Efficiency Improvement of the Algorithm Based on an Artificial Immune System Modeling Applied to Continuous and Combinatorial Problems

Timur Zheldak, Illia Ziborov, Vladyslav Lyman and Andrii Zhuk

Dnipro University of Technology, av. Dmytra Yavornytskoho, 19, Dnipro, 49005, Ukraine

Abstract

One of the heuristic optimization methods that implements the simulation of an artificial immune system is discussed in this article. The list of optimization problems, their classification and known approaches of solving those ones are reviewed on the whole. The research of the main settings of operators in the proposed algorithm is applied to the different dimensions test problems. For the first time, an attempt of finding a universal approach of solving both combinatorial and continuous problems, applying the proposed algorithm, was implemented based on the common approach to the algorithm operators. The most efficient settings, based on the minimum solution search time criteria, are recommended for agent generation search, parental individuals selection, mutation, crossover, local search and population compression operators.

Keywords [1]

Artificial immune systems, AIS, algorithm, optimization, heuristic method, mutation, Saaty selection, combinatorial problems

1. Introduction

Optimization problem solving in terms of the multicomponent industrial production is considered to be a daily practice. Decision makers are to incorporate several criterias, which are generally ranged by their significance, and to solve the problems in the multidimensional space. One of the mechanisms that allow solving this kind of complex mathematical problems is evolutionary decision-making technologies.

A feature of evolutionary technologies is the heuristic (usually not due to mathematical establishment) nature of the decision-making algorithm, which can be improved, using new, more effective in practice heuristics. Many of them belong to analogical methods - a tool of applied systems analysis, which is used for study and simulation of complex systems analogously to biological or inorganic nature.

One of described heuristics is an artificial immune systems simulation algorithm, which allows finding a solution of the conditional or unconditional optimization problem, using a combination of evolutionary operators that simulate the human immune system. A detailed analysis of artificial immune system simulation algorithms applied to optimization problems solving is given in [1] and [2]. At the moment, algorithms following the principles of artificial immune systems simulation are actively used to solve applied problems of technological control and optimization processes in industry. The purpose of this study is to improve the efficiency of the optimization algorithm based on artificial immune system simulation by increasing the efficiency of its individual operators: agent generation, parental individuals selection, mutation, crossover, local search and population compression.

II International Scientific Symposium «Intelligent Solutions» IntSol-2021, September 28–30, 2021, Kyiv-Uzhhorod, Ukraine
EMAIL: zheldak.t.a@nmu.one (A.1); Ziborov.Il.K@nmu.one (A.2); Lyman.V.I@nmu.one (A.3) Zhuk.A.V@nmu.one (A.4)
ORCID: 0000-0002-4728-5889 (A.1); 0000-0002-3118-387X (A.2); 0000-0002-2557-7266 (A.3); 0000-0001-7763-2469 (A.4)
© 2021 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).
CEUR Workshop Proceedings (CEUR-WS.org)

CEUR-WS.org/Vol-3106/Paper_8.pdf

2. The current state of the research subject

2.1. Algorithm of modelling artificial immune systems

Modeling artificial immune system (AIS) is a method of computable intelligence that uses the analogy with one of the natural systems, namely the human defense system. As artificial neural networks, artificial immune systems can learn new information, recall previously learned information, and perform pattern recognition in a highly decentralized manner. In [1] author provides an overview of the immune system approach from the computational viewpoint.

In research [2], authors classified, compare, and generalize the general nowadays techniques and algorithms based on an idea of modeling AIS. Also considered theoretical aspects, methods, and algorithms three of the main, and total techniques of AIS, namely algorithm of negative choice, the algorithm of an immune network, and algorithm of clonal selection. The primary focus for analysis of their similarities and differences, data types, learning methods, separate operators that using, and the sphere of implementation of AIS.

Authors [3] say that the emergence of nature-inspired algorithms (NIA) is a significant milestone in the computational intelligence community. As one of the NIAs, the AIS shows its effectiveness, implicit parallelism, flexibility, and applicability when solving various engineering problems. However, the authors notice that AIS also stays in problem with evolution premature, local localization of minima, and at slow convergence owing to stochastic search dynamics.

Many efforts have been made by various researchers to increase the search effectiveness of AIS on the different sides, namely: helping population diversity, adaptive parameters control, etc. Authors [3] concentrate on the most effective evolution operators - mutation operator using "multi-learning". With him depending on the result, giving from previous steps, for the heirs in every generation perhaps using differentiated by force and principles mutation. In this way, authors trying to improve the working dynamics of an algorithm, which, when trying to solve quite a big problem, quickly found a suboptimal solution but need a long time to clarify.

Instead, at work [4], suggested break problem with large dimension on a few small. The latter is significantly improves the working time of AIS when solving a combinatorial problem. For a problem where variables are long binary vectors, the authors proposed to break him on a few small vectors, and perform a local search of bits, that more than others affects the general solution. Then global solutions search as a combination of local influences signs. The proposed break large dimension of search on "subset" considerably minimizes the time of calculation of fitness function and the total time of the working algorithm. As the authors say, experimental results on an extensive diversity of synthetic datasets and UCI show that our proposed method achieves better performance as a modern global algorithm for solving the combinatory problem.

Previously with AIS, namely using an approach of clonal selection, resolve a combinatory problem of optimal choice a carrier of the plurality outdoor advertising offers, available in the formulation of the knapsack problem [5]. The proposed realization of the algorithm has been investigated on test plural with previously knowing global optimum for a various number of a potential solution. Also on research was shown, that the AIS method instead of classical genetic algorithm (GA) reliably finds the exact solution for a problem where dimension less than 1000 and find the same solution quickly in absolutely time and more quality investigate area next to the optimum. It is noted that modeling methods of an artificial immune system may be used to solve a large range of combinatorial search and constrained optimization problems.

Also, an algorithm founded on clonal selection was used in [6] for solving some tasks in real space. Noting random character of mutation, that present in AIS based on clonal selection main search operators, authors propose hybrid algorithm clonal selection learning. There are two mechanisms (Baldwinian learning and orthogonal learning) that have opposite directions. Baldwinian learning is used for investigation (global search), orthogonal - for investigating areas of current decision (local improvement). At the same time, computational complexity increases slightly (global optimum found slowly), but increase reliability (percentage of launches, where randomly find global optimum).

Another approach also based on learning while working propose authors [7], that named them algorithm "Two-Population Co-evolutionary Algorithm with Dynamic Learning Strategy for Many-

Objective Optimization”. This is an AIS algorithm used only as a common search pattern, where settings change in the process according to achieve the result.

Similar principles use one recent research [8], where researched self-study another famous heuristic method — genetic algorithm (GA). In research shows that change characters of mutation and use local purposeful search can give more productivity on some testing problems. In research, authors prove that the best operators for solving problems in multidimensional real space it is rank selection, discretion crossover, uneven mutation, and operator of local search that uses Newtonian approach (based on first- and second-order derivatives).

The idea of involvement local search to the modeling artificial immune system algorithm based on the clonal selection use one of author's researches [9]. Authors propose, by analogy from island GA, to break the population of cells on a subset, evolution in which occurs independently. After some periods, individuals can exchange genetic information with individuals from other generations. This gain opportunity investigates all areas of search. As in various versions of AIS, in [9] propose to save previously find solutions for future generation using them new individuals, when in some time search lost effectuality.

Summing up, mark the following features of modern algorithms that use AIS for modeling for solving optimization problems:

- applying different selection operators significantly improves search speed and effectivity;
- a new generation must create partially using crossover, not through only cloning;
- must use different mutation operators, and radius (size of surroundings) should be determined during the working of the algorithm (learn);
- not enough attention to compressed population operators, that determine the diversity of the new generation.

2.2. The HINO-SF algorithm

Majority mentioned ideas is incorporated in the Hybrid Immune Network Optimization algorithm with Saaty selection and Fibonacci search [10], named as HINO-SF by authors. It consists of the following steps:

1. The population of antibodies is generated randomly. A counter of generations is set to $t=0$.

2. If preassigned maximum number of generations is achieved then transition to step 11 should be made; otherwise - transition to step 3.

3. Depending on the cells' adaptability assessment of the current generation, the number of clones is assigned for each cell of the current generation - it is the selection operator.

4. Cloning of cells in an amount is determined by the selection operator.

5. Recombination of cloned individuals by using the operator of probability crossover.

6. Calculation the dynamic probability of mutation for clones and implementation of adaptive mutation operator.

7. Execution of the one-dimensional local search for each clone with a certain probability. The method of golden ratio (Fibonacci) for a random coordinate is used in continuous problems and hill climbing method is used in combinatorial problems instead – it is local search operator.

8. Compression of the general population of parents and clones using the appropriate to a specified level N_p – it is compression operator. Memorizing the best of solution, if it has not occurred before.

9. Destruction of cells, whose age has reached the set value T_{max}, from the current generation. Their place in the main population is occupied by cells, randomly generated uniformly in the search area.

10. The counter of generations increments $t = t + 1$. Go to the step 2.

11. The output of the current generation as a set of suboptimal solutions.

12. Stopping the algorithm.

Block diagram of the algorithm is shown in Figure 1. This algorithm research, its limitations, and optimal configuration of its operators are dedicated to the study.

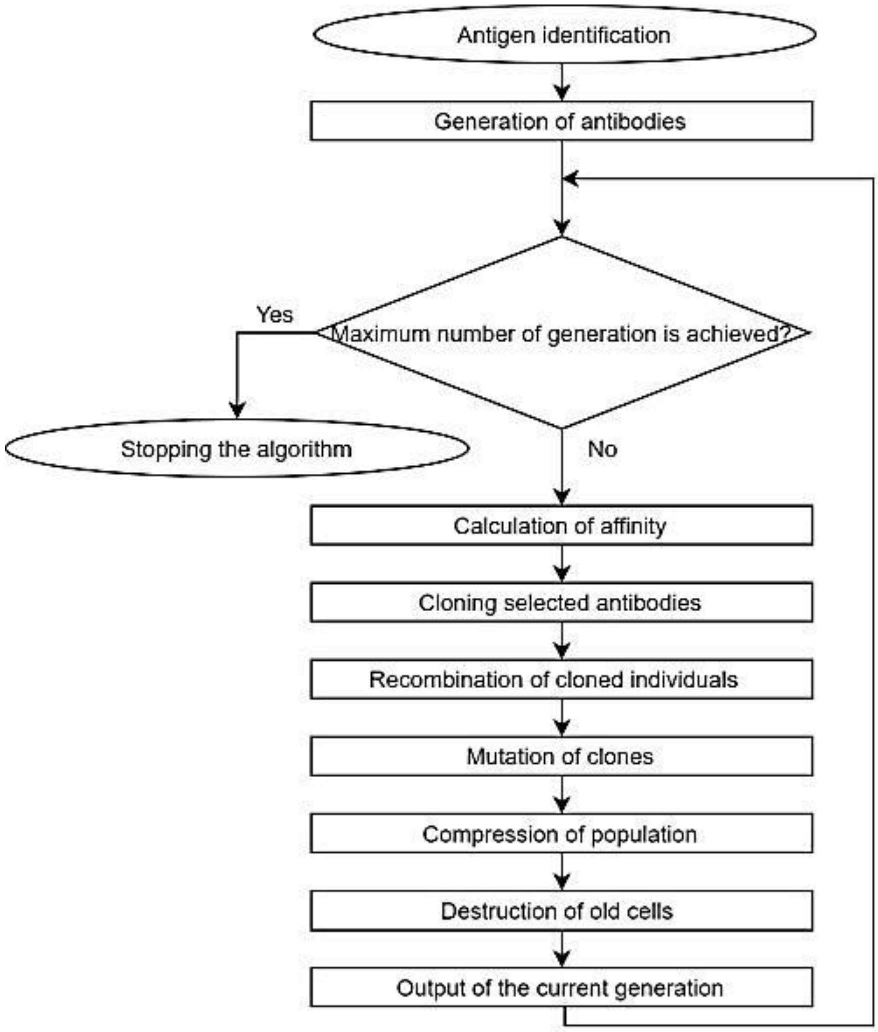

Figure 1: Mechanism of formation of antibodies in the immune system

2.3. Problems solving with the AIS

The foresaid Hybrid Immune Network Optimization algorithm was investigated both for solving problems in an infinite space and for solving combinatorial kind of problems. The last one is described in detail, following [10]. It is the problem of orders execution planning, implementing by a metallurgical enterprise, which typically belongs to the combinatorial optimization problem class of the scheduling theory. At the same time, the order execution scheduling problem is characterized with the features of another well-known combinatorial problem - the construction of a travel salesman's route (Traveling salesman problem, TSP), particularly, open Loop Traveling salesman problem.

The other one combinatorial applied optimization problem is the steel amount minimization problem which provides the steel order size limit of the only one cast volume [11]. The developed model specifies the possibility pre-evaluate the billet optimal size, based on the necessary cutting along the final product length, appropriate for the certain billet form of section, and ingot weight limits at the top and bottom.

In addition to reviewed applied problems, several test sets of various sizes and complexity were used, which describe the knapsack problem, the minimum vertex cover problem, and the set partition problem. All the problems are formulated according to the classic statements. All while in both case of problem [10] and [11] the transition from constrained optimization to an unconstrained problem, which is solved by the algorithm formulated in Section 1.2, was implemented, using the penalty functions method. Basically, solutions, which were not satisfied with the constraints, were acceptable, but worse than those, which were satisfied with the same ones.

The test problems of Rastrigin, Ackley, Rosenbrock, and others considered in [7] were used in an infinite space - a total of 30 test functions. Each of functions was tested in two-, ten-, twenty five- and one hundred- dimensional space. Under these circumstances the authors evaluated both the accuracy of the obtained solution (deviation from the known global optimum) for the allotted time, and the time spent on achieving acceptable accuracy of the calculation.

An additional factor that was studied in solving both combinatorial and continuous problems was the number of calls to the objective function. This factor is significant in case of the machine costs, especially in solving combinatorial problems, where the calculation of the objective function is based on matrix multiplication.

3. The study content

This issue is devoted to solving a number of applied and test problems using the algorithm described. The settings of algorithm operators that provide better convergence and search speed for the known optimal solutions are found.

3.1. Uniformity of new search agents generation

In any population-based global optimization algorithm like partial swarm optimization (PSO), evolution strategy (ES), GA, or AIS and among others a pseudo-random number generator is used to sample the first N_{pop} individuals of the initial generation from a uniform probability distribution $U(0;1)$. If its generation power N is smaller than $10^3 - 10^4$, it results in too nonuniform mesh.

The same authors suggest using quasi-random numbers, which have better homogeneity than pseudo-random numbers [13]. Quasi-random numbers share many similarities with pseudo-random numbers, but they are deterministically chosen based on low-discrepancy sequences.

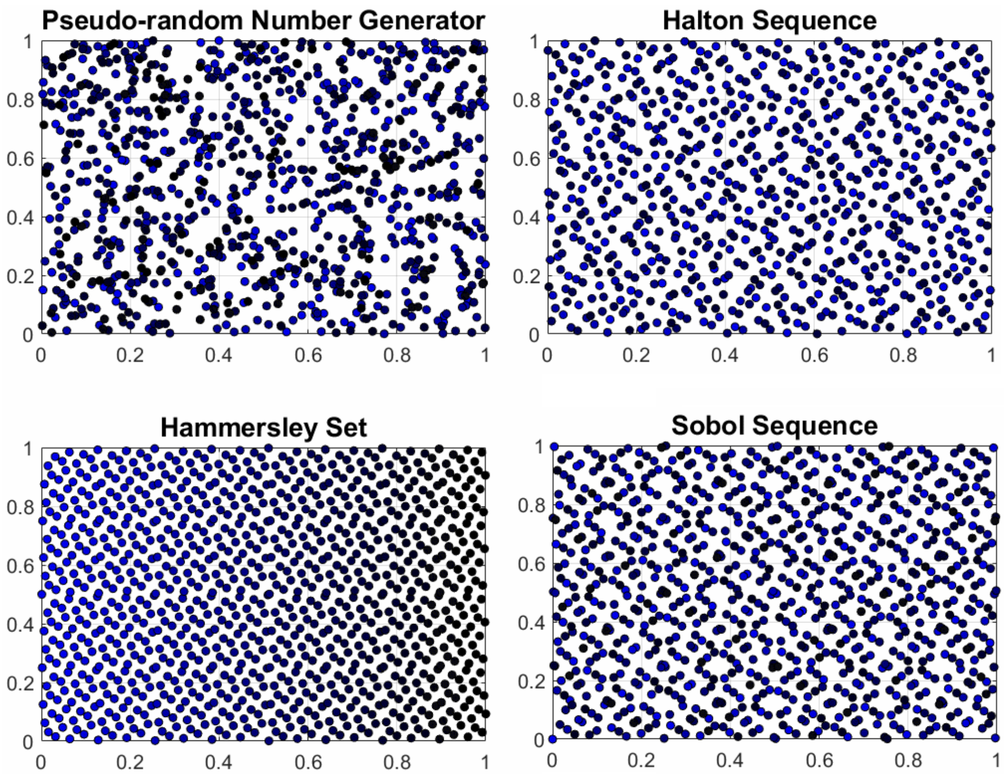

Figure 2: Comparing pseudo-random (left top) and quasi-random generators in 2-dimension space

The Halton sequence and the Hammersley set define two deterministic ways of sampling a D-dimensional space so that the consecutive points are as far apart as possible from each other. In essence, the Halton sequence and the Hammersley set are both a generalization of the Van der Corput sequence for multidimensional cases.

Sobol sequences are another widely used quasi-random number generator, which was invented by Ilya M. Sobol back in 1967. This quasi-random number generator uses a base of 2 to generate $U(0;1)$, to then perform a special re-ordering of the master sequence for each one of the dimensions of the sampled hyper-space [14].

The Figure 2 shows compared 1000 points in 2-dimension space from pseudo-random sample (top left), from the Halton sequence (top right), from the Hammersley set (bottom left) and from the Sobol sequence (bottom right). Quasi-random sequences cover the space more evenly than pseudo-random.

We applied the described generators in the algorithm shown in Figure 3 to the set covering problem *scpcyc07* from OR-Library (selection the minimum number of subsets from the 672 available to cover the set of 448 graph vertices. Each subset contains exactly 4 vertices) [15] and Rastrigin problem at multidimensional continuous space. At the same time built-in MATLAB system classical generators were used for comparison. Results of comparison at first 10000 iterations are illustrated in Table 1 and Figure 3.

Table 1

Time to fist finding of global optimum, iterations

Generator		Dimension of space 10	25	65	100
pseudo-random	Mersenne vortex (default)	136	1702	10274	64987
	Combined recursive	118	1476	11320	103951
	Multiplicative Fibonacci	86	1099	9920	68357
quasi-random	Halton sequence	111	1375	9811	61159
	Hammersley set	103	1258	8917	51478
	Sobol sequence	84	1005	8497	66123

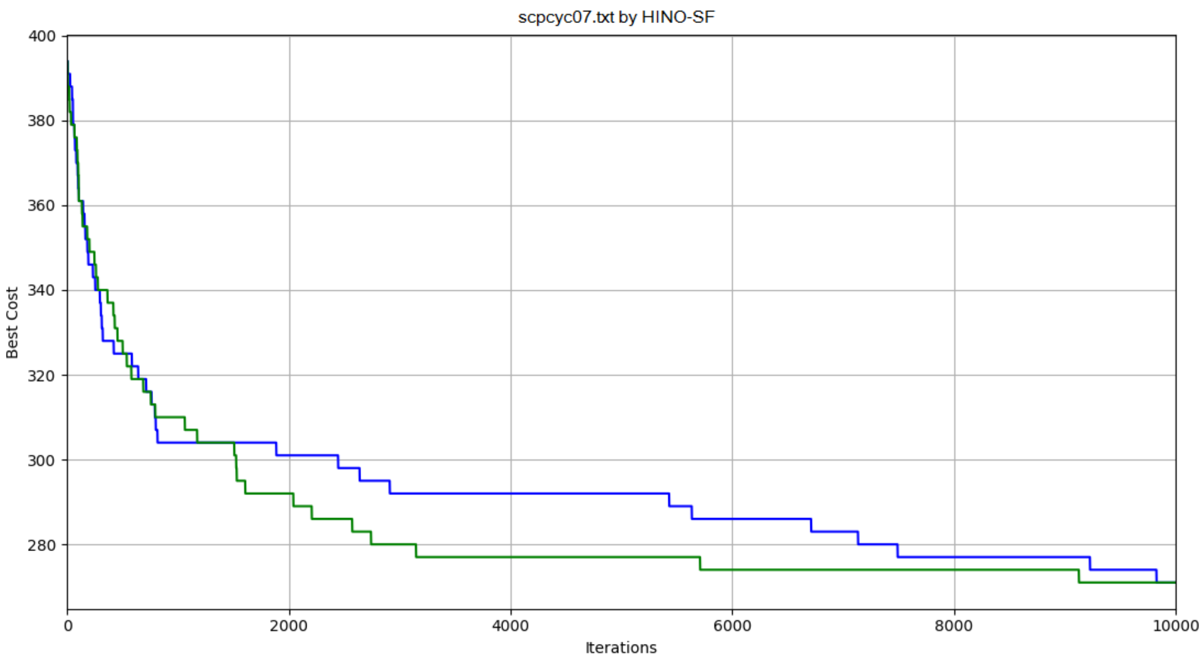

Figure 3: Best solution the set covering problem at first 10000 iterations

Table 1 contains comparing same pseudo-random and quasi-random generators for Rastrigin problem solving at 10, 25, 65 and 100 dimensions of continuous space. Instead, Figure 3 shows the time dependence of the best solution of the sets covering problem for the first 10,000 iterations.

All quasi-random generators show better result than pseudo-random generators. This result is also proved by increasing the dimension of the problem when filling the solution space becomes an important factor. Using of quasi-random generators accelerates on average the search for the global optimum by 12-20% of the time.

3.2. Parental individuals selection

One of the main factors that determine the efficiency of the optimization algorithm, based on the artificial immune system, is the choice of parental individuals used to obtain clones. In [10] it is proposed to use the selection operator based on Saaty's pairwise comparisons. In this case, the matrix of candidate pairwise comparisons for offspring was filled only for decisions that are included to the first half of the segment $[\varphi_{best}\ ; \varphi_{worst}]$. The rating of each of candidate j is calculated by

$$a_j = 1 + \lfloor \varphi(X_j) \cdot k \rfloor, \tag{1}$$

where k - the dimension of the scale (default k = 9).

Construction of a matrix of pairwise comparisons for the first line of which an expression $b_{1i} = a_1/a_i$ is used, and for the rest - $b_{ij} = b_{1j}/b_{1i}$.

The HINO-SF algorithm analysis with the described selector resulted in some limitations:

- with a small population among potential parents there are 1 or 2 applicants, the solution of which is better than the average value of the segment $[\varphi_{best}\ ; \varphi_{worst}]$;
- among the clones, the total majority (or all) will be descendants of one of the representatives of the current generation.

In fact, the selection operator described in [10] deprives the multi-agent search AIS heuristic of its one advantages - search by several agents in different parts of the search area simultaneously.

A modified selector, based on the method of Saaty's pairwise comparisons, also described in [16], was used to evade this problem. The matrix of pairwise comparisons is based on the following principle:

$$b_{ji} = \begin{cases} |a_i - a_j| + 1, & if\ j \geq i \\ \dfrac{1}{b_{ji}}, & if\ j < i \end{cases} \tag{2}$$

The calculation of the membership function is as follows:

$$\mu(\varphi(X_i)) = \sqrt[n]{\prod_{j=1}^{n} b_{ji}}\ . \tag{3}$$

The most wide-spread types of selectors, such as rank (ordinal) and tournament (roulette), were used for comparison of several problems solving, as well. Competitive ones are proved to be highly efficient among other evolutionary algorithms - GA [8], in particular.

The comparison of these selection operators is demonstrated in Table 2. In this table, the selection operators are compared by two main factors: the number of iterations (generations) to the first global optimum occurrence and the number of calls to the objective function. Both values are calculated on average over 10 runs with different random initial values. Both combinatorial and continuous problems, as well as different dimensions are considered.

According to the results in Table 2, the following conclusions can be outlined:

• The proposed selection operator based on the Saaty's scale pairwise comparisons method is equally effective in both binary and continuous space;

• In some problems (minimum graph tree, Ackley problem) the proposed selector shows a higher efficiency than the tournament 1.5 - 3 times). The rest of the problems use the same time and access to the target function.

3.3. Crossover applying

The proposed AIS algorithm mainly differs from other analogues on the application of an adaptive crossover operator to clones, which involves the exchange of genetic information. It is to diversify the search for the best solution in the algorithm. Therefore, the more successfully it will perform, the greater speed is to find the global optimum. The choice of crossover type depends on the space in which it is applied to. The combinatorial problems were solved with approaches which differ from other ones applied to the solutions of continuous problems. The most essential factors for evaluating the solutions efficiency are the time spent searching for a solution and the value of the deviation of the objective function from the known global optimum.

Table 2
The speed of the AIS algorithm depending on the selection operator

Problem	↓ Dimensionality Selector →	Iterations				Fitness Function Calls			
		Rank	Tournament	Saaty	Modif.Saaty	Rank	Tournament	Saaty	Modif.Saaty
Knapsak	1000	126	105	256	97	14015	11505	29711	10095
MinTree	100	536	455	450	143	59935	52869	53159	14865
SetCovP	672	829	746	758	807	87544	81991	82634	83505
Rastrigin	100	8426	7913	9510	6374	101206	88725	108462	78124
Ackley	25	889	849	1024	765	45873	44562	51268	36630
Rosenbrock	2	104	80	105	91	2518	1998	2478	2235

3.3.1. Crossover choice for combinatorial problems

Having used the selection, founded on modified Saaty operator, as well as the standard values of the mutation operators (their settings are drawn below) the previously described binary crossovers are compared. One-point, two-point and uniform crossovers were considered.

Figure 4: Comparison of three crossovers for combinatorial problems

A comparison of the solution determining speed (in the number of iterations) and the deviation from the exact solution (in percent) is illustrated in Figure 4. The data in the figure is averaged over 10 runs with random initial populations.

According to the results, the two-point crossover happens to be the best for all combinatorial problems solutions. On the one hand, it collects the most detail about the parental genes (as well as the single-point crossover), but also provides some diversification due to two breaks in the genome. At the same time, offspring, produced with a uniform crossover, significantly differs from each of the parents for considered problems. The solutions are discarded in the new search area and in most cases are much worse than the parental individuals, so they are no longer used.

Particular attention was paid to the probability P_{cross}, which the crossover is applied to the newly formed clone with. In a wide range from 0 to 0.9 (sometimes up to 0.95), higher P_{cross} probability values resulted in a faster search for the global optimum regardless of problem type and its dimension. At the same time, while $P_{cross} = 1$, the search efficiency drops rapidly. Therefore, the crossover operator is recommended to use with probability differs from 1.

3.3.2. Crossover choice for the continuous problems

Having applied the selection founded on the modified Saaty operator, the generation value $N_{pop} = 5$, the number of offsprings $N_c = 50$, the adaptive mutation operator described in [10] with the degree of mutation $mutLevel = 0{,}1$, crossovers for continuous space are compared. Such crossovers as direct, uniform, linear, arithmetic, heuristic, Laplace and SBX were tested.

A comparison of the solution determining speed (in the number of iterations) and the deviation from the exact solution is illustrated in Figure 5. The data in the figure is averaged over 10 runs with random initial populations.

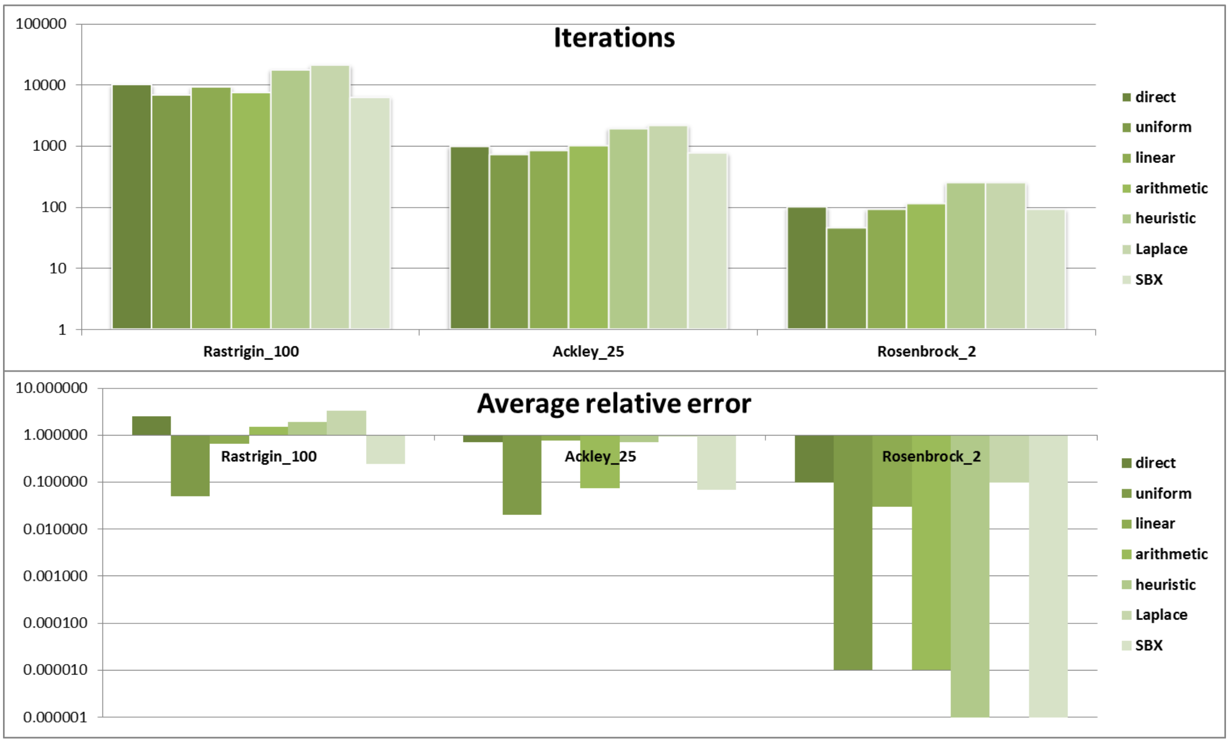

Figure 5: Comparison of seven crossovers for continuous problems

Analysis of the results in Figure 5 proves that the Laplace and heuristic crossovers perform 2-2.5 times more iterations, while not providing a sufficient quality result on different problems of various dimensions. The SBX crossover, which is used in the basic algorithm [10], and which is most often used in continuous space, is conventionally effective. It almost always provides a transition to the global optimum and requires a minimum of iterations for its achieving.

Meanwhile, a uniform crossover, which selects the parental genes with equal probability, resulted in higher values of accuracy on large-scale problems. Similar results were obtained earlier in the study of optimal settings of genetic algorithms [8]. It is further proposed to use a uniform crossover to solve problems in multidimensional continuous space, as this operator is an order of magnitude faster than a complex SBX crossover.

3.4. Mutation width and type

The main operator responsible for finding a solution in the proposed algorithm, as in most implementations of the method of artificial immune systems, is the mutation operator. An adaptive operator, which randomly applies either a narrow mutation to search in the current solution neighborhood, or a broad mutation to change the search area, is proposed. As in the case of the crossover operator, the mutation operator form significantly depends on the space, where the problem is solved. The choice of mutation range performance (wide or narrow) is based on the usefulness assessment of the current cell on the utility assessments scale on the current iteration throughout the population. The scale is determined for each j-th cell individually with the formula

$$mutSelect_j = \left(1 - \gamma \frac{t}{T_{max}}\right) \frac{\varphi_{max} - \varphi_j}{\varphi_{max} - \varphi_{min}}, \qquad j = [1{:}N_c], \tag{4}$$

where j – clone number in the population of clones; N_c – clone population size; t – current iteration number; T_{max} – provided maximum number of iterations; φ_{max} – the worst value of the objective function in the generation; φ_{min} – the best value of the objective function in the generation; φ_j – the value of the objective function for the current clone; γ – the maximum allowable proportion of broad mutations in the total.

If a random $U(0;1)$ is less than $mutSelect_j$ following (4), a narrow mutation is performed for clone j (search in the solution neighborhood). Otherwise - a broad mutation is performed - the transition to a new search area.

For continuous space, it is proposed to use a combination of a broad Gaussian mutation and a narrow polynomial distribution mutation. Both are described in detail in [10]. It was found that the value of γ, included in expression (4), should be low values of 0.1… 0.25 for the low dimension of the problem and increase up to 0.8… 0.9 at high dimension for continuous problems. It is explained with the fact that in a small dimension space, the main factor is to consider the neighborhood of the best current solutions. At 60 or more measurements, exactly the broad mutation demonstrates a significant impact on the speed of global optimum finding.

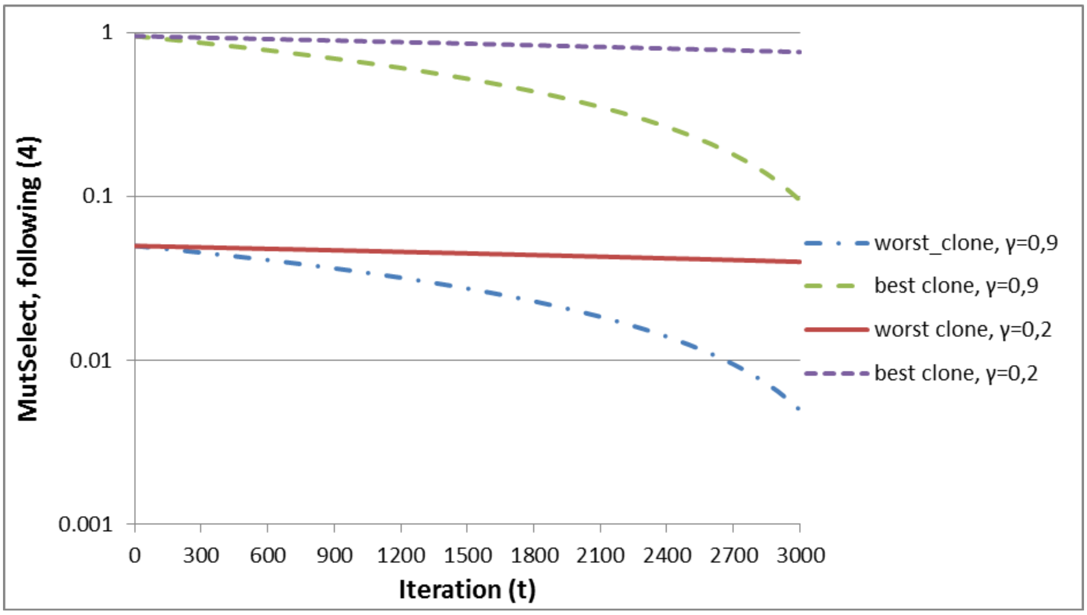

Figure 6: The optimal radius of the compression operator depending on the space dimension

Having analyzed formula (4), it is easy to notice that the best representatives in the generation at the initial stage of the algorithm are prone to narrow search mutations, at the final - to wide. Instead, the worst members of the generation almost always perform only broad mutations.

The dependence of the $mutSelect_j$ characteristic against the current iteration number, the clone adaptability degree and the admissible maximum proportion of wide mutations is illustrated by the graph in Figure 6.

Analyzing the obtained dependences, it is noticeable that in case of low γ values the worst clones in the population are almost always make a wide mutation, and successful clones make a narrow mutation. Increasing the dimension, the mutation nature choice is changing significantly. In multidimensional continuous space in case γ = 0.9 at the beginning of the calculation, the clones behave similarly to that described above. As we approach the end of the calculation, the proportion of narrow mutations decreases even for the best of the clones by 10 times.

When solving combinatorial problems, a point mutation of a certain number of genes is performed as a narrow mutation. The number of inverted genes is determined with the formula

$$mutLevel(t) = \begin{cases} MLevel_{min}\left(1 + P_{mut}^{0} - \dfrac{2t \cdot P_{mut}^{0}}{T_{max}}\right), & if\ t < T_{max}/2 \\ MLevel_{min}, & if\ t \geq T_{max}/2 \end{cases}, \tag{5}$$

where t - number of current iteration; $MLevel_{min}$ – the minimum level of mutation (usually 1 or 2 bits); P_{mut}^{0} – the prior probability of mutation operator using (global parameter of algorithm, in [10] recommended $P_{mut}^{0} = 0{,}5 \ldots 1$) and T_{max} – maximum number of generations.

Then, regardless of the time of algorithm on the initial iterations mutation will be per-formed for all clones, on final - less than for a half of them.

Extensive mutation in combinatorial problems can be realized through saltation. It is the process of the exchange between a certain number of bits, which are located at a certain distance from each other in the genome. Saltation mutation is effective due to its saving the total number of 0 and 1 in the genotype, thus not rejecting the current solution too far in the search space.

The mutation radius adapting issue in continuous space remains to stay opened. For today, the authors do not manage to substantiate any pattern. The best solution time and accuracy are achieved with different settings for various problems and multidimensional space.

3.5. Whether a local search is required

One-dimensional local search, proposed in [10], was used in the study of the algorithm to a limited extent. The matter is that the Fibonacci local search method calls the objective function dozens of times to find a solution in only one coordinate in the above one-dimensional notation. Therewith, this search method is not goal-oriented and it can be applied only to problems in continuous space. Analyzing the results of the previous studies [8], quasi-Newtonian methods which required to the using of first- and second-order derivatives are more effective in comparison with zero-order methods (Fibonacci, Nelder-Mead, Powell, etc.). In this case, local search operator should not be applied unconditionally to all offsprings, but it must be used with some probability only when traditional crossover and mutation operators are ineffective for a certain number of generations.

The topic of further research should be such a goal-oriented local search operator determination that would improve the objective function of a random descendant without spending significant computation time.

At the same time, the HINO-SF algorithm works effectively without the local search applying for combinatorial problems solving. The limited local search based on the hill climbing method is proposed earlier in [5]. It reduces the search time for a solution in the number of the algorithm iterations. However, the number of calls to the objective function increases in comparison with the option without local search. As a result, the astronomical time to obtain a solution, using local search, is several times longer than without its applying.

3.6. Population compression

The population compression operator determines the success of finding solutions in the paradigm of artificial immune systems. It prevents premature convergence and coming to a halt in local optimums, which is inherent in all evolutionary algorithms. In most sources [7, 9, 10], the so-called similarity radius of individuals, which determines the minimum distance in the solution space between individuals that are passed from generation to generation, is determined empirically. Most authors define it as a constant, or as a quantity that decreases evenly.

It is proposed to link the similarity radius with the offspring mutation radius in the current generation. Thus, the mutation operator will operate in the current solution neighborhood, and the population compression operator - outside it. This ensures diversity in each of the generations. The main operator, which is responsible for the search intensification, as in most implementations of the artificial immune systems method, continue to be the mutation operator in any case. At the same time, the population compression operator assumes the role of diversifier.

Solving continuous problems, it was found that the optimal value of the compression operator radius is getting greater meanwhile space dimension is increasing. Optimality should be understood as the speed of global optimum achieving while maintaining population diversity. The dependence of the compression radius on the space dimension in solving the Rastrigin problem is shown in Figure 7.

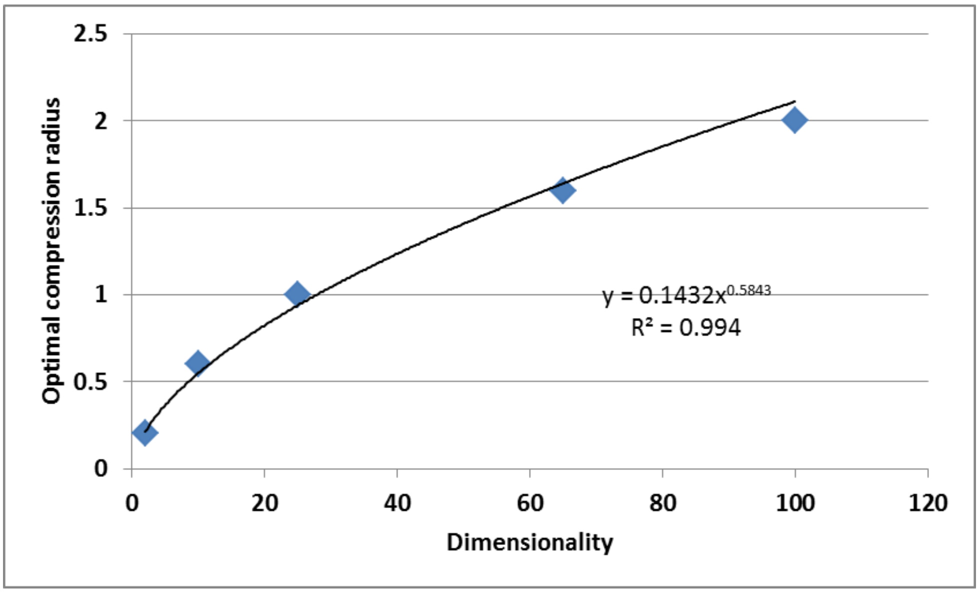

Figure 7: The optimal radius of the compression operator depending on the space dimension

Also, in Figure 7 it is clearly noticeably the best of several dozen regression models that describe the dependence of the compression operator radius on the space dimension. The power model with a power coefficient close to ½ is obtained. Therefore, the desired dependence can be assumed to have the form

$$R(N_{dim}) = a_0 \sqrt[2]{N_{dim}}, \tag{6}$$

where N_{dim} –problem dimension in continuous space ; a_0 – a constant term, depending on the problem type and must be obeyed empirically.

In addition to that, the tendency illustrated in Figure 6 was not refuted, but also received an extra development. In particular, scaling the knapsack problem and using different numbers of individuals in the population, it was found that the best results in terms of convergence and repeatability are provided by the use as a coefficient in (6) the value

$$a_0 = 1/N_{pop} \tag{7}$$

Thus, it is further recommended to set the radius in the population compression operator proportional to the radical of the space dimension and inversely proportional to the population individuals number.

4. Discussion

Based on the results of numerous experiments in solving known global optimization problems, the developed HINO-SF algorithm was investigated for its individual operators effectiveness. The following results were obtained:

• Applying of quasi-random generators instead of pseudo-random ones for first and new individuals generation during the algorithm reduces the global optimum searching time by 12-20%.

• Adopting of the modified selection operator, based on the Saaty`s paired comparison method, allows to improve the solution finding dynamics by 1.5-3 times, comparing with the previous realization and to achieve greater reliability than in case of the rank or tournament selection operators using.

• The global optimum finding becomes more reliable applying the crossover operator. The two-point crossover is the best for combinatorial problem, uniform crossover – for continuous problems relatively.

• The proposed adaptive mutation operator adjusts the narrow and wide mutations ratio, depending on the clone fitness and the iteration number. As soon as the space dimensionality increases, the proportion of wide mutations should be increased.

• The existing one-dimensional local search operator is considered to be ineffective and should be replaced. Applying one of the quasi-Newtonian methods instead of the zero-order methods might be promising.

• The population compression operator should be adaptive, considering the mutation radius at the current iteration, population size and space dimension.

The work [11] rejects the significance of high-dimensional combinatorial problems solving, adopting heuristic algorithms.

The search for the global optimum and several local optimal solutions close to global one, which can be used as an effective alternative, is an essential scientific and technical problem. Its solution, using a developed and investigated optimization algorithm based on an artificial immune system modeling might make specific social and economic impact.

5. Conclusions

The article includes the investigation of the global optimization algorithm based on artificial immune systems modeling and the most effective settings of its individual operators. For the first time, an attempt of finding a universal approach to solving both combinatorial and continuous problems applying the proposed algorithm was implemented, using a common approach to the algorithm operators.

The most efficient settings, based on the minimum solution search time criteria, are recommended for search agent generation, parental individuals selection, mutation, crossover, local search and population compression operators.

A significant reduction in the number of objective function evaluations and the algorithm performance time was achieved. It allowed the proposed algorithm to become competitive towards other widely used global optimization methods. The proposed settings of the algorithm operators allow to reduce the optimization time by 2-4 times without losing accuracy for some solving problems.

The set of several test functions of various dimensions in continuous and binary spaces was used for the algorithm testing, which demonstrated in improved time criteria performance. It is reasonable to outline that the greatest benefits from algorithm settings usage can be obtained in large dimension problems.

The practical value of the work is to make possible the developed algorithm applying for global optimization problems solving, especially in case of a high dimension mathematical model of a problem. In particular, the approach can be applied to solve planning and control problems in metallurgical production to optimize technological processes.

The direction of further research is the best local search operator selection as well as the adaptation mechanism formation of the mutation operator as the main search element of the AIS optimization algorithm. Other essential research direction is the formation of an adaptive population compression operator, which would help to discover suboptimal solutions.

6. Acknowledgements

This publication was prepared as a part of the scientific theme 0121U109788 "Problems of modeling, optimization, and decision making in complex systems of different nature", which is implemented on the System Analysis and Control Department at Dnipro University of Technology.

7. References

[1] D. Dasgupta, (1998). Artificial Immune Systems and Their Applications. DOI:10.1007/978-3-642-59901-9.

[2] R.M. Rasli, N.A.A. Aziz, F.M. Razali, N.M. Norwawi & N. Basir, “A Preliminary Survey on Artificial Immune Systems (AIS)”. International Journal of Academic Research in Business and Social Sciences. V. 9 I. 14 (2019): 121–144. DOI:10.6007/IJARBSS/v9-i14/6835

[3] Sh. Gao “A Multi-Learning Immune Algorithm for Numerical Optimization”. IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences. E98-A (2015): 362-377. DOI:10.1587/transfun.E98.A.362.

[4] Y. Wang, T. Li “Local feature selection based on artificial immune system for classification”. Applied Soft Computing. V. 87 (2020): 105989, DOI: 10.1016/j.asoc.2019.105989.

[5] J. Gao, G. He, R. Liang & Zh. Feng, “A quantum-inspired artificial immune system for the multiobjective 0–1 knapsack problem”. Applied Mathematics and Computation. V. 230, (2014): 120-137, https://doi.org/10.1016/j.amc.2013.12.088

[6] Y. Peng, B.-L. Lu “Hybrid learning clonal selection algorithm”. Information Sciences, V. 296 (2015): 128-146, DOI: 10.1016/j.ins.2014.10.056.

[7] W. Li, GG. Wang, & A.H. Gandomi “A Survey of Learning-Based Intelligent Optimization Algorithms”. Arch Computat Methods Eng (2021). DOI: 10.1007/s11831-021-09562-1

[8] S.E. Avramenko, T.A. Zheldak, L.S. Koriashkina “Guided hybrid genetic algorithm for solving global optimization problems”. Radio Electronics, Computer Science, Control. 2 (2021): 174-188. DOI:10.15588/1607-3274-2021-2-18.

[9] Weiwei Zhang, Weizheng Zhang, Gary G. Yen & HongLei Jing “A cluster-based clonal selection algorithm for optimization in dynamic environment”, Swarm and Evolutionary Computation, 50 (2019): 100454, DOI: 10.1016/j.swevo.2018.10.005.

[10] T.A. Zheldak, V.V. Slesarev & I.G. Gulina “The algorithm of artificial immune system simulation with Saaty selection operator and one-dimensional local search”. Naukovyi Visnyk Natsionalnoho Hirnychoho Universytetu. 5 (2016): 149-156.

[11] R.Q. Wang, L. Jiang, Y.D. Wang & A.P. Roskilly, “Energy saving technologies and mass-thermal network optimization for decarbonized iron and steel industry: A review”, Journal of Cleaner Production. V. 274. (2020): 122997, https://doi.org/10.1016/j.jclepro.2020.122997

[12] C-M. Pintea, Advances in Bio-inspired Computing for Combinatorial Optimization Problem. Springer. Intelligent Systems Reference Library. 57 (2014). DOI:10.1007/978-3-642-40179-4

[13] R. Rodrigues “A Simple Hack to Boost any Global Optimization Algorithm”, Towards Data Science. (2021). URL: https://towardsdatascience.com/a-simple-hack-to-boost-any-global-optimization-algorithm-bdea461b87

[14] H. Maaranen, K. Miettinen & M.M. Makela “Quasi-Random Initial Population for Genetic Algorithms”, Computers and Mathematics with Applications. 47. (2004) 1885-1895. DOI: 10.1016/j.camwa.2003.07.011

[15] OR-Library. The collection of test data sets for a variety of Operations Research (OR) problems. URL: http://people.brunel.ac.uk/~mastjjb/jeb/info.html

[16] O.V. Egorova & V.Ye. Snytuk “Bagatovymirna tekhnologiya spryamovanoi optymizacii” [Multidimensional technology of directed optimization]. International Conference “Computational intelligence”. (2015): 88–95. (In Ukrainian).