УДК 519.8

NEW OPTIMIZATION METAHEURISTIC BASED ON CO-OPERATION OF BIOLOGY RELATED ALGORITHMS

Sh. A. Akhmedova, E. S. Semenkin

Siberian State Aerospace University named after academician M. F. Reshetnev
31, Krasnoyarsky Rabochy Av., Krasnoyarsk, 660014, Russian Federation
E-mail: shahnaz@inbox.ru, eugenesemenkin@yandex.ru

In this paper we describe and investigate a new self-tuning metaheuristic approach called Co-Operation of Biology Related Algorithms (COBRA) based on five well-known nature-inspired optimization methods such as Particle Swarm Optimization, Wolf Pack Search, Firefly Algorithm, Bat Algorithm and Cuckoo Search Algorithm. Besides, two modifications of COBRA are introduced. Also new metaheuristic was used for adjustment of neural network's weight coefficients for solving different classification problems.

Keywords: nature-inspired algorithms, self-tuning, neural networks, classification.

НОВЫЙ КОЛЛЕКТИВНЫЙ МЕТОД ОПТИМИЗАЦИИ НА ОСНОВЕ КООПЕРАЦИИ БИОНИЧЕСКИХ АЛГОРИТМОВ

Ш. А. Ахмедова, Е. С. Семенкин

Сибирский государственный аэрокосмический университет имени академика М. Ф. Решетнева
Российская Федерация, 660014, Красноярск, просп. им. газ. «Красноярский рабочий», 31
E-mail: shahnaz@inbox.ru, eugenesemenkin@yandex.ru

Описывается и рассматривается новый самонастраивающийся коллективный подох, названный кооперацией бионических алгоритмов (COBRA), основанный на пяти хорошо известных бионических алгоритмах: стайные алгоритмы, алгоритмы волчьих стай, мотыльковые алгоритмы, поиск алгоритмом «кукушки». Кроме того, включены две модификации алгоритма COBRA. Также новый коллективный метод используется для настройки весовых коэффициентов нейронной сети в решении различных задач классификации.

Ключевые слова: бионический алгоритм, самонастройка, нейронная сеть, классификация.

Existing metaheuristic algorithms such as Particle Swarm Optimization or Firefly Algorithm, for example, start to demonstrate their power in dealing with tough optimization problems and even NP-hard problems. Five well-known and very similar to each other nature-inspired algorithms such as Particle Swarm Optimization (PSO) [1], Wolf Pack Search (WPS) [2], Firefly Algorithm (FFA) [3], Cuckoo Search Algorithm (CSA) [4] and Bat Algorithm (BA) [5] were investigated by authors of this paper. Each of above listed algorithms was originally developed for solving real-parameter unconstrained optimization problems and imitates a nature process or the behavior of an animal group. For instance Bat Algorithm is based on the echolocation behavior of bats; Cuckoo Search Algorithm was inspired by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of other host birds (of other species), and so on.

Unconstrained optimization. Various test unconstrained optimization problems with various dimensions were used for the preliminary investigation of all mentioned algorithms: Rosenbrock's function, Sphere function, Ackley's function, Griewank's function, Hyper-Ellipsoidal function and Rastrigin's functions [6]. The comparison of obtained results showed that we can't say which approach is the most appropriate for any function and any dimension. The best results were obtained by different methods for different problems and for different dimensions; in some cases the best algorithm differs even for the same test problem if the dimension varies. Each strategy has its advantages and disadvantages, so a natural question is whether is it possible to combine major advantages of above listed algorithms and try to develop a potentially better algorithm?

These observations brought researchers to the idea of formulating a new metaheuristic approach that combines major advantages of various algorithms. So a new optimization method based on PSO, WPS, FFA, CSA and BA was implemented and investigated. Proposed approach was called Co-Operation of Biology Related Algorithms (COBRA). Its basic idea consists in generating five populations (one population for each algorithm) which are then executed in parallel cooperating with each other (so-called island model).

Proposed algorithm is a self-tuning meta-heuristic. That is why there is no necessity to choose the population size for each algorithm. The number of individuals in each algorithm's population can increase or decrease depending on the fact weather was the fitness value improving on current stage or not. If the fitness value wasn't improved during a given number of generations, then the size of all populations increases. And vice versa, if the fitness value was constantly improved, then the size of all populations decreases. Besides, each population can "grow" by accepting individuals removed from other population. Population "grows" only if its average fitness is better than the average fitness of all others populations. Thereby we can determine "winner algorithm" on each iteration/generation. The result of this kind of competition allows presenting the biggest resource (population size) to the most appropriate (in the current generation) algorithm. This property can be very useful in case of a hard optimization problem when, as it is known, could be no single best algorithm on all stages of the optimization process execution.

The most important driving force of the suggested meta-heuristic is the migration operator that creates a co-operation environment for component algorithms. All populations communicate with each other: they exchange individuals in such a way that a part of the worst individuals of each population is replaced by the best individuals of other populations. It brings up to date information on the best achievements to all component algorithms and prevents their preliminary convergence to its own local optimum that improves the group performance of all algorithms.

For better understanding whether COBRA is workable and useful following experiments were conducted. First of all, the same six test problems by proposed algorithm were solved; it demonstrates comparative performance (table 1).

Additional observation was that COBRA outperformed component algorithms in a number of cases and this number increases with the problem dimension.

Next experiment was conducted in following way. All five original algorithms were compared with the new algorithm COBRA on 28 test functions from [7] following experiments settings described there. Problem dimensions used for comparing were 2, 3, 5, 10, 30. For each problem, "winner algorithm" was established by two criteria – algorithm given the overall best result ("best") and algorithm demonstrated the best mean result averaged over 51 runs ("mean"). Again it was observed that number of "wins" for COBRA by both criteria increases with the problem dimension. For the criterion "best" COBRA has 7 wins on the test problems of the dimension 2, 11 wins when dimension was 5 and 26 wins when dimension was 30. Number of wins for the criterion "mean" were, accordingly, 19, 24 and 28. Algorithms competition results for dimensions 10 and 30 are presented in table 2 below where table cells contain the name of winning algorithm.

So numerical experiments and comparison showed that COBRA is superior to its component algorithms (PSO, WPS, FFA, CSA, BA) when dimension grows and more complicated problems are solved. It means that the new algorithm should be used instead of components algorithm when the optimization problem dimension is high and optimizes function properties are complicated.

Table 1

Results obtained by COBRA for the first six test problems

Func	Dimension	Success rate	Average population size	Average number of function evaluations	Average function value	STD
1	2	100	20	263	0.000652238	0.000346687
	3	100	24	605	0.000750922	0.000290652
	4	100	27	757	0.000790054	0.000297926
2	2	100	20	284	0.000753919	0.000272061
	3	100	22	552	0.000783528	0.000290029
	4	100	27	932	0.000817905	0.00028088
3	2	100	29	867	0.000588745	0.000307145
	3	100	33	1470	0.000774339	0.000282613
	4	100	32	1604	0.000739637	0.000372214
4	2	100	20	202	0.000678884	0.000320224
	3	100	25	581	0.000749783	0.000282332
	4	100	28	1085	0.000756105	0.000286405
5	2	100	22	369	0.000806724	0.000140685
	3	100	22	574	0.000989866	0.00140048
	4	100	28	885	0.000695163	0.000159342
6	2	100	27	860	180.001	0.000273844
	3	100	40	2082	170.001	0.000247725
	4	100	56	3877	160.001	0.000336504

Table 2

Performance comparison of COBRA and component algorithms (28 test functions from [7])

Func	Best D = 10	Mean D = 10	Best D = 30	Mean D = 30
1	PSO	PSO	PSO	COBRA
2	COBRA	COBRA	COBRA	COBRA
3	COBRA	COBRA	COBRA	COBRA
4	COBRA	COBRA	COBRA	COBRA
5	PSO	PSO	WPS	COBRA
6	COBRA	COBRA	COBRA	COBRA
7	COBRA	COBRA	COBRA	COBRA
8	WPS	COBRA	COBRA	COBRA
9	COBRA	COBRA	COBRA	COBRA
10	COBRA	COBRA	COBRA	COBRA
11	COBRA	COBRA	COBRA	COBRA
12	COBRA	COBRA	COBRA	COBRA
13	COBRA	COBRA	COBRA	COBRA
14	WPS	COBRA	COBRA	COBRA
15	COBRA	COBRA	COBRA	COBRA
16	COBRA	COBRA	COBRA	COBRA
17	PSO	COBRA	COBRA	COBRA
18	COBRA	COBRA	COBRA	COBRA
19	COBRA	COBRA	COBRA	COBRA
20	WPS	COBRA	COBRA	COBRA
21	PSO	COBRA	COBRA	COBRA
22	COBRA	COBRA	COBRA	COBRA
23	WPS	COBRA	COBRA	COBRA
24	COBRA	COBRA	COBRA	COBRA
25	FFA	COBRA	COBRA	COBRA
26	COBRA	COBRA	COBRA	COBRA
27	COBRA	COBRA	COBRA	COBRA
28	PSO	COBRA	COBRA	COBRA

Constrained optimization. Next step in our study was about development and investigation a modification of COBRA that can be used for solving constrained real-parameter optimization problems. For these purpose three constraint handling methods were used: dynamic penalties [8], Deb's rule [9] and technique that was described in [10]. Method proposed in [10] was implemented to PSO-component of COBRA; at the same time other components were modified by implementing firstly Deb's rule and then calculating function values by using dynamic penalties. The performance of the proposed algorithm was evaluated on the set of 18 scalable benchmark functions provided for the CEC 2010 competition and special session on single objective constrained real-parameter optimization [11], when the dimension of decision variables is set to 10 and 30, respectively. For each function 25 runs are performed. A maximum function evaluation was 200000 for dimension 10 and 600000 for dimension 30. Obtained results are presented in table 3 and table 4.

Constrained modification of COBRA was compared with algorithms that took part in competition CEC 2010. Finally it was established that proposed approach is superior to 3–4 of 14 methods from this competition. Besides, COBRA outperforms all its component algorithms.

On COBRA effectiveness in binary space. As it was mentioned, all above described algorithms was originally developed for continuous valued spaces. However many applied problems are defined in discrete valued spaces where the domain of the variables is finite. For this purpose binary modification of COBRA was developed. COBRA was adapted to search in binary spaces by applying a sigmoid transformation to the velocity component (PSO, BA) and coordinates (FFA, CSA, WPS) to squash them into a range [0, 1] and force the component values of the positions of the particles to be 0's or 1's. The sigmoid expression is given in [12].

Table 3

Results obtained by constrained modification of COBRA for dimension D = 10

Func	Best	Worst	Mean	STD	Feasibility Rate
1	-0.727373	–0.559235	–0.637199	0.0594299	100
2	1.82813	4.34865	2.75755	0.926259	100
3	8.87653	8.89164	8.87895	0.00318718	92
4	5.57793	5.58908	5.58215	0.00312711	28
5	189.952	516.713	338.063	82.963	32
6	103.515	563.247	369.431	72.4124	24
7	0.529035	0.888604	0.566389	0.0676125	100
8	21.8649	53.8881	23.4468	6.23478	100
9	1.9133e+012	2.4654e+012	2.04001e+012	2.12584e+012	68
10	2.30765e+011	2.84432e+012	6.58941e+011	8.31906e+011	84
11	0.0002073305	0.00843533	0.00190705	0.000479151	68
12	–169.36	671.962	-102.82	228.253	0
13	–57.1851	-54.9831	-57.0463	0.429771	100
14	7.9878	1.72346e+007	6.97436e+006	8.37255e+006	100
15	6.9986e+009	4.94244e+011	7.56845e+010	1.22078e+011	100
16	0.724961	1.17519	0.826752	0.165962	56
17	103.275	321.092	111.988	42.6833	100
18	378.76	671.905	425.911	77.383	96

Table 4

Results obtained by constrained modification of COBRA for dimension D = 30

Func	Best	Worst	Mean	STD	Feasible rate
1	–0.625073	-0.270351	-0.42015	0.132247	100
2	4.0493	5.03476	4.57562	0.395698	92
3	28.6807	28.707	28.6861	0.00729205	36
4	9.13159	9.13525	9.13303	0.00120324	20
5	478.746	555.016	485.285	18.7327	40
6	493.301	600.586	504.521	25.1907	32
7	0.334309	4.23197	1.61077	1.30831	100
8	470.46	1023.44	896.351	220.978	100
9	2.63268e+012	7.38962e+012	3.12913e+012	1.05068e+012	56
10	1.25487e+012	3.94721e+012	1.96828e+012	5.87269e+011	44
11	–0.00825323	-0.00281327	-0.00443743	0.000266256	16
12	155.45	720.707	403.327	105.636	0
13	-64.3938	-53.8018	-58.2296	4.54096	100
14	398.499	2.90799e+007	1.24046e+006	5.68939e+006	100
15	4.93728e+009	4.92773e+012	2.14661e+012	1.80896e+012	100
16	1.15237	1.37727	1.19964	0.0478142	8
17	332.563	381.055	344.201	20.7101	92
18	290.033	357.092	355.187	13.2994	100

Table 5

Results obtained by binary modification of COBRA

Func	Dimension	Success rate	Average population size	Average number of function evaluations	Average function value	STD
1	2	100	31	740	0.000182069	0.000248506
	3	99	68	3473	0.000188191	0.000689647
	4	93	80	6730	0.00579879	0.0242297
2	2	100	27	567	0.000236274	0.000265351
	3	100	30	775	0.000150127	0.000168235
	4	100	32	916	0.000355086	0.00029257
3	2	100	32	1439	0.00019874	0.000330485
	3	91	51	2046	0.00150713	0.00245315
	4	83	62	3030	0.00126295	0.00281119
4	2	100	33	931	0.000209168	0.000268542
	3	100	32	868	0.000191162	0.000233884
	4	92	79	1710	0.000347666	0.000257291
5	2	100	30	899	0.00032841	0.000468681
	3	90	65	1332	0.000506847	0.00140048
	4	90	160	2258	0.00411721	0.158903
6	2	100	28	1734	180.0002	0.000185362
	3	98	36	3294	169.801	0.169149
	4	96	41	5462	159.2	0.279294

The same six problems (Rosenbrock's function, Sphere function, Ackley's function, Griewank's function, Hyper-Ellipsoidal function and Rastrigin's functions) were used for testing new algorithm; maximum number of function evaluations was equal to 100000. Obtained results are presented in table 5. Experiments showed that COBRA's binary modification works successfully and reliable but slower than original COBRA for the same problems with smaller success rate obtained. Such result was expected as the binary modification needs more computing efforts in continuous variables space and shouldn't be used instead of original COBRA. However, it can be recommend for solving optimization problems with binary representation of solutions.

New metaheuristic applications. COBRA was successfully used for the adjustment of neural network's weight coefficients for solving different classification problems such as bank scoring problems and medical diagnostic problems.

First two applied bank scoring problems were solved: bank scoring in Germany (20 attributes, 2 classes, 700 records of the creditworthy customers and 300 records for the non-creditworthy customers) and in Australia (14 attributes, 2 classes, 307 examples of the creditworthy customers and 383 examples for the non-creditworthy customers). Benchmark data were taken from [13]. For these two problems the structure of neural networks was fixed as a single hidden layer perceptron with 3 or 5 neurons, each having bipolar sigmoid as an activation function. From optimization view point, these problems have from 45 till 105 real-valued variables. Obtained results are presented in table 6 below where the portions of correctly classified instances from validation sets are presented. There are also results of other researchers used other approaches (as were found in scientific literature).

Then COBRA was used for adjustment of neural network's weight coefficients for solving medical diagnostic problems. Structure of neural networks was fixed as a single hidden layer perceptron with 3 or 5 neurons, each having bipolar sigmoid as an activation function. Also was used the network with three hidden layers with 3 neurons on each layer. Two applied problems were solved with developed network: Breast Cancer Wisconsin (11 attributes, 2 classes, 458 records of the patients with benign cancer and 241 records of the patients with malignant cancer) and Pima Indians Diabetes (9 attributes, 2 classes, 500 patients that were tested negative for diabetes and 268 patients that were tested positive for diabetes). Benchmark data were also taken from [13]. Obtained results are presented in table 7 and table 8 below where the portions of correctly classified instances from validation sets are presented. There are also results of other researchers used other approaches (as were found in scientific literature).

COBRA's binary modification for constrained optimization problems was successfully used in solving the investment allocation optimization problem for a machine building concern. This problem contains 50 binary variables and tens constraints. In all runs, the algorithm found the best known solution.

Conclusions. So in this paper the new meta-heuristic, called Co-Operation of Biology Related Algorithms (COBRA), based on five well-known nature-inspired algorithms such as Particle Swarm Optimization, Wolf Pack Search, Firefly Algorithm, Cuckoo Search algorithm and Bat Algorithm was introduced. New approach was developed for solving real-parameter unconstrained problems. Proposed algorithm is a self-tuning method, so one has no need in controlling the population size while this is the most essential parameter for above listed component algorithms. In fact, one does not have to fine tune this parameter for a specific problem. Then proposed algorithm was validated and compared with its component algorithms. Simulations and comparison showed that COBRA is superior to these existing component algorithms when dimension grows and complicated problems are solved.

Table 6

Classifiers's performance comparison for bank scoring problems

Classifier	Scoring in Australia	Scoring in Germany
2SGP	0.9027	0.8015
C4.5	0.8986	0.7773
Fuzzy	0.8910	0.7940
GP	0.8889	0.7834
CART	0.8744	0.7565
LR	0.8696	0.7837
CCEL	0.8660	0.7460
RSM	0,8520	0,6770
Bagging	0.8470	0.6840
Bayesian	0.8470	0.6790
Boosting	0.7600	0.7000
k-NN	0.7150	0.7151
This study: ANN+COBRA (5)	0,8907	0,7829
This study: ANN+COBRA (3)	0,8898	0,7809

Table 7

Classification accuracies obtained with COBRA and other classifiers for breast cancer problem

Author (year)	Method	Classification accuracy (%)
Quinlan (1996)	C4.5	94.74
Hamiton et al. (1996)	RAIC	95.00
Ster and Dobnikar (1996)	LDA	96.80
Nauck and Kruse (1999)	NEFCLASS	95.06
Pena-Reyes and Sipper (1999)	Fuzzy-GA1	97.36
Setiono (2000)	Neuro-rule 2a	98.10
Albrecht et al. (2002)	LSA machine	98.80
Abonyi and Szeifert (2003)	SFC	95.57
Übeyli (2007)	SVM	99.54
Polat and Günes (2007)	LS-SVM	98.53
Guijarro-Berdias et al. (2007)	LLS	96.00
Akay (2009)	SVM-CFS	99.51
Karabatak and Cevdet-Ince (2009)	AR + NN	97.40
Peng et al. (2009)	CFW	99.50
A. Marcano-Cedeño, J. Quintanilla-Domínguez, D. Andina (2011)	AMMLP	99.26
This study (2013)	ANN+COBRA (3x1)	97.62
	ANN+COBRA (5x1)	97.67
	ANN+COBRA (3x3)	98.16

Table 8

Classification accuracies obtained with COBRA and other classifiers for diabetes problem

Author (year)	Method	Classification accuracy (%)
Mehmet Recep Bozkurt1, Nilüfer Yurtay, Ziynet Yılmaz1, Cengiz Sertkaya (2012)	PNN	72.00
	LVQ	73.60
	FFN	68.80
	CFN	68.00
	DTDN	76.00
	TDN	66.80
	Gini	65.97
	AIS	68.80
H. Temurtas, N. Yumusak, F. Temurtas (2009)	MLNN with LM(10xFC)	79.62
	PNN (10xFC)	78.05
	MLNN with LM	82.37
	PNN	78.13
S. M. Kamruzzaman, Ahmed Ryadh Hasan (2005)	FCNN with PA	77.344
K. Kayaer., T. Yıldırım (2003)	GRNN	80.21
	MLNN with LM	77.08
L. Meng, P. Putten, H. Wang (2005)	AIRS	67.40
This study (2013)	ANN+COBRA (3x1)	79.65
	ANN+COBRA (5x1)	79.71
	ANN+COBRA (3x3)	79.83

After that COBRA was modified for solving real-parameter constrained problems. A set of scalable benchmark functions provided for the CEC 2010 competition and special session on single objective constrained real-parameter optimization was used for testing of a proposed method. As result its usefulness and workability were established. Besides, constrained modification was compared with algorithms from this competition and showed better results than some of them.

Also in this study binary modification of COBRA was formulated. COBRA was adapted to search in binary spaces by applying a sigmoid transformation. This method was originally proposed by Kennedy and Eberhart for PSO, but it can be used for other nature-inspired algorithms as well. Binary version of COBRA was validated and compared with real-parameter version of COBRA.

And the last, all implemented variants of new meta-heuristic were used for solving applied problems.

We should notice that in this study only original design of component algorithms was used having in mind the examination of our idea as such. Although some modifications of them are known that improve their performance. As this idea is useful in its simple implementation, we hope that further modifications will give also a positive effect. Among possible modifications are the use of improved version of component algorithms, the use of more sophisticated ways of algorithms cooperation, the including other meta-heuristics in cooperation, etc. Also this potentially powerful optimization strategy can easily be extended to study multi-objective optimization problems.

As about applications, we have already successfully used this algorithm for adjustment of neural network's weight coefficients for solving classification problems. COBRA was used for neural network's weight coefficients adjustment for solving two bank scoring problems (Australian and German) and two medical diagnostic problems (Breast Cancer Wisconsin and Indians Diabetes). We used simple and small structures for network. Artificial neural network was fully connected and there were a lot of inputs, so we had to find optimal solutions for big dimension problems. But even with these simple structures ANN showed good results while solving problems. In the future we intend to modify the algorithm for tuning also neural network's structure. For this purpose we'll use binary modification of COBRA.

References

1. Kennedy J., Eberhart R. Particle Swarm Optimization. In Proceedings of International Conference on Neural networks IV. 1995. P. 1942–1948.

2. Chenguang Yang, Xuyan Tu and Jie Chen. Algorithm of Marriage in Honey Bees Optimization Based on the Wolf Pack Search. In Proceedings of International Conference on Intelligent Pervasive Computing, IPC2007. 2007. P. 462–467.

3. Yang X. S. Firefly algorithms for multimodal optimization. In Proceedings of 5th Symposium on Stochastic Algorithms, Foundations and Applications, SAGA 2209. 2009. P. 169–178.

4. Yang X. S., Deb S. Cuckoo Search via Levy flights. In Proceedings of World Congress on Nature & Biologically Inspired Computing, NaBic2009. 2009. P. 210–214.

5. Yang X. S. A new metaheuristic bat-inspired algorithm. In Proceedings of International Workshop on Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), SCI 284. 2010. P. 65–74.

6. Molga M., Smutnicki Cz. Test functions for optimization need. 2005. 3 kwietnia.

7. Liang J. J., Qu B. Y., Suganthan P. N. and Hernandez-Diaz A. G. Problem Definitions and Evaluation Criteria for the CEC 2013 Special Session on Real-Parameter

Optimization. Zhengzhou University, Computational Intelligence Laboratory, Zhengzhou China, and Nanyang Technological University, Singapore. 2012.

8. Eiben A.E., Smith J.E. Introduction to evolutionary computation. Springer, Berlin, 2003.

9. Deb K. An efficient constraint handling method for genetic algorithms. Computer methods in applied mechanics and engineering, 186(2-4). 2000. P. 311–338.

10. Liang J. J., Shang Zhigang, Li Zhihui. Coevolutionary Comprehensive Learning Particle Swarm Optimizer. In Proceedings of Congress on Evolutionary Computation (CEC). 2010. P. 1505–1512.

11. Mallipeddi R., Suganthan P. N. Problem Definitions and Evaluation Criteria for the CEC 2010 Competition on Constrained Real-Parameter Optimization. Nanyang Technological University, Singapore. 2009.

12. Kennedy J., Eberhart R. C. A discrete binary version of the particle swarm algorithm. In Proceedings of the World Multiconference on Systemics, Cybernetics and Informatics 1997, Piscataway, NJ. 1997. P. 4104–4109.

13. Frank A., Asuncion, A. UCI Machine Learning Repository. Irvine, CA: University of California, School of Information and Computer Science. Citing Internet sources available from: <http://archive.ics.uci.edu/ml>. 2010.

УДК 519.87

DEVELOPMENT OF ADAPTIVE GENETIC ALGORITHMS FOR NEURAL NETWORK MODELS MULTICRITERIA DESIGN*

C. Yu. Brester, E. S. Semenkin

Siberian State Aerospace University named after academician M. F. Reshetnev
31, Krasnoyarsky Rabochy Av., Krasnoyarsk, 660014, Russian Federation
E-mail: abahachy@mail.ru, eugenesemenkin@yandex.ru

In this paper modifications of single- and multi-objective genetic algorithms are described and testing results of these approaches are presented. The gist of the algorithms is the use of the self-adaptation idea leading to reducing of the expert significance for the algorithm setting and expanding of GAs' application capabilities. On the basis of offered methods the program system realizing the technique for neural network models design was developed. The effectiveness of all algorithms was investigated on a set of test problems.

Keywords: genetic algorithms, multicriteria optimization, self-adaptation, neural networks, classification.

РАЗРАБОТКА АДАПТИВНОГО ГЕНЕТИЧЕСКОГО АЛГОРИТМА ДЛЯ МНОГОКРИТЕРИАЛЬНОГО ПРОЕКТИРОВАНИЯ НЕЙРОСЕТЕВЫХ МОДЕЛЕЙ

К. Ю. Брестер, Е. С. Семенкин

[1]Сибирский государственный аэрокосмический университет имени академика М. Ф. Решетнева
Российская Федерация, 660014, Красноярск, просп. им. газ. «Красноярский рабочий», 31
E-mail: abahachy@mail.ru, eugenesemenkin@yandex.ru

Рассматриваются модификации одно- и многокритериального генетических алгоритмов и представлены результаты численных исследований этих подходов. Преимущество разработанных методов заключается в использовании идеи самоадаптации, позволяющей отказаться от привлечения эксперта для настройки алгоритма и расширить возможности использования генетических алгоритмов. На основе предлагаемых подходов разработана программная система для проектирования нейросетевых моделей. Эффективность всех алгоритмов была исследована на тестовых задачах.

Ключевые слова: генетический алгоритм, многокритериальная оптимизация, самоадаптация, нейронные сети, классификация.

* The study was supported by The Ministry of education and science of Russian Federation, project 14.B37.21.1521.