УПРАВЛІННЯ У ТЕХНІЧНИХ СИСТЕМАХ

CONTROL IN TECHNICAL SYSTEMS

УПРАВЛЕНИЕ В ТЕХНИЧЕСКИХ СИСТЕМАХ

UDC 004.023

GUIDED HYBRID GENETIC ALGORITHM FOR SOLVING GLOBAL OPTIMIZATION PROBLEMS

Avramenko S. E. – Ms. Sc., Department of System Analysis and Control, Dnipro University of Technology, Machine Learning Engineer, COMPARUS.UA, Dnipro, Ukraine.

Zheldak T. A. – PhD, Head of the Department of System Analysis and Control, Dnipro University of Technology, Dnipro, Ukraine.

Koriashkina L. S. – PhD, Associate Professor of the Department of System Analysis and Control, Dnipro University of Technology, Dnipro, Ukraine.

ABSTRACT

Context. One of the leading problems in the world of artificial intelligence is the optimization of complex systems, which is often represented as a nonlinear function that needs to be minimized. Such functions can be multimodal, non-differentiable, and even set as a black box. Building effective methods for solving global optimization problems raises great interest among scientists.

Objective. Development of a new hybrid genetic algorithm for solving global optimization problems, which is faster than existing analogues.

Methods. One of the crucial challenges for hybrid methods in solving nonlinear global optimization problems is the rational use of local search, as its application is accompanied by quite expensive computational costs. This paper proposes a new GBOHGA hybrid genetic algorithm that reproduces guided local search and combines two successful modifications of genetic algorithms. The first one is BOHGA that establishes a qualitative balance between local and global search. The second one is HGDN that prevents re-exploration of the previously explored areas of a search space. In addition, a modified bump-function and an adaptive scheme for determining one of its parameters – the radius of the "deflation" of the objective function in the vicinity of the already found local minimum – were presented to accelerate the algorithm.

Results. GBOHGA performance compared to other known stochastic search heuristics on a set of 33 test functions in 5 and 25-dimensional spaces. The results of computational experiments indicate the competitiveness of GBOHGA, especially in problems with multimodal functions and a large number of variables.

Conclusions. The new GBOHGA hybrid algorithm, developed on the basis of the integration of guided local search ideas and BOHGA and HGDN algorithms, allows to save significant computing resources and speed up the solution process of the global optimization problem. It should be used to solve global optimization problems that arise in engineering design, solving organizational and management problems, especially when the mathematical model of the problem is complex and multidimensional.

KEYWORDS: nonlinear optimization, global minimum, randomized search heuristics, hybrid approach, genetic algorithm, deflation operator, guided local search.

ABBREVIATIONS

GA is a genetic algorithm;

BFGS is a Broyden-Fletcher-Goldfarb-Shanno algorithm;

L-BFGS is a modification of the Broden-Fletcher-Goldfarb-Shanno algorithm (with limited memory);

L-BFGS-B is an extended modification of the Broden-Fletcher-Goldfarb-Shanno algorithm;

BOHGA is a best offspring hybrid genetic algorithm;

PSO is a particle swarm optimization algorithm;

HGDN is a hybrid genetic deflated Newton method;

GBOHGA is a guided best offspring hybrid genetic algorithm;

BH is basin-hopping, a technique for global optimization;

DA is a dual annealing algorithm;

DE is a differential evolution, a population-based metaheuristic search algorithm.

NOMENCLATURE

$f(\mathbf{x})$ is an objective function;

$\nabla f(\mathbf{x})$ is a gradient of a function $f(\mathbf{x})$;

$H(\mathbf{x})$ is a Hessian matrix $f(\mathbf{x})$;

DOI 10.15588/1607-3274-2021-2-18

$\mathbf{x}^*$ is a global optimum of a function $f(\mathbf{x})$;

$\mathbf{x} = (x_1, x_2, ..., x_n)$ is a point of n-dimensional real space, current position;

γ is an improvement from the current positions to the next;

$\mathbf{x}_0^k$, $k = \overline{1, K}$ are deflation points of the function, local optimums found at the current moment of the algorithm;

r is a radius around the deflation point;

r_k is an adjustable radius around the deflation point;

$b_{\mathbf{x}_0}(\mathbf{x})$ is a bump-function (a smooth function with compact support);

$\lambda \in (1,2]$ is a parameter of the GBOHGA;

α is a coefficient for adjusting the shape of the bump function.

INTRODUCTION

Today in the world of artificial intelligence one of the leading problems is the optimization of complex systems, which are reduced to many socio-economic, technical, organizational and managerial tasks [1, 2]. It is often presented as a nonlinear function that needs to be optimized. Problems of nonlinear optimization also arise in engineering design and in the fundamental sciences, such as in physics, chemistry, molecular biology, and others [3, 4]. Numerical solutions to such problems can be associated with significant difficulties because the objective function can be multi-extreme, non-differentiable and, in general, given in the form of a black box. In addition, each calculation of a function may require significant computational resources.

Unfortunately, there is no universal method for determining the global solution of the nonlinear optimization problem in the general formulation. Most traditional optimization methods are deterministic and local. They are often unable to leave the areas of attraction of local optimums. The use of the found local solutions may be insufficient because the global optimum can give a significant advantage over local ones. But there is an alternative approach – the use of evolutionary methods of global optimization and the deliberate introduction of an element of randomness into the search algorithm. These methods are based on the natural behavior of biological and physical systems and are able to overcome the shortcomings of traditional methods of "stuck" in the local optimum [5]. In addition, randomness can serve the purpose of collecting information about the behavior of the object of study and management objectives. The advantages of such methods are considered to be their increased speed; low sensitivity to irregular behavior of the objective function and the presence of random errors in calculating the function; relatively simple internal implementation; low sensitivity to the growing dimensionality of the optimization problem; the possibility of natural introduction into the search process of learning and self-learning. Besides, within the known random search schemes, new algorithms are easily built that implement various heuristic adaptation procedures.

A popular representative of evolutionary strategies is a genetic algorithm. In its classical form, it successfully explores the search space but has a slow convergence. To get rid of this shortcoming developed and widely used in the practice implementations that combine genetic algorithm and methods that in the process of their work use local information about promising areas of search. Such methods are called hybrid. One of the important issues in them is the balance between their components, and one of the well-known problems is the repeated exploration of the same area with the repeated finding of the already known local solution. Therefore, the development and research of new effective methods and algorithms for finding the global optimum of a multi extreme function continues.

This paper presents a new hybrid genetic algorithm GBOHGA for solving the global optimization problems, which integrates methods that can solve the above problems. One way to balance the components of the algorithm is to manage the proportion of the population that should perform a local search. For this purpose, the scheme laid down in the BOHGA method was used [6]. To prevent re-exploration of previously visited areas, the idea of "deflation" of the function from HGDN [7] was borrowed to "remove" already found solutions, but it is proposed to use it to the objective function, which is ideologically similar to guided local search [8] from combinatorial optimization. In addition, a new adaptive scheme is proposed to establish the size of the radius of the "deflation" area in order to avoid the unsuccessful selection of this parameter. As will be shown below, all this provides significant savings in computing resources and speeds up the algorithm, as it prevents the unnecessary use of local search.

The object of study is the problems of global optimization.

The subject of study is local search methods and randomized search heuristics for solving global optimization problems, their computational characteristics, and conditions of application.

The purpose of the work is to develop a new hybrid genetic algorithm for solving the problem of global optimization, devoid of the shortcomings of existing analogs.

1 PROBLEM STATEMENT

We will consider the problem of optimization in the next statement:

$$f^* = \min_{x \in \Omega} f(x), \quad (1)$$

where the feasible set is n-dimensional parallelepiped

$$\Omega = \{x \in R^n, a_i \le x_i \le b_i, i = \overline{1, n}\}. \quad (2)$$

Since the aim of the research is to develop an algorithm for solving the problem (1), (2) for the widest pos-

DOI 10.15588/1607-3274-2021-2-18

sible class of objective functions, we will assume that the function $f(x)$ is continuous, but can be multimodal, nonlinear, non-differentiable, have a ravine relief or high computational complexity.

2 REVIEW OF THE LITERATURE

The variety of global optimization problems entails a variety of approaches to their solution [1, 9]. An extensive overview of such problems and methods is given in [10]. Many studies are devoted to the study of the peculiarities of the use of methods for specific problems of global optimization. Let us recall only some of the known approaches, and review the latest developments in this field, without claiming to be complete.

There are so-called passive (non-adaptive) algorithms, in which the calculation points of the objective function are selected before the search from some a priori ideas about the problem and can not be changed during the operation of the algorithm. This approach usually requires a very large amount of computation to obtain a reliable estimate of the global solution. When using sequential (adaptive) schemes for placing the calculation points of the function, both a priori and information about the problem obtained during the search process are taken into account. Adaptive methods are generally characterized by a denser placement of the points of calculation of the function in the vicinity of the solution and rarer outside of it. Consistent are, for example, branch-and-bound algorithm, Bayesian methods, algorithms based on interval arithmetic, methods of multiple local descent, and many others [10]. Within the framework of interval analysis, the technique of multiple partitioning is intensively developed, as well as more complex partitioning strategies based on simplexes [11] and various auxiliary functions [12]. Each of the techniques is common enough to solve different classes of global optimization problems.

For many practical problems (solving systems of nonlinear equations and inequalities, optimization of hierarchical models associated with placement problems, service systems, etc.), the assumption that the functions characterizing the system of study are Lipschitz functions is typical. The development of the theory and methods of numerical solution of problems of this type is the subject of Lipschitz global optimization [13, 14].

Special mention should be made of adaptive stochastic search strategies, many of which underlie heuristic global optimization algorithms (simulated annealing methods, genetic algorithms, tabu search methods, and others). Different strategies for approaching and relaxing the original problem are also used, which involve the consistent construction of auxiliary subtasks: their solution allows to improve the estimation of the global optimum [15–17]. In [15, 16] methods and algorithms for solving optimization problems are studied, which do not use information about the derivatives of the objective function but require only knowledge of its values. In [15] a systematic comparison of existing implementations of such algorithms on a large set of test problems, convex and nonconvex, smooth and non-smooth is presented. The authors single out global solvers that are superior to local ones, even for convex problems, and demonstrate excellent performance in terms of improving suboptimal solutions.

Geometric and information bases for the construction of global optimization algorithms are considered in [16].

Here are some new ideas for speeding up the search so that accelerated optimization methods automatically implement local behavior in promising regions and do not stop the search process of a global minimum. The authors emphasize that a reasonable combination of new and traditional computational steps leads to two dozen different global optimization algorithms. All of them are studied and numerically compared on three test sets of functions and applied problems.

For costly, in terms of computing resources, black-box problems surrogate models are now widely used to reduce computation time and effort in finding the global optimum. Thus, in [17] a new global optimization method is proposed, which uses an ensemble of surrogates and reduction of hierarchical design. The search uses an ensemble of three representative surrogate methods with optimized weights to select promising sampling points, narrow the study space, and determine the global optimum. The new proposed global optimization method tested using eighteen representative functions and two engineering design optimization problems, demonstrated improved capabilities in identifying promising search areas and reducing design space, as well as excellent search efficiency and reliability in determining global optimum. Fundamental problems that arise during global optimization based on surrogates from the point of view of the practitioner, including the selection of concepts, methods, techniques, and engineering applications, are discussed in [18]. Providing a comprehensive discussion of the issues raised, the paper explores the latest achievements in the planning of experiments, methods of surrogate modeling, criteria for filling and reducing the project space.

New sampling points are needed to increase the accuracy of the surrogate model forecast. The decision-making strategies for the following promising samples are called the filling criteria, the so-called adaptive sampling methods. The filling criteria (which are conventionally classified as exploitation, exploration, combined exploitation, and research) may well guide the selection of new sampling points depending on the information from the optimization process to be used. On the one hand, operating methods may focus on regions located in the area of the best point that has been found so far, but which may not even be stationary for the function. This can only lead to a local approximation and a local optimum. On the other hand, exploration and combined exploitation usually explore rare regions or regions with high uncertainty. However, only the use of exploration strategy can lead to the depletion of computational resources when approaching the global minimum with a given accuracy. In general, high accuracy is needed in potentially promising regions. Therefore, the authors [19] propose to combine exploitation and exploration to balance competing goals between lower cost and more accurate assessment of the optimal solution. The work focuses on parallel algorithms of effective global optimization. Here, a multi-point fill criterion is developed, which uses entropy to accurately measure the uncertainty of the Kriging surrogate, and then balances global exploration and the local exploitation of multi-point selection criteria. A strategy for optimizing the decomposition of domains, which provides a small amount of training

DOI 10.15588/1607-3274-2021-2-18

data, is also proposed. According to the authors, compared to several other methods, it has an obvious advantage in solving complex optimization problems in a large-scale parallel computing environment.

Bayesian optimization, which is used to search for the global minimum of computationally expensive black-box functions when the evaluation of functions is costly and the optimum can be achieved with a relatively small number of evaluations, is becoming increasingly popular today. However, its application for multidimensional problems with several thousand observations remains a difficult task, and the improvement of appropriate methods and approaches is an important area of modern research. Thus, [20] proposed ensemble Bayesian optimization to simultaneously solve the problems of large-scale observations, large dimensionality of the input space, and the choice of batch requests that balance quality and diversity. In [21] the authors proposed the TuRBO algorithm based on a collection of local probability models that provide local search trajectories that can be used to quickly identify the best values of the objective function. This approach is complemented by a so-called gangster strategy, which distributes samples across trust regions, implicitly trading exploration and exploitation.

Therefore, deterministic methods have an advantage in the guaranteed quality of the solution found, but the computational complexity of such methods is exponential in the worst case. In fact, no method is better for the worst case of a function than passive grid search. Because optimization problems usually come from an arbitrary source and therefore tend to have unknown statistical characteristics, alternative approaches to solving such problems, including adaptive search strategies and statistical modeling tools, are becoming important tools, especially in the case of high-dimensional problems.

Significant results show evolutionary strategies that mimic biological and social models of evolution. Different deterministic and stochastic algorithms can be built on the basis of different evolutionary rules. This is evidenced, for example, by the successful application of the genetic algorithm to real problems in a wide range of industries [22]. Although metaheuristics are considered reliable, their convergence slows down when solving complex problems, because they do not involve the use of local information about promising areas of search. To combat this defect, the idea of combining local search with various metaheuristic algorithms is being increasingly introduced.

The following is a brief description of the genetic algorithm, together with its modifications and methods of constructing hybrids based on it.

3 MATERIALS AND METHODS

The genetic algorithm is an iterative optimization method based on the concept of natural selection known in genetics. Variables that characterize the solution are represented as a set of genes on a chromosome. For example, a point in n-dimensional space can be encoded by a characteristic vector $x=(x_1,...,x_n)$. GA operates with a finite set of solutions (population), generates new solutions as different combinations of parts of others, using operators such as selection, crossover and mutation [22]. The new solutions are positioned in the population according to their position on the surface of the studied function.

At the beginning of the algorithm, initialization is performed – the formation of the initial population randomly. Next, the selection of chromosomes is done, which consists of selecting those that will produce offspring for the next population. Genetic operators that form a new population are applied to the selected chromosomes. Crossover is a key operator based on the assumed high probability that the new solution obtained from two sufficiently good solutions of the problem will be good or even better than the previous ones. An important but still secondary role is played by the mutation operator, which introduces new genetic material into the population to maintain population diversity and prevents losses that could occur due to the exclusion of any significant gene as a result of crossbreeding.

The steps described above are repeated many times until the algorithm's stop criterion is met, for example, when the already achieved value of the objective function does not improve for some time. Or the operation of the algorithm can be strictly limited by a predetermined duration or number of iterations.

In order not to waste significant computing resources when solving complex problems, after crossover and mutation, an additional operator is used. It performs a local search, starting in a certain area of the promising area. Such algorithms are called hybrid. Methods and algorithms of local search most often find the nearest extremum, and the trajectory of their movement significantly depends on the choice of the starting point and the nature of the target function.

The choice of local search method for inclusion in the hybrid should be made taking into account the benefits and costs of using each of the approaches. Genetic algorithm operators also partially perform the role of local search with relatively low computational cost compared to more accurate local search methods. Therefore, it is rational to limit the usage of a local search operator. A specific scheme for determining individuals from the population to whom it is desirable to apply local search is offered by the BOHGA [6]. The algorithm's scheme proposes to perform a local search only when the best representative of the offspring population is also the best in the current parent population. When such best offspring appear, it is very likely that such offspring will explore a new area, and therefore local knowledge will be used to accelerate the search for the most promising region.

However, in the described scheme, there can still be a reapplication of local search to individuals that fall into the same area of gravity of the search space, so in order to prevent unnecessary computational costs in [7], a deflated Newton modification was proposed that is a key idea of the HGDN method. This hybrid uses a genetic algorithm as a global search method and deflated Newton scheme as a local search operator. This modification of Newton's method effectively identifies several local optimums in the immediate vicinity of the starting point and accordingly changes the function. Here is the main idea of the method.

It is known that Newton's method calculates the gradient and Hessian of the objective function at the current point and uses this information to predict the new location of the solution by solving the equation $H(\mathbf{x})\boldsymbol{\gamma}=-\nabla f(\mathbf{x})$. The deflated Newton scheme is that Newton's algorithm looks for station-

© Avramenko S. E., Zheldak T. A., Koriashkina L. S., 2021
DOI 10.15588/1607-3274-2021-2-18

ary points of function $f(\mathbf{x})$. Let $\mathbf{x}_0^1, \mathbf{x}_0^2, ..., \mathbf{x}_0^K$ – be the stationary points defined at the moment. Then other such points can be found by applying the deflation of the gradient of the function f by the formula:

$$\nabla f_{\mathbf{x}_0}(\mathbf{x}) = \frac{\nabla f(\mathbf{x})}{\prod_{k=1}^{K} \left\| \mathbf{x} - \mathbf{x}_0^k \right\|^2}. \tag{3}$$

The deflated function has no roots in the found optimums, and therefore, the local search method will no longer be able to converge to these points. The principle of operation of this approach is shown in Fig. 1 [7]. Thus, in Fig. 1, a the genetic algorithm applies the local search operator to each individual of the population, and in the next iteration, the same optimum is identified again. To combat this behavior, formula (3) is used to calculate the gradient of the function, i.e. the roots are removed (Fig. 1,b), and therefore the already found optimums cannot be identified again, and each individual can find new optimums (Fig. 1, c).

Since the proposed deflation formula is often unreliable in the computational sense, alternative deflation operators are being developed [7, 23].

The localized operator proposed in [7], in order to affect only the area close to the deflation, uses the bump-function (smooth function with compact support) in the following form:

$$b_{\mathbf{x}_0}(x) = \begin{cases} \prod_{i=1}^{n} \dfrac{\exp\left(\dfrac{-\alpha}{r^2 - (x_i - x_{0i})^2} \right)}{\exp\left(\dfrac{-\alpha}{r^2} \right)}, & \text{if } x_{0i} - r < x_i < x_{0i} + r, \\ 0 & \text{else,} \end{cases}$$

with $\mathbf{x}_0$ as a deflation point. And the deflated function itself has the form:

$$\nabla f_{\mathbf{x}_0}(\mathbf{x}) = \frac{\nabla f(\mathbf{x})}{1 - b_{\mathbf{x}_0}(\mathbf{x})}.$$

We will build a new global optimization algorithm that combines the above-described BOHGA and HGDN hybrids, as both seek to reduce the number of function evaluations by eliminating the useless use of the local search operator to individuals who are either in non-promising regions or in the same one.

In addition, we develop the idea of a guided local search algorithm [8], which has successfully proven itself in combinatorial optimization [24–26]. The latter was proposed by Voudouris and Tsang and is a meta-heuristic method that during the search process constructs penalties and uses them to help local search algorithms get out of the minimum already found, and even from almost horizontal areas.

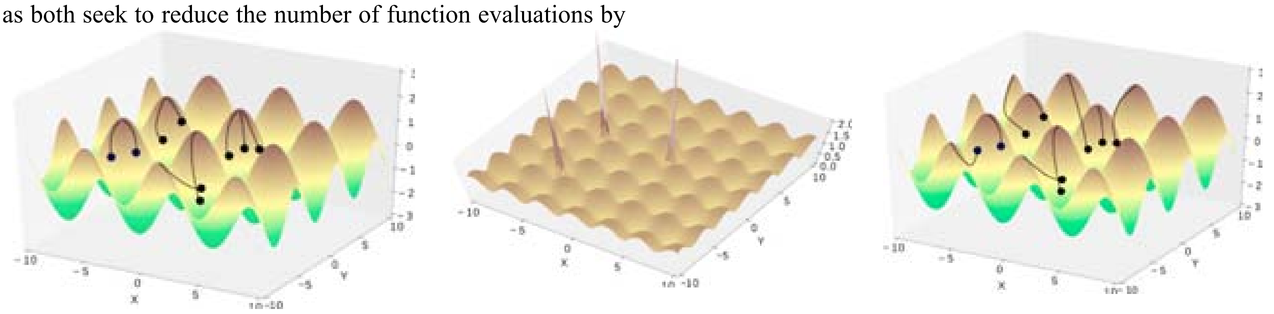
As soon as the search algorithm falls within the local minimum, the objective function is modified in a certain way, and the local search works with this modified function.

In the new algorithm GBOHGA to modify the objective function in the areas of attraction of already found local extrema, we will use the bump-function introduced in HGDN. That is, the function $f(\mathbf{x})$ in these areas is replaced by the following:

$$f_{\mathbf{x}_0}(\mathbf{x}) = \frac{f(\mathbf{x})}{1 - b_{\mathbf{x}_0}(\mathbf{x})}. \tag{5}$$

Thus, in this way, the algorithm forces the individual to look for a new optimum elsewhere, preventing reconvergence to the point $\mathbf{x}_0$. However, with the long-term operation of an algorithm with a highly multimodal function, the set of local optimums found can become very large, which affects both the amount of memory consumed and computational costs. Besides, in practice, there is a question of choosing a value for the parameter r that is responsible for the size of the radius of the deflation point of the previously found optimum. If you choose too small a value, its positive effect on the algorithm will not be noticeable, in addition, it can lead to a significant number of local optimums in the same area of attraction of the search space. On the contrary, choosing too large a value of this parameter, we assume that the local search algorithm qualitatively explored this region and provided the best possible solution from it.

Although, of course, it may happen that this relatively small area itself is very complex, and, therefore, such a choice of the radius of deflation can cause premature convergence of the algorithm.



a b c

Figure 1 – Demonstration of deflated Newton's method [7]: a – the hybrid method applies local optimization to each individual. In the next iteration, the same optima can be identified again; b – the gradient of the function is deflated at the optima of the function; c – HGDN removes any identified optima after each iteration. Therefore, already found optima cannot be identified again

© Avramenko S. E., Zheldak T. A., Koriashkina L. S., 2021
DOI 10.15588/1607-3274-2021-2-18

To overcome this uncertainty, the new algorithm proposes to use an adaptive scheme for selecting the value of the parameter r, that responds to the frequency of falling into the area of attraction of the local optimum. Given the type of objective function, for each found point $\mathbf{x}_0$ of the local optimum, a certain rather small initial value r_0 is choosen. Next, in the case of repeated convergence to the previously found optimum at a distance less than or equal to λr_i, from two points of the optimum – new x_0^k and x_0^i save the one that corresponds to the smaller value of the function, with radius $r = \lambda r_i$. The bump-function has the form:

$$b_{\mathbf{x}_0^k}(x) = \begin{cases} \prod\limits_{i=1}^{n} \dfrac{\exp\left(\dfrac{-\alpha}{r_k^2 - (x_i - x_{0i}^k)^2}\right)}{\exp\left(\dfrac{-\alpha}{r_k^2}\right)}, \text{ if } \left\| x - x_0^k \right\| < r_k, \\ 0 \quad \text{else.} \end{cases} \quad (6)$$

In order not to perform a costly operation of local search in those areas that are likely to be of no use, we will use the BOHGA scheme of a selection of individuals to which the deflation operator (5), (6) is applied with an adaptive choice of parameter r. Thus, the scheme of the GBOHGA is as follows.

GBOHGA algorithm

1. The operators of the genetic algorithm are determined; parameters r_0, α, λ are set.

2. The initial population is randomly generated.

3. Evaluation of the objective function for each individual of the initial population is performed.

4. A population of offspring is created using genetic algorithm operators.

5. The objective function of each individual of the population of offspring according to formulas (5), (6) is estimated.

6. It is determined whether the best offspring individual among the offspring population is also the best among the current parent population.

6.1 If not, go to step 7 (local search is not performed).

6.2 In the case of such a descendant, the local search operation is applied only to it and the suitability of the new solution x_0^k is estimated.

6.3 It is checked whether the new solution is similar to any previously found. That is, whether the new solution x_0^k is located at a distance less than λr_i, from the i-th previously found local optimum.

6.3.1 If not, then a new solution x_0^k with a radius r_k is added to the set of local optimums $\left\{\left(x_0^1, r_1\right), \left(x_0^2, r_2\right), ..., \left(x_0^{k-1}, r_{k-1}\right)\right\}$.

6.3.2 Otherwise, from the two points of the local optimum – the new x_0^k and x_0^i – we keep the one that corresponds to the smaller value of the function, with a radius $r = \lambda r_i$.

7. It is decided which individuals to include in the next population and which parents should also be left in the new generation according to the elitist approach.

8. If the stop criterion is met, the procedure stops, otherwise, the transition to step 4 is performed.

4 EXPERIMENTS

To test and compare the efficiency of the developed and existing, the best for today, optimization algorithms among the known modern artificial landscapes [27] selected 33 functions with different properties in terms of modality, separability, scalability, and landscape of the valley. The choice of functions for testing is justified by the following considerations.

Multimodal functions with many local minima form one of the most complex classes of optimization problems for many algorithms and are used to test the ability of the algorithm to proceed from any local minimum. If the algorithm has a poorly designed research process, it cannot effectively investigate the landscape of functions and is stuck at the local minimum. Also for algorithms, functions with flat surfaces can cause difficulties, because their invariance does not give the algorithm any information to direct the search process to the minima (for example, Stepint or Powell Sum).

Among the test functions, there are separable and non-separable. Usually, it is much easier to optimize the former, since each variable of such a function is independent of the others, it is possible to perform a sequence of n independent optimization processes to find the optimal value of the respective variables.

An important indicator of the effectiveness of optimization algorithms is the ability to overcome the problem of dimensionality because it is known that with the increasing number of parameters, the search space also increases, and exponentially. Therefore, the set of test functions includes functions with the ability to increase the number of their components.

For some functions, the area containing global minima is very small compared to the whole search space, for example, Powell or Schaffer problems. For the last problem, the global minimum is very close to the local minima. And if the algorithm does not have time to change direction in functions with a narrow curved valley, or inefficiently explores the search space, in the case where the functions have several global minima, such an algorithm will not be able to cope with these problems. The choice of test functions is made taking into account the variety of the functions' hypersurfaces [28].

Table 1 contains a general description of a set of selected artificial landscapes listed above. The list of all test functions used in the work with their mathematical notation is given in the Table 2, and in Fig. 2 are presented the landscapes of some of them.

DOI 10.15588/1607-3274-2021-2-18

Table 1 – General characteristics of a set of artificial landscapes

A property that is typical for a function from the set of artificial landscapes	Percentage of functions in the set
continuity	91
differentiability	73
separability	36
unimodality	58
convexity	42

Table 2 – Artificial landscapes

N	Name, mathematical notation, optimal solution	N	Name, mathematical notation, optimal solution
1	Ackley 1 $f(x)=-20e^{-0.02\sqrt{n^{-1}\sum_{i=1}^{n}x_i^2}}-e^{n^{-1}\sum_{i=1}^{n}\cos(2\pi x_i)}+20+e$ $-35\le x_i\le 35$, $x^*=(0,0,...,0)$, $f(x^*)=0$	11	Schwefel 2.21 $f(x)=\max_{1\le i\le n}\lvert x_i\rvert$, $-100\le x_i\le 100$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
2	Alpine 1 $f(x)=\sum_{i=1}^{n}\lvert x_i\sin(x_i)+0.1x_i\rvert$, $-10\le x_i\le 10$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$	12	Schwefel 2.22 $f(x)=\sum_{i=1}^{n}\lvert x_i\rvert+\prod_{i=1}^{n}\lvert x_i\rvert$, $-100\le x_i\le 100$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
3	Chung Reynolds $f(x)=\left(\sum_{i=1}^{n}x_i^2\right)^2$, $-100\le x_i\le 100$, $x^*=(0,0,...,0)$, $f(x^*)=0$	13	Schwefel 2.23 $f(x)=\sum_{i=1}^{n}x_i^{10}$, $-10\le x_i\le 10$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
4	Exponential $f(x)=-\exp\left(-0.5\sum_{i=1}^{n}x_i^2\right)$, $-1\le x_i\le 1$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=1$	14	Sphere $f(x)=\sum_{i=1}^{n}x_i^2$, $0\le x_i\le 10$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
5	Griewank $f(x)=\sum_{i=1}^{n}\frac{x_i^2}{4000}-\prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$, $-100\le x_i\le 100$, $x^*=(0,0,...,0)$, $f(x^*)=0$	15	Step $f(x)=\sum_{i=1}^{n}\left(\lfloor\lvert x_i\rvert\rfloor\right)$, $-100\le x_i\le 100$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
6	Hapy Cat $f(x)=\left[\left(\lVert x\rVert^2-n\right)^2\right]^{\alpha}+\frac{1}{n}\left(\frac{1}{2}\lVert x\rVert^2+\sum_{i=1}^{n}x_i\right)+\frac{1}{2}$, $-2\le x_i\le 2$, α – real; $x^*=(-1,-1,...,-1)$, $f(x^*)=0$	16	Step 2 $f(x)=\sum_{i=1}^{n}\left(\lfloor x_i+0.5\rfloor\right)^2$, $-100\le x_i\le 100$, $i=\overline{1,n}$, $x^*=(0.5,0.5,...,0.5)$, $f(x^*)=0$
7	Periodical $f(x)=1+\sum_{i=1}^{n}\sin^2(x_i)-0.1e^{\left(\sum_{i=1}^{n}x_i^2\right)}$, $-10\le x_i\le 10$, $x^*=(0,0,...,0)$, $f(x^*)=0.9$	17	Step 3 $f(x)=\sum_{i=1}^{n}\left(\lfloor x_i^2\rfloor\right)$, $-100\le x_i\le 100$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
8	Powell Sum $f(x)=\sum_{i=1}^{n}\lvert x_i\rvert^{i+1}$, $-1\le x_i\le 1$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$	18	Streched V Sine Wave $f(x)=\sum_{i=1}^{n-1}\left(x_{i+1}+x_i^2\right)^{\frac{1}{4}}\left[\sin^2\left\{50\left(x_{i+1}+x_i^2\right)^{0.1}\right\}+0.1\right]$, $-10\le x_i\le 10$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
9	Rastrigin $f(x)=10n+\sum_{i=1}^{n}\left(x_i^2-10\cos(2\pi x_i)\right)$, $-5.12\le x_i\le 5.12$, $i=\overline{1,n}$, $x^*=(0,...,0)$, $f(x^*)=0$	19	Sum Squares $f(x)=\sum_{i=1}^{n}ix_i^2$, $-10\le x_i\le 10$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$
10	Rosenbrock $f(x)=\sum_{i=1}^{n-1}\left[100\left(x_{i+1}-x_i^2\right)^2+\left(x_i-1\right)^2\right]$, $-30\le x_i\le 30$, $i=\overline{1,n}$, $x^*=(1,1,...,1)$, $f(x^*)=0$	20	Trigonometric 1 $f(x)=\sum_{i=1}^{n}\left[n-\sum_{j=1}^{n}\cos x_j+i(\cos(x_i)-\sin(x_i))\right]^2$ $0\le x_i\le\pi$, $i=\overline{1,n}$, $x^*=(0,0,...,0)$, $f(x^*)=0$

DOI 10.15588/1607-3274-2021-2-18

Continuation of the Table 2

21	Salomon $f(x)=1-\cos\left(2\pi\sqrt{\sum_{i=1}^{n}x_i^2}\right)+0.1\sqrt{\sum_{i=1}^{n}x_i^2}$, $-100\le x_i\le 100,\ \ i=\overline{1,n},\ \ x^*=(0,...,0),\ f(x^*)=0$	28	Trigonometric 2 $f(x)=1+\sum_{i=1}^{n}8\sin^2\left(7(x_i-0.9)^2\right)+6\sin^2\left[14(x_1-0.9)^2\right]+$ $+(x_i-0.9)^2,\ -500\le x_i\le 500,\ i=\overline{1,n}$ $x^*=(0.9,0.9,...,0.9),\ f(x^*)=1$
22	Sargan $f(x)=\sum_{i=1}^{n}n\left(x_i^2+0.4\sum_{j\ne 1}x_ix_j\right)$, $-100\le x_i\le 100,\ \ i=\overline{1,n},\ \ x^*=(0,...,0),\ f(x^*)=0$	29	W / Wavy Function $f(x)=1-\frac{1}{n}\sum_{i=1}^{n}\cos(kx_i)e^{-x_i^2/2}$, k – дійсне число $-\pi\le x_i\le\pi,\ \ i=\overline{1,n},\ \ x^*=(0,0,...,0),\ f(x^*)=0$
23	Schaffer F6 $f(x)=\sum_{i=1}^{n}\left(0.5+\frac{\sin^2\sqrt{x_i^2+x_{i+1}^2}-0.5}{\left[1+0.001(x_i^2+x_{i+1}^2)\right]^2}\right)$ $-100\le x_i\le 100,\ \ i=\overline{1,n},\ \ x^*=(0,...,0),\ f(x^*)=0$	30	Weierstrass $f(x)=\sum_{i=1}^{n}\left[\sum_{k=0}^{k_{\max}}a^k\cos(2\pi b^k(x_i+0.5))-n\sum_{k=0}^{k_{\max}}a^k\cos(\pi b^k)\right]$ $-0.5\le x_i\le 0.5,\ \ i=\overline{1,n},\ \ x^*=(0,0,...,0),\ f(x^*)=0$
24	Schumer Steiglitz $f(x)=\sum_{i=1}^{n}x_i^4$, $x^*=(0,0,...,0),\ f(x^*)=0$	31	Whitley $f(x)=\sum_{i=1}^{n}\sum_{j=1}^{n}\left[\frac{(100(x_i^2-x_j)^2+(1-x_j)^2)^2}{4000}-\right.$ $\left.-\cos(100(x_i^2-x_j)^2+(1-x_j)^2+1)\right]$ $x^*=(1,1,...,1),\ f(x^*)=0$
25	Schwefel 2.20 $f(x)=-\sum_{i=1}^{n}\lvert x_i\rvert$, $-100\le x_i\le 100,\ \ i=\overline{1,n}$, $x^*=(0,0,...,0),\ f(x^*)=0$		
26	Schwefel $f(x)=\left(\sum_{i=1}^{n}x_i^2\right)^{\alpha}$, $\alpha\ge 0$ – дійсний параметр $-100\le x_i\le 100,\ \ i=\overline{1,n}$, $x^*=(0,0,...,0),\ f(x^*)=0$	32	Xin-She Yang 3 $f(x)=e^{-\sum_{i=1}^{n}(x_i/\beta)^{2m}}-2e^{-\sum_{i=1}^{n}x_i^2}\cdot\prod_{i=1}^{n}\cos^2(x_i)$ $-20\le x_i\le 20,\ \ m=5,\beta=15$, $x_i^*=0\forall i,\ f(x^*)=-1$
27	Schwefel 1.2 $f(x)=\sum_{i=1}^{n}\left(\sum_{j=1}^{i}x_j^2\right)^2$, $-100\le x_i\le 100,\ \ i=\overline{1,n}$, $x^*=(0,0,...,0),\ f(x^*)=0$	33	Xin-She Yang (Function 4) $f(x)=(\cdot\sum_{i=1}^{n}\sin^2(x_i)-e^{-\sum_{i=1}^{n}x_i^2})e^{-\sum_{i=1}^{n}\sin^2\sqrt{\lvert x_i\rvert}}$, $-10\le x_i\le 10,\ \ i=\overline{1,n},\ \ x^*=(0,0,...,0),\ f(x^*)=-1$

For all the problems consider an objective function subject to bound constraints on the values of the variables. All functions are scalable, which allows us to check the behavior of algorithms in large-scale problems. Two-thirds of all test functions are not separable, which is also a challenge for most algorithms, but perhaps the most difficult is multimodality, especially in multidimensional spaces.

The testing was performed in comparison with other popular evolutionary algorithms, including the BH, DA, PSO, DE. For correct analysis, instead of independent implementation of these methods, their open-source versions from the SciPy library were used.

A key aspect of developing a new algorithm is the desire to reduce the required number of calculations of the objective function with an adequate level of reliability of the algorithm. Instead of counting the number of calls to the objective function, we can also use the physical time spent on solving problems, but this characteristic strongly depends on the quality of the implementation of the algorithm and the level of hardware, and therefore it is less universal.

In order to obtain reliable comparison results, the experiment was performed 50 times with random initial populations, and the results were averaged. Fig. 3 showed the performance profiles for BH, DA, DE, PSO, and GBOHGA on the test data set using function evaluations as performance criteria.

DOI 10.15588/1607-3274-2021-2-18

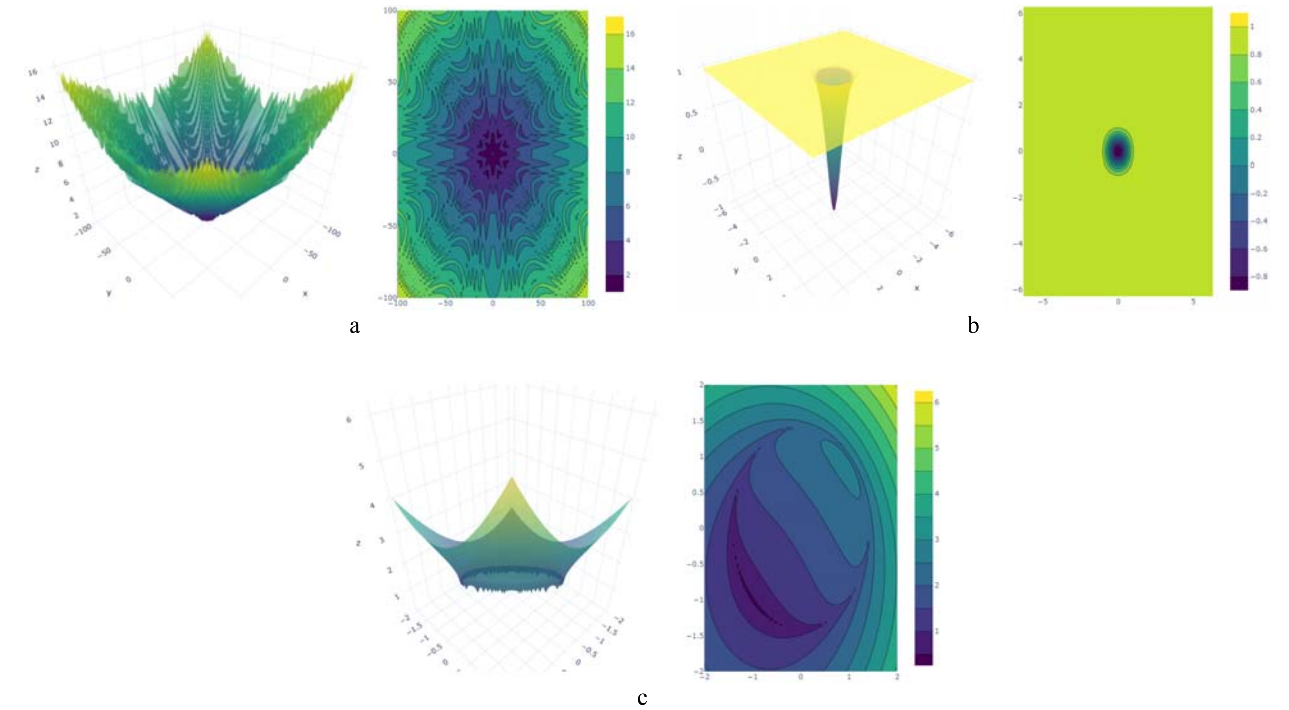


Figure 2 – 3d-Surface and Contour Map of some test functions:
a – Salomon Function; b – Xin-She Yang Function; c – Hapy Cat Function

5 RESULTS

Fig. 3 shows that for the test set of problems GBOHGA is the best algorithm in terms of performance, followed by DA and DE. The GBOHGA algorithm requires 3 times fewer evaluations of the objective function than DA and 6 times less than DE. At the same time, these algorithms tend to outperform GBOHGA, solving more problems by 5% and 6%, respectively. A detailed description of the results obtained in terms of each function is given separately in the Table 3. For each of the 5 optimization algorithms in the first column (μ) is given the average value of the number of calls to the objective function for each of the 33 test functions based on the results of 50 runs of the algorithm. The second column (σ), according to the same scheme, indicates the standard deviation, and the third (%) indicates the success level (values from 0.0 to 1.0) of the algorithm on this problem, which characterizes the robustness of the algorithm. The results of this table are also summarized for all functions and given in the last line. For greater clarity, in Fig. 4 a comparison of algorithms by their performance is demonstrated. Thus, GBOHGA is in 62.5% of cases faster than competing algorithms.

When solving optimization problems with 25 variables, the GBOHGA algorithm is still the best in terms of performance, and it can only compete with DA, which solves 3% more problems from the test suite, but consumes 5 times more computing resources (see Table 4 for more details). In terms of performance speed, GBOHGA shows even better results than in the previous experiment, namely in 69.7% of cases GBOHGA is faster than competing methods.

Obviously, with the increase in the number of variables, GBOHGA shows not worse, but even better results compared to other methods.

It should be noted that when designing the algorithm, separately, ten different crossover operators were compared, of which the best results were found by a discrete crossover. Also, five mutation operators and two selection operators were tested and it was shown that the best combination is the rank selection with the non-uniform mutation.

Further, in combination with the genetic algorithm, seven different methods of local optimization were used. According to the obtained results, the advantage is attributed to quasi-Newton methods (L-BFGS-B turned out to be the best option).

Therefore, the above-mentioned operators were chosen as the key components of the proposed hybrid algorithm, other parameters of the genetic algorithm were chosen arbitrarily, just as all competing algorithms were performed without prior optimization of parameters, ie the values were chosen either arbitrarily or most used.

DOI 10.15588/1607-3274-2021-2-18

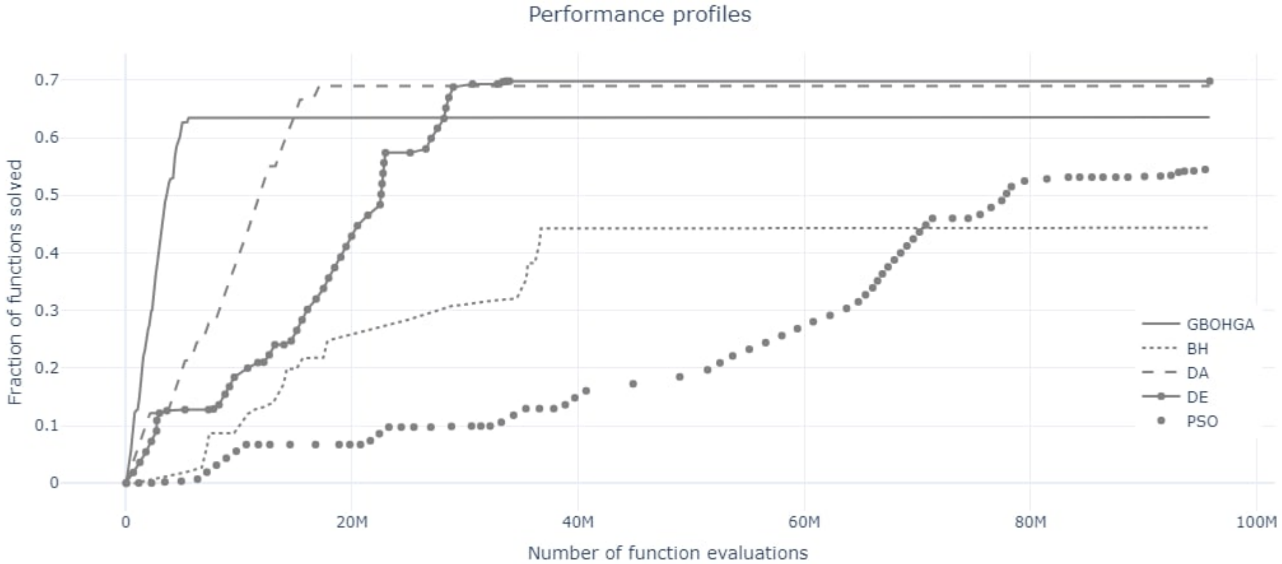


Figure 3 – Algorithms performance profiles

Table 3 – Mean (μ) and standard deviation (σ) of the number of function evaluations and success rate (%) on the results of 50 runs for each function from the test set for each method

Function name	BH			DA			DE			GBOHGA			PSO		
	μ	σ	%	μ	σ	%	μ	σ	%	μ	σ	%	μ	σ	%
Ackley1	14692	560	0.0	10701	131	1.0	20829	3097	0.96	4022	1050	0.76	53297	3732	0.0
Alpine1	119324	7602	0.88	11347	538	1.0	17643	1118	1.0	4602	627	1.0	72846	14714	0.16
Chung Reynolds	8712	154	1.0	10356	43	1.0	15525	554	1.0	2444	224	1.0	42601	2534	1.0
Exponential	3572	44	1.0	10032	5	1.0	968	97	1.0	2688	1931	1.0	43793	2147	1.0
Griewank	10105	305	0.0	10376	96	0.04	20303	5524	0.2	3184	884	0.36	54546	10131	0.0
HappyCat	27875	5363	0.0	10857	311	0.0	69822	8046	0.0	4054	1407	0.0	108498	6777	0.0
Periodic	5659	52	0.0	10038	6	0.0	14710	3839	0.04	2628	485	0.34	47473	2551	0.0
PowellSum	20654	470	1.0	10255	39	1.0	17677	536	1.0	2258	198	1.0	40994	3348	1.0
Rastrigin	15567	1078	0.48	10366	82	1.0	12508	2928	0.8	3587	692	1.0	54366	3089	0.0
Rosenbrock	31886	13068	0.34	10480	101	1.0	42053	9270	0.94	2838	424	1.0	92979	5267	0.02
Salomon	19196	5393	1.0	10453	124	0.02	15849	1596	0.0	3140	797	0.32	41447	2331	0.0
Sargan	5864	111	1.0	10057	7	1.0	16165	667	1.0	2394	274	1.0	53666	3147	1.0
SchafferF6	18194	3525	0.0	10860	255	0.06	26292	5890	0.0	3764	797	0.24	62897	14239	0.0
Schumer Steiglitz	12213	5943	0.76	10348	29	1.0	16150	625	1.0	2468	231	1.0	42139	2148	1.0
Schwefel	36485	1342	0.0	10811	233	0.0	15537	538	1.0	2841	287	0.0	204860	9943	1.0
Schwefel12	7494	52	1.0	10088	14	1.0	25597	837	1.0	2375	261	1.0	54120	3811	1.0
Schwefel220	125013	1957	1.0	11077	409	1.0	15243	548	1.0	3481	414	1.0	73152	5418	0.98
Schwefel221	93100	6074	1.0	11548	409	1.0	18376	698	1.0	4100	982	0.98	69960	4544	1.0
Schwefel222	115859	11836	0.46	12324	568	1.0	15709	511	1.0	4943	635	1.0	74771	8138	0.94
Schwefel223	16868	686	1.0	10450	41	1.0	16888	577	1.0	2393	148	1.0	32622	1430	1.0
Sphere	921	67	1.0	10013	0	1.0	36873	1075	1.0	3524	472	1.0	21472	307	1.0
Step	707	0	0.0	10091	16	1.0	2775	320	1.0	3835	786	0.46	27776	650	1.0
Step2	707	0	0.0	10093	17	1.0	2830	406	1.0	3556	756	0.6	27970	760	1.0
Step3	707	0	0.0	10101	20	1.0	2764	282	1.0	3958	705	0.56	27786	613	1.0
StrechedV SineWave	12512	3271	0.0	11485	513	0.0	70214	8716	0.0	4868	2056	0.02	70630	21089	0.0
SumSquares	8757	103	1.0	10069	4	1.0	15352	558	1.0	2427	225	1.0	48238	2280	1.0
Trigonometric1	1334	381	0.96	10211	97	0.74	19919	4577	0.94	2379	312	0.8	21115	317	1.0
Trigonometric2	61209	2235	0.0	12641	477	1.0	5823	686	1.0	6383	1432	0.58	99812	22224	0.28
WWavy	6590	1751	0.0	10340	76	1.0	13221	907	1.0	3454	592	1.0	52355	8667	0.0
Weierstrass	67323	2585	0.0	13265	471	0.0	75201	0	0.0	8846	2923	0.0	54581	13629	0.0
Whitley	44508	3693	0.0	11484	448	0.32	14540	3778	0.14	3895	706	0.2	71142	12189	0.02
XinSheYangN3	1152	237	0.0	10289	53	0.54	282	42	0.16	2330	61	0.02	30334	12807	0.32
XinSheYangN4	7806	1467	0.0	10080	27	0.0	14059	1027	0.0	2414	310	0.04	41851	6388	0.02
Всього в середньому:	**27956**	**36013**	**0.45**	**10697**	**836**	**0.69**	**20839**	**18627**	**0.7**	**3517**	**1631**	**0.64**	**58063**	**34567**	**0.54**

DOI 10.15588/1607-3274-2021-2-18

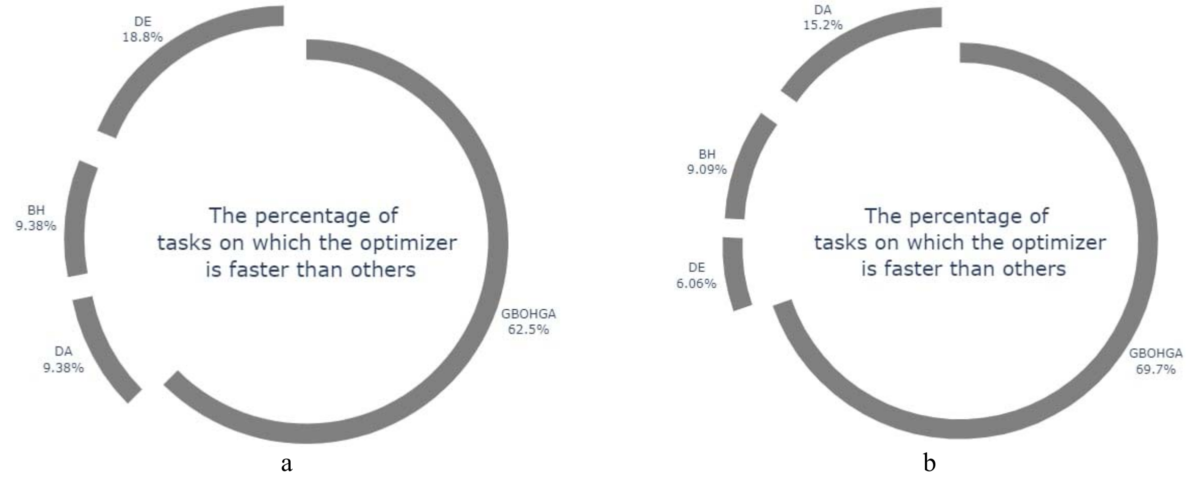


Figure 4 – The percentage of problems on which the optimizer is faster than others: a – with 5; b – with 25 variables

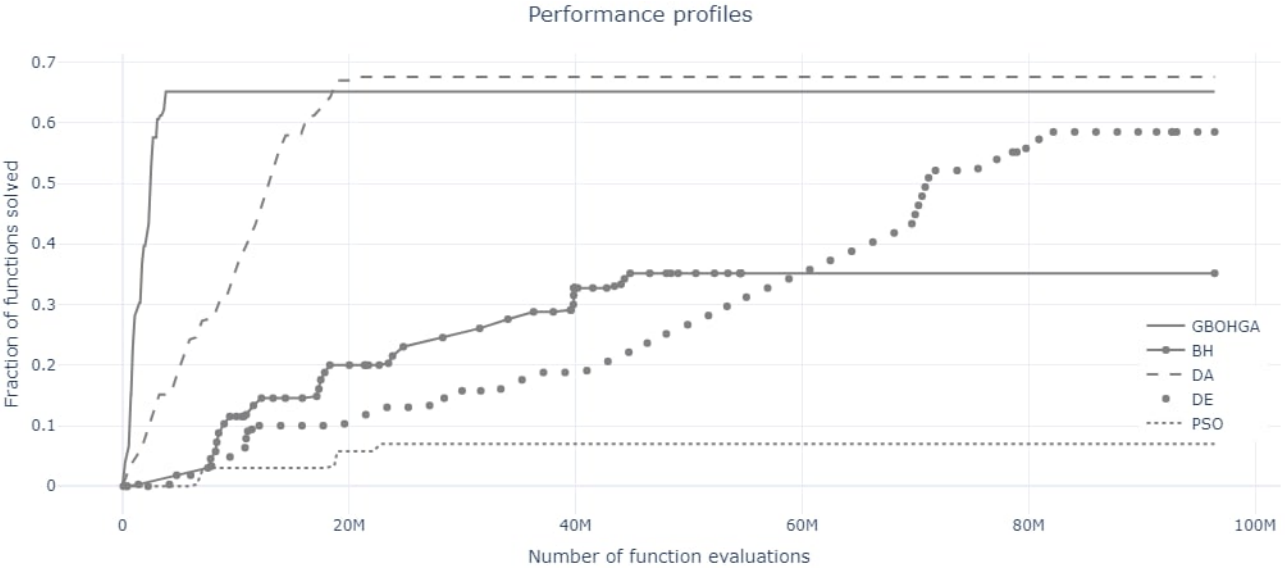


Figure 5 – Performance profiles in problems with 25 variables

6 DISCUSSION

Therefore, based on the results of numerical experiments of solving known global optimization problems, the proposed GBOHGA algorithm demonstrated the following advantages:

– due to the reduction in the number of evaluations of the objective function, the performance speed of the algorithm is higher compared to any other considered evolutionary algorithm (almost 70% of problems with 25 variables were solved faster using GBOHGA than using other solvers);

– by the criterion of reliability, GBOHGA is better than the basin-hopping algorithm and particle swarm optimization in problems with 5 variables;

– by the criterion of the robustness of solving problems with 25 variables, the algorithm is better than three of the four other solvers among the considered.

The work [29] shows the importance of the development and wide applicability of mobile neural networks. Such networks, as a rule, have parameters, the choice of which is necessary to achieve the minimum value of the loss function. Genetic algorithms (including ours) have the potential to improve the quality of such neural networks.

CONCLUSIONS

The paper proposes a new hybrid genetic algorithm GBOHGA, an optimization scheme that combines elements of BOHGA and HGDN methods and which integrates the strengths of local search with powerful global search capabilities. The advantage of the proposed method can be considered the simplicity of its implementation, as the key operators of the genetic algorithm and local search methods remain unchanged. Due to the prevention of a significant part of the unnecessary use of the local search operator, significant savings were achieved in the number of evaluations of the objective function and execution time of the algorithm, which made the method competitive with other widely used methods of global optimization.

DOI 10.15588/1607-3274-2021-2-18

Table 4 – Mean value and standard deviation of the number of function evaluations and success rate after 10 runs for each function with 25 variables from the test set for each method

Function name	**BH**			**DA**			**DE**			**GBOHGA**			**PSO**		
	μ	**σ**	**%**	**μ**	**σ**	**%**	**μ**	**σ**	**%**	**μ**	**σ**	**%**	**μ**	**σ**	**%**
Ackley1	69614	696	0.0	57876	690	1.0	375609	32	0.0	12222	3230	1.0	89650	4805	0.0
Alpine1	679007	9705	0.9	106564	13930	1.0	375921	0	1.0	42749	11233	1.0	96936	5266	0.0
Chung Reynolds	71201	74140	0.9	53552	193	1.0	327326	17048	1.0	3446	74	1.0	112826	7586	0.0
Exponential	19350	162	1.0	50222	39	1.0	19226	2552	1.0	4270	1350	1.0	78228	4271	0.0
Griewank	95882	260	1.0	50921	208	1.0	71432	72252	0.3	2934	464	1.0	84713	4532	0.0
HappyCat	117521	2317	0.0	54454	1293	0.0	377231	852	0.0	7539	2368	0.0	105716	4896	0.0
Periodic	27205	47	0.0	50248	46	0.0	375531	0	0.0	2925	291	0.0	83399	6435	0.0
PowellSum	126551	2321	1.0	52819	342	1.0	375401	0	1.0	2572	153	1.0	50547	5022	1.0
Rastrigin	206042	6159	0.0	55903	872	1.0	375658	25	0.0	7739	2091	1.0	98980	5377	0.0
Rosenbrock	299921	43662	0.1	54974	505	1.0	288349	141600	0.9	7858	1246	1.0	114993	9423	0.0
Salomon	34913	12131	0.8	54005	2108	0.0	340540	39914	0.0	5742	1442	0.2	45660	3652	0.0
Sargan	50216	60667	0.9	50315	59	1.0	375401	0	1.0	3862	973	1.0	112947	5184	0.0
SchafferF6	338304	36814	0.0	106756	10820	0.1	384706	4170	0.0	52606	11817	0.8	96166	7029	0.0
Schumer Steiglitz	17080	2347	0.0	54072	205	1.0	372626	5500	1.0	3277	128	1.0	108191	9569	0.0
Schwefel	188737	9032	0.0	53854	1343	0.0	328746	10168	1.0	5557	1477	0.0	82255	5516	0.0
Schwefel12	68272	10944	0.9	52554	339	1.0	377137	227	1.0	5503	721	1.0	103254	4828	0.0
Schwefel220	695733	6717	1.0	56755	2396	1.0	329758	14548	1.0	10814	1445	1.0	99846	9536	0.0
Schwefel221	498875	17899	1.0	75031	4684	1.0	375785	89	1.0	26047	7896	0.3	88771	7744	0.0
Schwefel222	341734	107144	0.0	73008	4702	1.0	366096	8306	1.0	23501	8166	1.0	116924	18060	0.0
Schwefel223	34150	39015	0.3	55713	296	1.0	375401	0	1.0	3638	160	1.0	96984	9145	0.0
Sphere	5043	236	1.0	50053	0	1.0	375427	0	1.0	7031	2890	1.0	25165	1330	0.9
Step	2727	0	0.0	51316	109	1.0	58713	5417	1.0	9532	2276	1.0	37924	1620	0.0
Step2	2727	0	0.0	51233	101	1.0	60326	7288	1.0	9625	2363	0.9	40472	3364	0.0
Step3	2727	0	0.0	51462	113	1.0	57626	6490	1.0	8432	2868	0.8	45182	5523	0.0
StrechedV SineWave	266810	83587	0.0	149045	41857	0.0	381953	1510	0.0	28011	41761	0.0	95680	5320	0.0
SumSquares	134815	12197	0.2	50775	95	1.0	329988	14109	1.0	4710	1188	1.0	96382	5384	0.0
Trigonometric1	52392	35191	0.6	52520	753	0.0	101946	96678	0.1	4426	1028	0.0	50013	24921	0.4
Trigonometric2	345622	12423	0.0	153122	17328	1.0	227821	45852	1.0	58542	11799	0.5	154987	15666	0.0
WWavy	81105	4064	0.0	54506	487	1.0	375991	123	0.0	10902	1623	1.0	80479	6991	0.0
Weierstrass	309193	12501	0.0	101761	5373	0.0	375921	0	0.0	41843	26637	0.0	65630	17056	0.0
Whitley	267065	168786	0.0	79505	3630	0.2	333035	46053	0.0	19983	1315	0.0	145926	7956	0.0
XinSheYangN3	6185	941	0.0	54670	165	0.0	2814	486	0.0	4129	79	0.0	23832	551	0.0
XinSheYangN4	6515	1496	0.0	55450	432	0.0	373601	5692	0.0	2956	126	0.0	30242	117	0.0
Total average:	**165552**	**190266**	**0.35**	**65909**	**28012**	**0.68**	**292213**	**133191**	**0.58**	**13483**	**17486**	**0.65**	**83603**	**33631**	**0.07**

According to the results of testing on a set of 33 functions with different landscapes, the proposed algorithm showed high results in terms of performance, and it is important to emphasize that the greatest benefits from its use can be obtained in problems of large dimensionality of search space.

The practical value of the work lies in the possibility of using the developed algorithm to solve global optimization problems that arise in engineering design, solving organizational and management problems, especially when the mathematical model of the problem has a high dimensionality.

The direction of further research is the analysis of the duration of local search, necessary to find a global solution to the optimization problem with acceptable accuracy, as well as the choice of truncation criterion in the proposed scheme.

ACKNOWLEDGEMENTS

The work was performed within the framework of the international cooperation program between the Dnipro University of Technology and the University of Koblenz-Landau (Germany). The authors would like to thank Doctor of Sciences, Professor of the Department of Modeling Simulation & Optimization of Complex Systems University of Koblenz-Landau Michael Hinze for support and a fruitful paper discussion.

REFERENCES

1. Locatelli M., Schoen F. Global optimization: theory, algorithms, and applications, *Society for Industrial and Applied Mathematics*, 2013, 432 p. DOI: 10.1137/1.9781611972672
2. Floudas C. A., Gounaris C. E. A review of recent advances in global optimization, *Journal of Global Optimization,* 2009, Vol. 45, No. 3, pp. 3–38. DOI 10.1007/s10898-008-9332-8
3. Fasano G., Pinter J. D. Space engineering: modeling and optimization with case studies. Switzerland, Springer, 2016, 492 p. DOI 10.1007/978-3-319-41508-6_18
4. Rao S. S. Engineering optimization: theory and practice. Hoboken, New Jersey, John Wiley & Sons, 2009, 813 p.
5. Siddique N., Adeli H. Nature inspired computing: an overview and some future directions, *Cognitive Computation*, 2015, Vol. 7, No. 6, pp. 706–714.
6. Wan W., Birch J. B. An improved hybrid genetic algorithm with a new local search procedure, *Journal of Applied Mathematics*, 2013, Vol. 2013, Article ID 103591, 10 p. DOI: 10.1155/2013/103591
7. Noack M. M., Funke S. W. Hybrid genetic deflated newton method for global optimisation, *Journal of Computational*

© Avramenko S. E., Zheldak T. A., Koriashkina L. S., 2021
DOI 10.15588/1607-3274-2021-2-18

and Applied Mathematics, 2017, Vol. 325, pp. 97–112. DOI: 10.1016/j.cam.2017.04.047

8. Voudouris C., Tsang E.P., Alsheddy A. Guided local search, *Handbook of Metaheuristics. International Series in Operations Research & Management Science* Eds.: M. Gendreau, JY. Potvin. Boston, Springer, 2010, Vol. 146, pp. 321–361. DOI: 10.1007/978-1-4419-1665-5_11
9. Pintér J. D. Global Eds.: P. M. Pardalos, H. E. Romeijn Optimization: Software, Test Problems, and Applications, *Handbook of Global Optimization. Nonconvex Optimization and Its Applications*. Boston, Springer, 2002, Vol. 62, pp. 515–569. DOI: 10.1007/978-1-4757-5362-2_15
10. Pardalos P. M., Zhigljavsky A. A., Žilinskas J. Advances in Stochastic and Deterministic Global Optimization. New York, Springer, 2016, 296 p. DOI: 10.1007/978-3-319-29975-4
11. Paulaviˇcius R., Zilinskas J. Simplicial Global Optimization, *Journal of Global Optimization,* 2014, Vol. 60, No. 4, pp. 61–86. DOI: 10.1007/978-1-4614-9093-7_3
12. Beliakov G. Cutting angle method – A tool for constrained global optimization, *Optimization Methods and Software,* 2004, Vol. 19, № 2, pp. 137–151.
13. Kvasov D. E., Sergeyev Y. D. Univariate geometric Lipschitz global optimization algorithms, *Numerical Algebra, Control and optimization*, 2012, Vol. 2, No. 1, pp. 69–90.
14. Malherbe C., Vayatis N. Global optimization of Lipschitz functions, *Machine Learning : 34th International Conference, Sydney, 6–11 August 2017 : proceedings*. Australia, PMLR, 2017, Vol. 70, pp. 2314–2323.
15. Rios L. M., Sahinidis N. V. Derivative-free optimization: a review of algorithms and comparison of software implementations, *Journal of Global Optimization*, 2013, Vol. 56, pp. 1247–1293. DOI 10.1007/s10898-012-9951-y
16. Sergeyev Y. Mukhametzhanov M., Kvasov D. et al. Derivative-free local tuning and local improvement techniques embedded in the univariate global optimization, *Journal of Optimization Theory and Applications*, 2017, Vol. 171, pp. 186–208. DOI: 10.1007/s10957-016-0947-5
17. Ye P., Pan G., Dong Z. Ensemble of surrogate based global optimization methods using hierarchical design space reduction, *Structural and Multidisciplinary Optimization*, 2018, Vol. 58, pp. 537–554. DOI: 10.1007/s00158-018-1906-6
18. Pengcheng Ye. A Review on Surrogate-Based Global Optimization Methods for Computationally Expensive Functions, *Software Engineering*, 2019, Vol. 7, No. 4, pp. 68–84. DOI: 10.11648/j.se.20190704.11
19. Li Z., Ruan Sh., Gu J. et al. Investigation on parallel algorithms in efficient global optimization based on multiple points infill criterion and domain decomposition, *Structural and Multidisciplinary Optimization*, 2016, Vol. 54, No. 4, pp. 747–773. DOI: 10.1007/s00158-016-1441-2
20. Wang Z., Gehring C., Kohli P. et al. Batched large-scale Bayesian optimization in high-dimensional spaces, *Artificial Intelligence and Statistics : Twenty-First International Conference, Lanzarote, 9–11 April 2018 : proceedings*. Spain, PMLR, 2018, Vol. 84, pp. 745–754.
21. Eriksson D., Pearce M., Gardner J. R. et al. Scalable Global Optimization via Local Bayesian Optimization, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8–14, 2019, Vancouver*. Canada, BC, 2019, pp. 5497–5508.
22. Mitchell M. An introduction to genetic algorithms. Cambridge, Mass., MIT Press, 1998, 221 p.
23. Farrell P. E., Birkisson Á., S. W. Funke Deflation techniques for finding distinct solutions of nonlinear partial differential equations, *Society for Industrial and Applied Mathematics. Journal on Scientific Computing*, 2015, Vol. 37, No. 4, pp. A2026–A2045.
24. Nieto-Fuentes R., Segura C., Valdez S. I. A Guided Local Search Approach for the Travelling Thief Problem, IEEE Congress on Evolutionary Computation, Rio de Janeiro, 8–13 July 2018. Brazil, 2018, pp. 1–8. DOI: 10.1109/CEC.2018.8477821
25. Touat M. Tayeb F. B., Benhamou B. et al. An Integrated Guided Local Search considering Human Resource Constraints for the Single-machine Scheduling problem with Preventive Maintenance, *IEEE International Conference on Systems, Man and Cybernetics, Bari, 6–9 October 2019*. Italy, 2019, pp. 3799–3804. DOI: 10.1109/SMC.2019.8914261
26. Peh S. C. W., Hong J. L. Eds.: Gervasi O. et al. GLSDock – Drug Design Using Guided Local Search, *Computational Science and Its Applications. Lecture Notes in Computer Science: 16th International Conference, Beijing, July 4–7 2016 : proceedings*. China, Springer, 2016, Vol. 9788, pp. 11–21. DOI: 10.1007/978-3-319-42111-7_2
27. Jamil M., Yang X. S. A literature survey of benchmark functions for global optimisation problems, *International Journal of Mathematical Modelling and Numerical Optimisation*, 2013, Vol. 4, No. 2, pp. 150–194. DOI: 10.1504/IJMMNO.2013.055204
28. Dall'Igna Júnior A., Silva R. S., Mundim K. C. et al. Performance and parameterization of the algorithm simplified generalized simulated annealing, *Genetics and Molecular Biology*, 2004, Vol. 27, № 4, pp. 616–622.
29. Khabarlak K., Koriashkina L. Fast Facial Landmark Detection and Applications: A Survey, arXiv: 2101.10808, 2021, 20 p.

Received 25.02.2021.
Accepted 30.04.2021.

УДК 004.023

КЕРОВАНИЙ ГІБРИДНИЙ ГЕНЕТИЧНИЙ АЛГОРИТМ РОЗВ'ЯЗАННЯ ЗАДАЧ ГЛОБАЛЬНОЇ ОПТИМІЗАЦІЇ

Авраменко С. Є. – магістр кафедри системного аналізу і управління Національного технічного університету «Дніпровська політехніка», інженер з машинного навчання, COMPARUS.UA, Дніпро, Україна.

Желдак Т. А. – канд. техн. наук, завідувач кафедри системного аналізу і управління Національного технічного університету «Дніпровська політехніка», Дніпро, Україна.

Коряшкіна Л. С. – канд. фіз.-мат. наук, доцент кафедри системного аналізу і управління Національного технічного університету «Дніпровська політехніка», Дніпро, Україна.

DOI 10.15588/1607-3274-2021-2-18

АНОТАЦІЯ

Актуальність. Провідною проблемою в світі штучного інтелекту є оптимізація складних систем, що нерідко подається у вигляді нелінійної функції, яку необхідно мінімізувати. Такі функції можуть виявитися мультимодальними, недиференційованими, і навіть, заданими у формі чорної скриньки. Побудова ефективних методів розв'язання задач глобальної оптимізації і сьогодні викликав значний інтерес серед науковців.

Мета. Розробка нового гібридного генетичного алгоритму розв'язання задачі безумовної глобальної оптимізації, швидшого за існуючі аналоги.

Методи. Одним з важливих викликів, з якими стикаються гібридні методи під час розв'язування задач нелінійної глобальної оптимізації, є раціональне використання локального пошуку, оскільки його реалізація супроводжується достатньо коштовними обчислювальними витратами. В даній роботі запропонований новий гібридний генетичний алгоритм GBOHGA, який відтворює керований локальний пошук, і поєднує дві успішні модифікації генетичних алгоритмів, перша з яких BOHGA встановлює якісний баланс між локальним та глобальним пошуком, друга – HGDN – запобігає повторному дослідженню раніше вже розвіданої області пошукового простору. Окрім цього, було запропоновано модифіковану bump-функцію та адаптивну схему визначення її параметру – радіусу області «дефляції» цільової функції в околі вже знайденого локального мінімуму – задля прискорення роботи алгоритму.

Результати. Роботу GBOHGA з іншими відомими стохастичними пошуковими евристиками на наборі із 33 тестових функцій в 5 та 25 вимірному просторах. Результати обчислювальних експериментів свідчать про конкурентну спроможність GBOHGA, особливо в задачах з мультимодальними функціями і великою кількістю змінних.

Висновки. Новий гібридний алгоритм GBOHGA, розроблений на основі інтеграції ідей керованого локального пошуку і алгоритмів BOHGA та HGDN, дозволяє значно економити обчислювальні ресурси та прискорювати розв'язання задачі глобальної оптимізації. Його варто застосовувати для розв'язання задач глобальної оптимізації, що виникають в інженерному проектуванні, вирішення організаційно-управлінських проблем, особливо коли математична модель задачі є складною і має високу розмірність.

КЛЮЧОВІ СЛОВА: нелінійна оптимізація, глобальний мінімум, стохастичні пошукові евристики, гібридний підхід, генетичний алгоритм, оператор дефляції, керований локальний пошук.

УДК 004.023

УПРАВЛЯЕМЫЙ ГИБРИДНЫЙ ГЕНЕТИЧЕСКИЙ АЛГОРИТМ РЕШЕНИЯ ЗАДАЧ ГЛОБАЛЬНОЙ ОПТИМИЗАЦИИ

Авраменко С. Є. – магистр кафедры системного анализа и управления Национального технического университета «Днепровская политехника», інженер по машинному обучению, COMPARUS.UA, Днепр, Украина.

Желдак Т.А. – канд. техн. наук, заведующий кафедрой системного анализа и управления Национального технического университета «Днепровская политехника», Днепр, Украина.

Коряшкина Л. С. – канд. физ.-мат. наук, доцент кафедры системного анализа и управления Национального технического университета «Днепровская политехника», Днепр, Украина.

АННОТАЦИЯ

Актуальность. Одной из ведущих проблем в мире искусственного интеллекта является оптимизация сложных систем, нередко представляемая в виде нелинейной функции, которую необходимо минимизировать. При этом функции могут оказаться многоэкстремальными, негладкими, и даже, заданными в форме черного ящика. Построение эффективных методов решения таких задач и сегодня вызывает значительный интерес в научном сообществе.

Цель. Разработка нового гибридного генетического алгоритма решения задачи глобальной оптимизации, который по скорости превосходил бы существующие аналоги.

Метод. Одним из важных вызовов, с которыми сталкиваются гибридные методы при решении задач нелинейной глобальной оптимизации, является рациональное использование локального поиска, поскольку его реализация сопровождается высокими вычислительными затратами. В данной работе предложен новый гибридный генетический алгоритм GBOHGA, который воспроизводит управляемый локальный поиск, и объединяет две успешные модификации генетических алгоритмов, первая – BOHGA – устанавливает качественный баланс между локальным и глобальным поиском, вторая – HGDN – предотвращает повторное исследование ранее уже разведанных областей поискового пространства. Кроме этого, для ускорения работы алгоритма предложена модифицированная bump-функция и адаптивная схема определения ее параметра, а именно, радиуса области «дефляции» целевой функции в окрестности уже найденного локального минимума.

Результаты. Проведено сравнение работы GBOHGA с другими известными стохастическими поисковыми эвристиками на наборе из 33 тестовых функций в 5 и 25 мерном пространствах. Результаты вычислительных экспериментов свидетельствуют о конкурентной способности GBOHGA, особенно при решении задач с мультимодальными функциями и большим количеством переменных.

Выводы. Новый гибридный алгоритм GBOHGA, разработанный на основе интеграции идей управляемого локального поиска и алгоритмов BOHGA и HGDN, позволяет значительно экономить вычислительные ресурсы и ускорять решение задачи глобальной оптимизации. Его целесообразно применять для решения задач глобальной оптимизации, возникающих в инженерном проектировании, решения организационно-управленческих проблем, особенно когда математическая модель задачи является сложной и имеет высокую размерность.

КЛЮЧЕВЫЕ СЛОВА: нелинейная оптимизация, глобальный минимум, стохастические поисковые эвристики, гибридный подход, генетический алгоритм, оператор дефляции, управляемый локальный поиск.

DOI 10.15588/1607-3274-2021-2-18

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Locatelli M. Global optimization: theory, algorithms, and applications / M. Locatelli, F. Schoen. – Society for Industrial and Applied Mathematics, 2013. – 432 p. DOI: 10.1137/1.9781611972672
2. Floudas C.A. A review of recent advances in global optimization / C.A. Floudas, C.E. Gounaris // Journal of Global Optimization. – 2009. – Vol. 45, № 3. – P. 3–38. DOI 10.1007/s10898-008-9332-8
3. Fasano G. Space engineering: modeling and optimization with case studies / G. Fasano, J.D. Pinter. – Switzerland: Springer, 2016. – 492 p. DOI 10.1007/978-3-319-41508-6_18
4. Rao S. S. Engineering optimization: theory and practice / S. S. Rao. – Hoboken, New Jersey : John Wiley & Sons, 2009. – 813 p.
5. Siddique N. Nature inspired computing: an overview and some future directions / N. Siddique, H. Adeli // Cognitive Computation. – 2015. – Vol. 7, № 6. – P. 706–714.
6. Wan W. An improved hybrid genetic algorithm with a new local search procedure / W. Wan, J. B. Birch // Journal of Applied Mathematics. – 2013. – Vol. 2013. – Article ID 103591, 10 p. DOI:10.1155/2013/103591
7. Noack M. M. Hybrid genetic deflated newton method for global optimisation / M. M. Noack, S. W. Funke // Journal of Computational and Applied Mathematics. – 2017. – Vol. 325. – P. 97–112. DOI: 10.1016/j.cam.2017.04.047
8. Voudouris C. Guided local search / C. Voudouris, E. P. Tsang, A. Alsheddy // Handbook of Metaheuristics. International Series in Operations Research & Management Science / Eds.: M. Gendreau, JY. Potvin. – Boston : Springer, 2010. – Vol. 146. – P. 321–361. DOI: 10.1007/978-1-4419-1665-5_11
9. Pintér J. D. Global Optimization: Software, Test Problems, and Applications / János D. Pintér // Handbook of Global Optimization. Nonconvex Optimization and Its Applications / Eds.: P. M. Pardalos, H. E. Romeijn. – Boston : Springer, 2002. – Vol. 62. – P. 515–569. DOI: 10.1007/978-1-4757-5362-2_15
10. Pardalos P. M. Advances in Stochastic and Deterministic Global Optimization / P. M. Pardalos, A. A. Zhigljavsky, J. Žilinskas. – New York : Springer, 2016. – 296 p. –DOI: 10.1007/978-3-319-29975-4
11. Paulaviˇcius R. Simplicial Global Optimization / R. Paulaviˇcius, J. Zilinskas // Journal of Global Optimization. – 2014. – Vol. 60, № 4. – P. 61–86. DOI: 10.1007/978-1-4614-9093-7_3
12. Beliakov G. Cutting angle method – A tool for constrained global optimization / G. Beliakov // Optimization Methods and Software. – 2004. – Vol. 19, № 2. – P. 137–151.
13. Kvasov D. E. Univariate geometric Lipschitz global optimization algorithms / D. E. Kvasov, Y. D. Sergeyev // Numerical Algebra, Control and optimization. – 2012. – Vol. 2, № 1. – P. 69–90.
14. Malherbe C. Global optimization of Lipschitz functions. / C. Malherbe, N. Vayatis // Machine Learning : 34th International Conference, Sydney, 6–11 August 2017 : proceedings. – Australia : PMLR, 2017. – Vol. 70. – P. 2314–2323.
15. Rios L. M. Derivative-free optimization: a review of algorithms and comparison of software implementations / L. M. Rios, N. V. Sahinidis // Journal of Global Optimization. – 2013. – Vol. 56. – P. 1247–1293. – DOI 10.1007/s10898-012-9951-y
16. Sergeyev Y. Derivative-free local tuning and local improvement techniques embedded in the univariate global optimization / [Y. Sergeyev, M. Mukhametzhanov, D. Kvasov et al.] // Journal of Optimization Theory and Applications. – 2017. – Vol. 171. – P. 186–208. DOI: 10.1007/s10957-016-0947-5
17. Ye P. Ensemble of surrogate based global optimization methods using hierarchical design space reduction / P. Ye, G. Pan, Z. Dong // Structural and Multidisciplinary Optimization. – 2018. – Vol. 58. – P. 537–554. DOI: 10.1007/s00158-018-1906-6
18. Pengcheng Ye. A Review on Surrogate-Based Global Optimization Methods for Computationally Expensive Functions / Ye. Pengcheng // Software Engineering. – 2019. – Vol. 7, № 4. – P. 68–84. DOI: 10.11648/j.se.20190704.11
19. Investigation on parallel algorithms in efficient global optimization based on multiple points infill criterion and domain decomposition / [Z. Li, Sh. Ruan, J. Gu et al.] // Structural and Multidisciplinary Optimization. – 2016. – Vol. 54, № 4. – P. 747–773. DOI: 10.1007/s00158-016-1441-2
20. Wang Z. Batched large-scale Bayesian optimization in high-dimensional spaces / [Z. Wang, C. Gehring, P. Kohli et al.] // Artificial Intelligence and Statistics : Twenty-First International Conference, Lanzarote, 9–11 April 2018 : proceedings. – Spain : PMLR, 2018. – Vol. 84. – P. 745–754.
21. Eriksson D. Scalable Global Optimization via Local Bayesian Optimization / [D. Eriksson, M. Pearce, J. R. Gardner et al.] // Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8–14, 2019, Vancouver. – Canada: BC, 2019. – P. 5497–5508.
22. Mitchell M. An introduction to genetic algorithms / M. Mitchell. – Cambridge, Mass. : MIT Press, 1998. – 221 p.
23. Farrell P. E. Deflation techniques for finding distinct solutions of nonlinear partial differential equations / P. E. Farrell, Á. Birkisson, S. W. Funke // Society for Industrial and Applied Mathematics. Journal on Scientific Computing. – 2015. – Vol. 37, № 4. – P. A2026–A2045.
24. Nieto-Fuentes R. A Guided Local Search Approach for the Travelling Thief Problem / R. Nieto-Fuentes, C. Segura, S. I. Valdez // IEEE Congress on Evolutionary Computation, Rio de Janeiro, 8–13 July 2018. – Brazil, 2018. – P. 1–8. DOI: 10.1109/CEC.2018.8477821
25. An Integrated Guided Local Search considering Human Resource Constraints for the Single-machine Scheduling problem with Preventive Maintenance / [M. Touat, F. B. Tayeb, B. Benhamou et al.] // IEEE International Conference on Systems, Man and Cybernetics, Bari, 6–9 October 2019. – Italy : 2019. – P. 3799–3804. – DOI: 10.1109/SMC.2019.8914261
26. Peh S. C. W. GLSDock – Drug Design Using Guided Local Search / S. C. W. Peh, J. L. Hong // Computational Science and Its Applications. Lecture Notes in Computer Science. / Eds.: Gervasi O. et al. // 16th International Conference, Beijing, China, July 4–7 2016 : proceedings.– Springer, 2016. – Vol 9788. – P. 11–21. – DOI: 10.1007/978-3-319-42111-7_2
27. Jamil M. A literature survey of benchmark functions for global optimisation problems / M. Jamil, X. S. Yang // International Journal of Mathematical Modelling and Numerical Optimisation. – 2013. – Vol. 4, № 2. – P. 150–194. DOI: 10.1504/IJMMNO.2013.055204
28. Dall'Igna Júnior A. Performance and parameterization of the algorithm simplified generalized simulated annealing / [A. Dall'Igna Júnior, R. S. Silva, K. C. Mundim et al.] // Genetics and Molecular Biology. – 2004. – Vol. 27, № 4. – P. 616–622.
29. Khabarlak K. Fast Facial Landmark Detection and Applications: A Survey / K. Khabarlak, L. Koriashkina // arXiv: 2101.10808. – 2021. – 20 p.

DOI 10.15588/1607-3274-2021-2-18