structure complete models of technological objects, the computing complex allows an experiment to be carried out on a model of a real technological device and makes it possible to investigate the effect of various disturbances on the operation of the entire system as a whole. Of particular interest is the modeling and study of various modes in extraction plants, which are among the most complex technological control objects due to their multiplicity and large inertia. [1-3].

The problem of managing dynamic objects and systems under conditions of structural and parametric uncertainty is one of the important directions in modern control theory. When solving it, the application of the methods of neural networks (NS) is particularly relevant, which is currently receiving quite a lot of attention [4]?

In the method with the use of NA there are no restrictions on the linearity of the system, it is effective in the conditions of noise and after graduation provides real-time control. Neural network control systems (CS) are more flexibly tuned to real-world conditions, forming models that are fully adequate to the task, and do not contain restrictions related to the construction of formal systems. In addition, neural network control systems not only implement adaptive control methods, but also offer their algorithmic approaches to a number of tasks, the solution of which causes difficulty due to their insufficient formalization. Thus, it becomes possible to process within the same data model of the same nature - for the NA only their correlation is important. It is useful to combine traditional management with the potential and prospects of using systems based on the use of artificial NA.

The modern way to build neural networks is an evolutionary approach to searching the space of possible architectures. One of the most universal approaches is Holland's genetic algorithm [5]. The idea of genetic algorithms (GA) is based on the principle of evolution of biological creatures: out of the entire set of individuals of this species, only the most adapted individuals survive to the prevailing environmental conditions. Their beneficial properties are fixed and accumulated at the genetic level in subsequent generations due to the exchange of genes. Applying the GA approach to neural networks, we can say that the set of all possible neural networks acts as a population, the network architecture (number of layers, neurons, connections) is the genetic code, and the fitness criterion is the best accuracy of the results and the small size of the network.

Each individual population is a possible solution to the problem. The solution parameters are encoded into a character string – the genetic string. In the original method [5] proposed by Holland, a fixed-length binary string was

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 6, Issue 6 , June 2019

used. If the problem has several parameters, the genetic string contains a set of substrings (orgenes) corresponding to each parameter. In modern GA variants, a genetic string of variable length [6] of various structures is used: vectors, graphs, ordered lists, Lisp expressions [7]. It should be noted that the method of coding a task into a genetic string will essentially determine the effectiveness of the work of the GA. The coding of the solution of the problem into the genetic string is performed by the user of the GA. In the process of operation, the GA sthemselves do not possess information about the value of the encoded string. The main task of the GA is to manipulate sets of genetic strings — populations in the reproduction process using genetic operations.

In the reproductive process, new solutions (individuals) are created using choices, recombination and changes to existing solutions, based on the values of the evaluation function (fitnessfunction, fitnessfunction). The fitnessfunction assesses the fitness of each individual in relation to the "environment". Since each individual is a possible solution to the problem, the "environment" is the task at hand. If the task is to find the optimal neural network with the help of GA, then the function of fitness can be the value opposite to the network error or equal to zero if the network cannot solve the problem. It should be noted that the GA uses only the value of the fitnessfunction, but not the meaning. This allows you to build an arbitrarily complex evaluation function by combining several factors.

There are three main approaches to the integration of the GA and neural networks [8]:

1) the use of GA as a tool for finding the parameters of the neural network learning algorithm;
2) for direct learning network (search space weights);
3) to find the neural network architecture.

The first two tasks are sufficiently studied; a review on the use of GA as a tool for finding the parameters of the learning algorithm and direct network training is given in [9].

The general scheme of searching for the architecture of neural networks with the help of GA can be presented as follows. Information about the neural network architecture is coded in a special way into the genetic code. Then generations are generated, to which standard genetic operations are applied. Next, each individual is assessed: the neural network is restored from the genetic line and tested. The value of the fitnessfunction, as a rule, will depend on the value of the network error on the test set. GA operation ends when the desired error value is reached. [10].

II. NEURAL NETWORK ARCHITECTURE CONSTRUCTION USING ROUTE CODING METHOD.

When building a neural network using the approach of the genetic algorithm (GA), additional requirements may arise for the coding scheme. The genetic information in the line should contain information about the method of adding a building block in the main network.

Studies conducted in [11] show that generating coding for a number of problems is ineffective and requires more time to achieve a result. Therefore, to implement the search more efficiently use direct encoding scheme.

In the present work, a variant of direct coding is developed using the "routes" of signal transmission in a neural network proposed in [12]. This variant of direct coding implies the preservation in the genetic code of information not about the number of layers, neurons or connections in the neural network, but about conditional "routes" of signal transmission in the network from inputs to outputs.

In the variant proposed in [12], the "route" in the network is determined by a list of neurons, which begins with any of the input neurons and ends only with one of the output neurons. There are no restrictions on the representation and order of the neurons within such a "route". This option has some limitations: it assumes a limitation on the size of the network from above and can generate arbitrary neural networks, not limited to the class of direct propagation neural networks.

In the improved method, all requirements for the construction of a neural network using this method are described as follows.

1) Suppose that before the formation of the genetic string there is a basic direct-propagation neural network consisting of neurons T = {h1, ...,hγ} and having k inputs and o outputs. Let's set some number of building neurons U = {hγ + 1, ...,hγ + u}. All neurons {Γ, U} will participate in the formation of the "routes", that is, new neurons will participate equally in the construction of the "routes" along with the neurons of the core network.

2) The neural network architecture is defined by a set of sets - the "routes" of the {Pi} signal advancement. The route of signal advancement in the neural network is defined by an ordered list of neurons from G and U. This simulates the location of neurons sequentially in all layers, starting from the layer in which the initial neuron is located and ending with the layer with the final neuron in the network. Each route P can begin only from a neuron from the set of admissible starting neurons S and can be terminated by one neuron from the set of admissible output

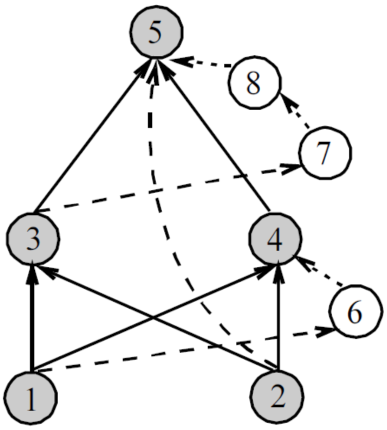
ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 6, Issue 6 , June 2019

neurons E. The repetition of neurons within one “route” is not allowed. There is no other restriction on the order of neurons within a route.

3) There are several possible ways of forming the sets S and E. In the practical implementation of this coding method, the authors stopped on the following. The set of admissible starting neurons S is formed from all the neurons of the basic neural network G, except for the network outputs and the last hidden layer. A set of permissible output neurons E, respectively, from all neurons in the core network G except for the network inputs and the first hidden layer. Of course, the U-building neurons do not fall into both sets.

4) In the genetic line, neurons are represented directly by their indices. Each route is separated from each other by a special service symbol '#'.

the proposed scheme is shown in Fig. 1.



Building neurons U = { 6 7 8 }
Set of valid starting neurons S = {1 2 3 4}
Set of valid output neurons E = {3 4 5}
Possible ways:
{1 6 4 5} 3 7 8 5} {2 5}
Genetic String: (variable part)

#3 7 8 5#	#1 6 4 5#	#2 5#

(The constant part, is not involved in the evolution)

#1 35#	#1 4 5#	#2 3 5#	#2 4 5#

Fig. 1. The proposed route coding option

In the method proposed in [12], the following genetic operations are applied to the constructed route: a crossing operator with two break points and four variants of the mutation operator. In the modification of the route coding method for working with genetic strings, the following genetic operations were used: crossing with one split point and two variants of the mutation operation. The operation of crossing (intersection), is used to consolidate the properties of the genetic string that are useful for solving the set problem. Crossing randomly selects a point between the routes in the genetic line of two individuals. As a result, two descendants will appear, composed of routes that belong to both parents.(рис. 2).

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 6, Issue 6 , June 2019

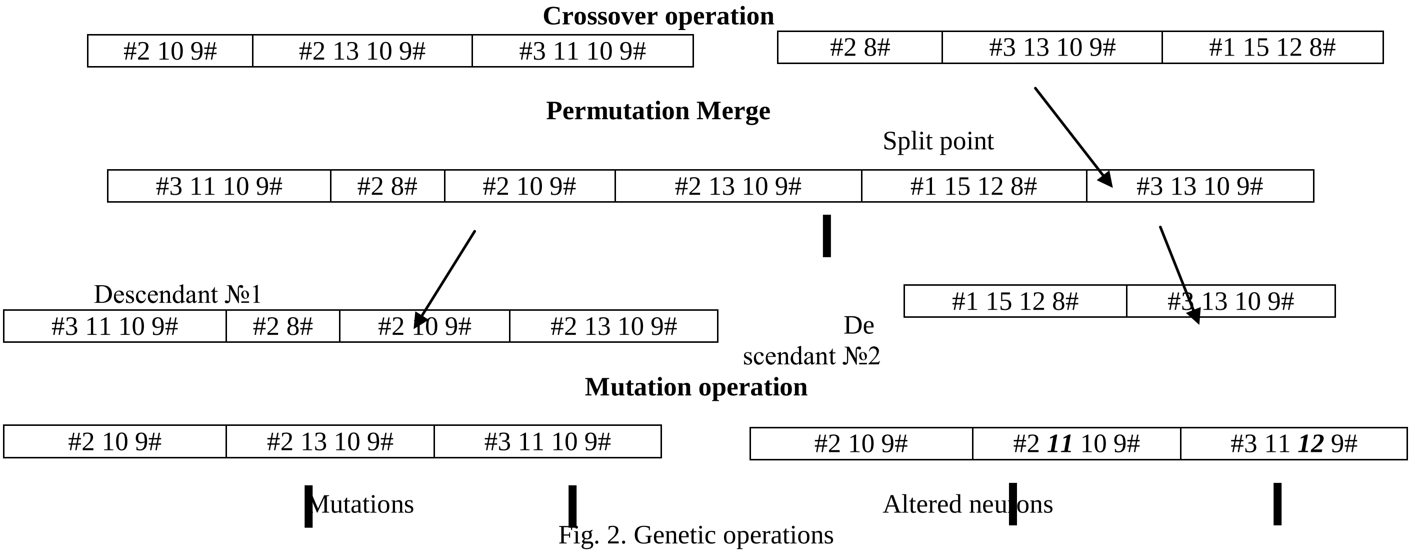


Fig. 2. Genetic operations

There are two options for making mutations in the genetic code. The first variant of the mutation operator replaces randomly the neurons in an arbitrary number of “routes” that constitute the individual’s genetic line. The second variant of the mutation operator provides for the complete creation of a genetic string at random. Making random changes - mutations - allows you to make jumps in the space of architectures and get out of local extremes that can be reached when searching for the optimal architecture. It should be noted that the use of genetic operations can lead to the appearance of incorrect architectures for a neural network of direct propagation. Recognition and correction of the genetic code of such individuals is made at the stage of recovery of the neural network from the genetic code.

III. CONCLUSION

The novelty and effectiveness of the above approach lies in the way of organizing evolutionary enumeration. In contrast to the generally accepted approaches of the evolutionary construction of neural networks, the genetic code that determines Δ at each stage will always have a small size and, accordingly, a smaller size of the search space. Conventional approaches deal with individuals trying to solve the entire problem at once.

In a short genetic line, building blocks are formed much faster, which speeds up the convergence of the method.

Thus, a new approach to the formalization of the control process for the production of EPC on the basis of neural network control systems has been substantiated and implemented, which not only implement adaptive management methods, but also offer their algorithmic approaches to a number of tasks whose solution is difficult. An improved scheme for searching the architecture of neural networks using genetic algorithms (route coding method) is proposed.

REFERENCES

[1]. A.G. Goryunov, Yu.A. Chursin, S.S. Mikhalevich, D.G. Sickly. Dynamic model of a multicomponent nonequilibrium extraction process in a column extractor // News of higher educational institutions. Physics, 2010.-№1182. - p.210-214.

[2].Mukhitdinov D.P., Avazov Yu.Sh. Dynamic models of distillation columns // scientific and practical journal "Modern materials, equipment and technology" № 5 (8), 2016, pp. 136-141.

[3].Kadirov YO.B., Mukhitdinov D.P. Calculation of flow rates and pressure drop // scientific and practical peer-reviewed journal "Modern materials, equipment and technology." –Kursk, 2016 No. 5 (8) -page. 84-92.

[4].Bodyansky Ye.V., Zaporozhets O.V., Ostrovskaya Zh.N. Adaptive neuroregulator of reduced order // Herald of Kharkov State Polytechnic University. - Kharkiv: KSPU. - 1999. - Issue.70. - P.112-117.

[5].Holland J.H. Adaptation in Natural and Artificial Systems. Univ. of Michigan Press., Second Ed. 1992, MA: The MIT Press edition, 1975.

[6].Fjalldal J.B. Evolving Neural Network Controllers Using Genetic Algorithms with Variable Length Genotypes // Tech. Rep., School of Cognitive and Computing Sciences at the Univ. Of Sussex, March 1999. (http://www.cogs.susx.ac.uk/users/johannf/report/report.html).

[7].Spears W.M., A. De Jong, Back T., et al. An Overview of Evolutionary Computation // Proceedings of the 1993 European Conference on Machine Learning, 1993. (http://www.aic.nrl.navy.mil/spears/papers/ecml93.ps).

ISSN: 2350-0328
International Journal of Advanced Research in Science, Engineering and Technology
Vol. 6, Issue 6 , June 2019

[8].Xin Yao. A Review of Evolutionary Artificial Neural Networks // International Journal of Intelligent Systems, 1991. (ftp://archive.cis.ohio-state.edu/pub/neuroprose/yao.eann.ps.gz).

[9].Hussain T.S. An Introduction to Evolutionary Computation // Tech. Rep., CITO Researcher Retreat, Ontario, 1998. (http ://www. cs. queensu. ca/home/hussain/).

[10].Branke J. Evolutionary Algorithms for Neural Network Design and Training // 1st Nordic Workshop on Genetic Algorithms and its Applications, 1995. (ftp://ftp.aifb.uni-karlsruhe.de/pub/jbr/Vaasa.ps.gz).

[11].Gruau F., Whitley D., and Pyeatt L. A comparison between cellular encoding and direct encoding for genetic neural networks // Proceedings of the First Genetic Programming Conference. – 1996. – P. 81 – 89.

[12].Jacob W., Rehder M. Evolution of neural net architectures by a hierarchical grammar-based genetic system // Proc. of the International Joint Conference on Neural Networks and Genetic Algorithms. – 1993. – P. 72 – 79.