The algorithm of overall optimization based on the principles of intraspecific competition of orb-web spiders

V A Baranov¹, N M Lugovaya², A S Mikhalev³, V I Kudymov⁴ and T V Strekaleva⁵

¹LCC "Techvision", 12, Kapitanskaya str., 660064, Krasnoyarsk, Russia
²LCC "Siberian Integration systems", 5, Belinsky str., 660032, Krasnoyarsk, Russia
³Siberian Federal University, 26, Kirensky str., 660074, Krasnoyarsk, Russia
⁴JSC Academician M.F. Reshetnev «ISS» Zheleznogorsk, Krasnoyarsk region, Russia
⁵Reshetnev Siberian State University of Science and Technology, 31, Krasnoyarsky Rabochy Av., 660037, Krasnoyarsk, Russia

E-mail: asmikhalev@yandex.ru

Abstract. This work presents a new global optimization algorithm based on the behavior of orb-web spiders. The spider method is a heuristic competitive iterative method of random search whose main idea is to model the behavior of Garden orb-web spiders. The paper describes a solution search strategy based on the features of building a web and competitive behavior; the step-by-step algorithm for solving the problem is presented. The concept of the method is simple; the algorithm does not contain complex calculations. The positions of spiders and flies (test points) are generated randomly, herewith, due to the correct selection of the obtained values, the search for an optimum lead to a rather accurate result. The article describes all the parameters used in the method and presents recommendations for changing settings. The selection of the optimal parameters for various classes of test functions is performed. Parameter settings were performed on three classes of test functions: unimodal, ravine, and multiextremal. As part of the work, a computational experiment was conducted to study the effectiveness of the developed method as well. We compared the proposed method with other proven metaheuristic optimization algorithms. The method showed good results both when working with simple (unimodal) functions, and when finding the optimum of functions with a more complex landscape (multiextremal functions). Based on the above advantages, the spider method can be adapted to solve applied problems with relative ease.

1. Introduction

The task of finding the optimal value of the objective function is one of the urgent tasks in the field of science and technology. Most of the practical problems that arise in various fields of human activity can be reduced to optimization problems. These are various tasks of calculating the parameters of structures, the trajectory of the object, the distribution of resources. Application of the optimum search strategy for constructing and solving the synthesis problems of complex systems leads to an increase in the quality of management, planning, and design.

The rapidly growing complexity of modern optimization problems requires the use of effective tools to solve them. For solving global optimization problems, the class of heuristic methods that use...
laws and principles borrowed from nature are particularly well established. These include evolutionary optimization methods and Swarm Intelligence methods.

These methods belong to the class of population methods, which are based on modeling the collective behavior of self-organizing systems, the interacting elements of which are called agents.

Evolutionary algorithms are based on a collective learning process within the population of agents. The population is randomly initialized, and then at each step the algorithm simulates the process of natural selection, when the stronger agents from the population outlive the weaker ones and produce the next generation.

The Swarm Intelligence methods, as a new type of evolutionary computer technology, use certain rules for their work that specify the procedure for exchanging information between agents and with the external environment to achieve a predetermined goal. Unlike evolutionary algorithms, it is no longer necessary to create a new population at every step of the algorithm. The concept of Swarm Intelligence is based on the collective decentralized motion of population agents.

Researchers are rather interested in the population methods as they have a number of advantages: operating speed, high reliability and noise immunity, simple implementation and the absence of restrictions on the dimension of the problem. Within the framework of random search schemes, it is easy to build new algorithms that implement various heuristic adaptation procedures.

This paper proposes a new optimization algorithm based on the behavior of orb-web spiders. Currently, scientists are implementing a social spider algorithm [1], which is based on simulating the behavior of the subspecies Mallos gregalis and Oecobius civitas. This algorithm is based on a search strategy for social spiders that use web vibrations to determine the location of their prey. However, it is believed that spiders are single and rarely interact with each other, and it is worth considering that spiders are predators and compete with each other [2]. The proposed algorithm is based on the features of building spider webs and the competitive behavior of spiders.

2. Statement of problem
In the general statement, the global optimization problem is formulated as follows [3]:

\[ f^* = f(x^*) = \min_{x \in X} f(x) \] (1)

where \( f(x) \) – an objective, \( x^* \) – globally optimal point or globally optimal solution, \( x = \{x_1, x_2, \ldots, x_m\} \) – \( m \)-dimensional vector of the decision variables, \( X \subset \mathbb{R}^m \) – area of Euclidean space.

The area in which the solution to the global optimization problem is sought is defined as follows:

\[ D = \{x \mid x_{i_{\text{min}}} \leq x_i \leq x_{i_{\text{max}}}, i = 1, m \} \subset \mathbb{R}^m \] (2)

The optimized function satisfies the Lipschitz condition with constant \( L \), i.e. there is inequality

\[ |f(x) - f(z)| \leq L \|x - z\| \leq \max_{i \in \Omega_{\text{opt}}} |x_i| \] (3)

This condition limits the growth of the function. The function is restricted and continuous almost everywhere on \( \mathbb{X} \).

3. Description of behavior
Orb-web spiders make large wheel-shaped networks from a web. They include four main components: an outer frame made of very thick and strong threads on which the network is held; radial threads on which adhesive threads are attached and along which a spider moves; an adhesive spiral thread, on which the catchability of the hunting net depends; and the central net, where there is normally a spider (not always).

First of all, a web frame is being weaved. When radial threads and a frame are ready, a spider returns to the center and begins to lay a temporary auxiliary spiral. An auxiliary spiral fastens the structure and serves as a spider path for the construction of a hunting spiral. The entire main frame of
the network, including the radii, is made of a non-adhesive thread, and a double thread coated with an adhesive is used for the hunting spiral.

An orb-web spider can not only hang upside down in the center of the network, but it can comfortably recline in a special crib from the web or be in a shelter that has been moved outside the network. The shelter can be a cone of folded leaves, lined with the web from the inside, or a cobwebby cone-tube built in a rock fissure. In the diet of orb-web spiders flying insects predominate, mainly dipterans. Some large species (e.g. Argiope), in addition to flies, eat bees and orthopterous insects [4].

The web for orb-web spiders is not of great value, as, for example, for other spiders, therefore they can rebuild it day by day. Many authors believe that orb-web spiders behave like predators when actively searching for their prey and, perhaps, they even weave test webs using less silk. Spiders can search areas to make new webs as well because of such negative reasons as web destruction, parasitism and interference from conspecifics. Despite these rules adjusting the movement of a spider and its network, the more significant reasons are the risk of predation and the presence of more abundant prey [5].

Since the weaving of webs is a labour-consuming process, some types of orb-web spiders resort to the behavior of “invasion”. It is assumed that they will capture the web of other spiders, and they do not create their own, since creating a web not only requires a much energy and time, but significantly increases the risk of predation as well. A network that caught more prey is more likely to be captured [6].

4. Decision-making strategy

The spider method is a heuristic competitive iterative method of random search, the main idea of which is to model the behavior of Garden orb-web spiders from the family of orb-web spiders.

Each iteration represents a day. During the day, the spider hunts, exploring the search area D. At the end of the day, the spider searches for a new, good place to weave a web and destroys the old one, since the trapping net becomes unusable because some insects get into it (small insects, as well as large ones the spider disposes of), breaking off spider webs around an unsuitable prey. The construction of a new web occurs at night, and in the morning it is ready to catch insects. Such diurnal activity is an example of spider’s rational behavior, since there are less enemies at night: insectivorous birds sleep, but for others a spider is not visible in the dark. Despite the lack of light, the construction of a new web at night is very successful, since a spider is guided in his work not by sight, but by the sense of touch [7].

To describe the behavior of Garden orb-web spiders in nature, the following concepts are introduced:

Web. While weaving a web, four parameters are used: the maximum step size $S_{\text{max}}$, the range parameter of generation of flies $k_g$, the number of sticky areas $N_{st}$, and the radius of the non-sticky area $R_n$. The maximum step size is responsible for the radius of a web (in both coordinates, labeled $m$ in the figure), the range parameter is responsible for how much of the maximum distance the first non-sticky part of the web is (labeled $n$). Further, the entire remaining part of the web is divided into alternating non-sticky and sticky areas (figure 1). The radius of non-sticky areas is defined by the parameter $R_n$ (marked as $k$, it is meant as the thickness of non-sticky areas; we set it very small, as if they are non-sticky threads), all the rest is filled with sticky areas. After weaving a web, $N_F$ of flies are generated in each of the sticky areas.
Hunt. The spider sits in the center of the web and waits for the prey to fall into the nets. When an insect enters the sticky area of the web, the spider moves to the prey in specified steps. The size of the steps is determined by the parameter $S_{\text{max}}$. The segment between the center of the web and a certain fly is divided into equal parts. The value of the function at each point is fixed when each fly is captured. After that, obtained values are compared and the lowest one is searched. At the end of the hunt, the spider moves to the minimum point.

Search for a place to build a web. At the end of the day, the spider is searching a new place to build a web. When searching, it selects a place with high humidity and a certain angle of sunlight falling. With the good choice of the place of weaving, the web becomes invisible to insects and predators. Insects often fly to the light (especially in the twilight and in the early morning), and spiders know this destructive “passion”. If a spider enters the territory of another spider during the movement, competition begins [6]. A stronger spider with the smallest value (greatest) of the function at this iteration wins and eats the spider with a larger (smaller) value of the function.

The general scheme of the spider method is shown in figure 2.

**Figure 2.** General scheme of the method.

### 5. The algorithm for solving the problem

Step 1. Setting initial parameters, let us put $k = 0$.

Step 2. Number $N_s$ of spiders generated using uniform law from $-D/2$ to $D/2$.

Step 3. Generation of webs and flies for each spider, passage to each fly, movement of the spider.

Step 3.1. Calculate the size of the sticky area based on the formula

$$R_s = 1.0 - k_s - R_s \cdot (N_{st} - 1)$$

Step 3.2. Build a web that includes: non-sticky area of radius $S_{\text{max}} \cdot k_s$ and alternating sticky and non-sticky areas of the radii $R_s \cdot S_{\text{max}}$ and $R_s \cdot S_{\text{max}}$ respectively (the number of sticky areas is determined by the $N_{st}$ parameter).

Step 3.3. Generate $N_f$ of flies in each of the sticky areas using the uniform distribution law.
Step 3.4. Make a passage to each fly, for which divide the segment from the center of the web to the fly into $S$ equal. Fix the result of the function at each point.

Step 3.5. Search for the best point from the previous step and move the spider to this point.

Step 4. Search for the best function value among all spiders.

Step 5. For all spiders except the best one, take a step towards the best spider of $S_I$ size.

Step 6. Check the distance for each spider. If any spiders are at the distance less than $D_{min}$, remove that spider whose current function value is higher.

Step 7. Multiply $S_{max}$ and $S_I$ by $k_d$.

Step 8. $k = k + 1$. If $k < K$, return to the step 3.

Step 9. Search for the best function value among all remaining spiders. It will be a search value.

6. The description of method parameters and recommendations for choosing values

The number of spiders $N_s$. This parameter determines the number of spiders in the first iteration. The more the definition domain is investigated, the more spiders must be set.

The number of iterations $K$. The larger this parameter, the more accurate the solution. It should be taken into consideration that all other parameters must be selected appropriately. Like with any optimization method, as the value of this parameter increases, the computational complexity of the algorithm increases.

Dimensions of the area $D$. The area in which the solution to the global optimization problem is sought.

The number of non-sticky areas $N_{st}$. The number of non-sticky areas in the web. The generation of flies occurs in them. The larger the number of areas, the more flies will be generated.

The radius of non-sticky $R_{nc}$. The coefficient determining the radius of each non-sticky area (except the first one).

The number of steps $S$. The number of sections into which the segment is divided from the center of the web to each fly.

The maximum step size $S_{max}$. The maximum step size of $S$, web radius equivalent.

The step after iteration $S_I$. The step to the determined minimum function value.

The number of flies $N_F$. The number of flies generated in each sticky area.

The range factor for the generation of flies $k_g$. The coefficient determining the radius of the first non-sticky area.

Minimum distance between spiders $D_{min}$. The minimum distance at which spiders do not compete with each other.

The reduction factor $k_d$. Coefficient to reduce variables $S_{max}, S_I$.

7. Finding the optimal method parameters

The efficiency of the algorithms when searching for a global minimum is due to the setting of their respective parameters. In the framework of this work, the parameters of the proposed method were selected on three classes of test functions: unimodal (sphere function), ravine (Rosenbrock function) and multimodal (Rastrigin, Ackley, Goldstein-Price and Beale functions). The setting was carried out according to three most significant parameters: the number of iterations, the number of spiders and the number of flies. As a criterion of efficiency, the dispersion value was chosen.

The following study was carried out for each class of functions. A series of 100 solutions to the same problem with the same parameter values was performed. For the resulting fetch $\{f^1, f^2, ..., f^{100}\}$ the dispersivity was calculated as $\sigma_f = \frac{1}{99} \sum_{i=1}^{100} (f^i - \bar{f})^2$. For this criterion, those parameter values are considered optimal for which the dispersivity takes the smallest values.

The obtained results are shown in tables 1-6.
Table 1. Selection of parameters for the sphere function $x^2 + y^2 + z^2$.

| $K$ | $N_s$ | $R_n$ | $S$ | $S_{max}$ | $S_l$ | $N_F$ | $k_g$ | $k_d$ | $\bar{f}$ | $F_{min}$ | $\sigma_f^2$ |
|-----|-------|-------|-----|-----------|-------|-------|-------|-------|-----------|-----------|-----------|
| 100 | 10    | 0.05  | 5   | 1         | 1     | 10    | 0.3   | 0.95  | 3.42E-7   | 8.395E-8  | 1.882E-14 |
| 120 | 20    | 0.05  | 5   | 1         | 1     | 15    | 0.3   | 0.95  | 3.633E-8  | 2.297E-9  | 2.145E-16 |
| 140 | 30    | 0.05  | 5   | 1         | 1     | 20    | 0.3   | 0.95  | 4.497E-9  | 5.418E-11 | 2.656E-18 |
| 160 | 40    | 0.05  | 5   | 1         | 1     | 25    | 0.3   | 0.95  | 5.894E-10 | 1.862E-10 | 3.577E-20 |
| 180 | 50    | 0.05  | 5   | 1         | 1     | 30    | 0.3   | 0.95  | 7.763E-11 | 2.419E-11 | 5.865E-22 |
| 200 | 60    | 0.05  | 5   | 1         | 1     | 35    | 0.3   | 0.95  | 8.525E-12 | 2.850E-12 | 7.826E-24 |

Table 2. Selection of parameters for the Rosenbrock function $(1-x)^2 + 100\cdot(y-x^2)^2$.

| $K$ | $N_s$ | $R_n$ | $S$ | $S_{max}$ | $S_l$ | $N_F$ | $k_g$ | $k_d$ | $\bar{f}$ | $F_{min}$ | $\sigma_f^2$ |
|-----|-------|-------|-----|-----------|-------|-------|-------|-------|-----------|-----------|-----------|
| 100 | 10    | 0.05  | 5   | 1         | 1     | 10    | 0.3   | 0.95  | 2.206E-3  | 2.188E-10 | 4.666E-4  |
| 120 | 20    | 0.05  | 5   | 1         | 1     | 15    | 0.3   | 0.95  | 3.444E-8  | 1.146E-11 | 4.150E-15 |
| 140 | 30    | 0.05  | 5   | 1         | 1     | 20    | 0.3   | 0.95  | 3.494E-9  | 1.999E-11 | 1.505E-17 |
| 160 | 40    | 0.05  | 5   | 1         | 1     | 25    | 0.3   | 0.95  | 3.67E-10  | 9.57E-12  | 1.793E-19 |
| 180 | 50    | 0.05  | 5   | 1         | 1     | 30    | 0.3   | 0.95  | 3.733E-11 | 7.594E-13 | 8.837E-22 |
| 200 | 60    | 0.05  | 5   | 1         | 1     | 35    | 0.3   | 0.95  | 4.051E-12 | 1.51E-13  | 1.167E-23 |

Table 3. Selection of parameters for the Ackley function $20\cdot e^{0.2\cdot\sqrt{0.5(x^2+y^2)}} - e^{0.5(\cos 2\pi x+\cos 2\pi y)}$.

| $K$ | $N_s$ | $R_n$ | $S$ | $S_{max}$ | $S_l$ | $N_F$ | $k_g$ | $k_d$ | $\bar{f}$ | $F_{min}$ | $\sigma_f^2$ |
|-----|-------|-------|-----|-----------|-------|-------|-------|-------|-----------|-----------|-----------|
| 100 | 10    | 0.05  | 5   | 1         | 1     | 10    | 0.3   | 0.95  | 1.03E-3   | 3.671E-4  | 5.909E-8  |
| 120 | 20    | 0.05  | 5   | 1         | 1     | 15    | 0.3   | 0.95  | 3.681E-4  | 1.607E-4  | 6.87E-9   |
| 140 | 30    | 0.05  | 5   | 1         | 1     | 20    | 0.3   | 0.95  | 1.302E-4  | 5.395E-5  | 6.931E-10 |
| 160 | 40    | 0.05  | 5   | 1         | 1     | 25    | 0.3   | 0.95  | 4.446E-5  | 1.937E-5  | 7.22E-11  |
| 180 | 50    | 0.05  | 5   | 1         | 1     | 30    | 0.3   | 0.95  | 1.578E-5  | 8.195E-6  | 7.36E-12  |
| 200 | 60    | 0.05  | 5   | 1         | 1     | 35    | 0.3   | 0.95  | 5.634E-6  | 2.368E-6  | 1.74E-12  |

Table 4. Selection of parameters for the Rastrigin function $20 + (x^2 - 10\cdot\cos(2\pi x)) + (y^2 - 10\cdot\cos(2\pi y))$.

| $K$ | $N_s$ | $R_n$ | $S$ | $S_{max}$ | $S_l$ | $N_F$ | $k_g$ | $k_d$ | $\bar{f}$ | $F_{min}$ | $\sigma_f^2$ |
|-----|-------|-------|-----|-----------|-------|-------|-------|-------|-----------|-----------|-----------|
| 100 | 10    | 0.05  | 5   | 1         | 1     | 10    | 0.3   | 0.95  | 0.288     | 8.113E-6  | 0.203     |
| 120 | 20    | 0.05  | 5   | 1         | 1     | 15    | 0.3   | 0.95  | 0.089     | 4.682E-7  | 0.081     |
| 140 | 30    | 0.05  | 5   | 1         | 1     | 20    | 0.3   | 0.95  | 0.0298    | 8.2E-8    | 0.028     |
| 160 | 40    | 0.05  | 5   | 1         | 1     | 25    | 0.3   | 0.95  | 0.019     | 1.408E-8  | 0.019     |
| 180 | 50    | 0.05  | 5   | 1         | 1     | 30    | 0.3   | 0.95  | 0.009     | 1.789E-9  | 0.009     |
| 200 | 60    | 0.05  | 5   | 1         | 1     | 35    | 0.3   | 0.95  | 7.93E-10  | 2.879E-10 | 7.359E-20 |

Table 5. Selection of parameters for the Goldstein-Price function $(1+(1+x+y)(19-14x+3x^2-14y+6xy+3y^2))(30+(2x-3y)(18-32x+12x^2+48y-36xy+27y^2))$.

| Method parameters | $\bar{f}$ | $F_{min}$ | $\sigma_f^2$ |
|-------------------|-----------|-----------|-------------|
|                   |           |           |             |
Table 6. Selection of parameters for the Beale’s function

\[(1.5 - x + xy) + (2.25 - x + xy)^2 + (2.625 - x + xy^3)^2.\]

| \(K\) | \(N_s\) | \(R_n\) | \(S\) | \(S_{max}\) | \(S_f\) | \(N_F\) | \(k_g\) | \(k_d\) | \(\bar{f}\) | \(F_{min}\) | \(\sigma_f^2\) |
|------|-------|-------|-----|-------|-----|-----|-----|-----|-----|-------|-------|
| 100  | 10    | 0.05  | 5   | 1     | 1   | 10  | 0.3 | 0.95 | 3.0000386 | 3.00000599 | 5.377E-10 |
| 120  | 20    | 0.05  | 5   | 1     | 1   | 15  | 0.3 | 0.95 | 3+4.833E-6 | 3+8.285E-7  | 5.77E-12   |
| 140  | 30    | 0.05  | 5   | 1     | 1   | 20  | 0.3 | 0.95 | 3+5.50E-7  | 3+1.947E-8  | 8.3E-14    |
| 160  | 40    | 0.05  | 5   | 1     | 1   | 25  | 0.3 | 0.95 | 3+5,857E-8 | 3+8.240E-9  | 8.571E-16  |
| 180  | 50    | 0.05  | 5   | 1     | 1   | 30  | 0.3 | 0.95 | 3+7,629E-9 | 3+2.305E-9  | 9.103E-18  |
| 200  | 60    | 0.05  | 5   | 1     | 1   | 35  | 0.3 | 0.95 | 3+1,3E-9   | 3+2,509E-10 | 1.873E-19  |

For all the functions according to the selected criterion, the efficiency of the algorithm is caused by the increase in the values of the corresponding parameters. In turn, for unimodal functions, the accurate tracking of the global point of extremum is achieved at lower values of the parameters. It is similar for the ravine functions. In this case, the value of the parameters increases slightly. In the case of multi-extreme functions, judging by the dispersion value, the value of the configurable parameters should be higher than for unimodal and ravine ones.

Thus, it is clear that for each class of test functions its own set of parameter values will be optimal. In this regard, the urgent task is to select more universal parameters for all the classes of functions.

8. Analysis and comparison

Further the authors compare the proposed method (Araneidae algorithm, AA) with a number of algorithms that proved to be efficient for solving optimization problems: Particle Swam Optimization (PSO) [8], Independent Component Analysis (ICA) [9], Firefly Algorithm (FA) [10], Genetic Algorithm (GA) [11] and Artificial Bee Colony (ABC) [12].

The efficiency of the algorithms when searching the point of extremum on the test functions being considered was evaluated according to two criteria: the average number of calculations of the objective function (NFE) and the time of calculations. Figures 3-6 show graphs of NFE values (red) and NFE performance rating [13] (blue) for each of the compared algorithms on test functions. A series of 100 solutions to the same problem with the same parameter values was performed as well.
8

Figure 3. The graphic charts of NFE according to the Goldman-Price function.

Figure 4. The graphic charts of NFE according to the Ackley function.

Figure 5. The graphic charts of NFE according to the Rosenbrock function.

Figure 6. The graphic charts of NFE according to the Rastrigin function.

9. Conclusion
In the framework of this article, the algorithm developed by the authors based on the behavior of orb-web spiders was described. The concept of the method is simple, the algorithm does not contain complex calculations, which undoubtedly is its advantage. In the study of the algorithm, the selection of the parameter values of the proposed method was performed. The corresponding conclusions were made.

The authors studied the effectiveness of the proposed method as well. The algorithm was compared with other metaheuristic methods on test functions. The results of the study show that for the selected test functions the proposed method provides sufficiently high convergence rate and is rather competitive in comparison with the proven algorithms.

In the development of the work, it is planned to change all the settings of the algorithm at which the algorithm will be more efficient, as well as some parts of the algorithm can be changed in order to reduce computational complexity and increase the accuracy of the found solution.

References
[1] James J Q Yu and Victor O K Li 2015 A Social Spider Algorithm for Global Optimization Applied Soft Computing 30 614-27
[2] Gilyarov M S and Sokolov V E 1984 Life of Animals 3 (Moscow: Prosveshchenie) p 60
[3] Weise T 2011 Global optimization algorithms. Theory and Applications 3 1217
[4] Logunov D V and Kazenas V L 2015 The animals of Kazakhstan in photographs
[5] Gan W, Liu S, Yang X, Li D and Lei C 2015 Prey interception drives web invasion and spider size determines successful web takeover in nocturnal orb-web spiders Biol Open 4 1326–9

[6] Casas J 2011 Advances in Insect Physiology 41 (Cambridge: Academic Press) pp 175-262

[7] Gilyarov M S and Sokolov V E 1984 Life of Animals 3 (Moscow: Prosveshchenie) p 47

[8] Kennedy J and Eberhart R 1995 Particle swarm optimization International Conference on Neural Networks vol 4 pp 1942-8

[9] Kaveh A and Talatahari S 2010 Optimum design of skeletal structures using imperialist competitive algorithm Computers and Structures 88 1220-9

[10] Johari N, Zain A, Mustaffa N and Udin A 2013 Firefly Algorithm for Optimization Problem. Applied Mechanics and Materials Applied Mechanics and Materials 421 512

[11] Emel’yanov V V, Kurejchik V V and Kurejchik V M 2003 Theory and Practice of evolutionary modeling (Moscow: Fizmatlit) p 432

[12] Panteleev A V, Metlickaya D V and Aleshina E A 2013 The methods of global optimization: Metaheuristic strategies and algorithms (Moscow: Vuzovskaya kniga) pp 142-8

[13] Beiranvand V, Hare W and Lucet Y 2017 Best Practices for Comparing Optimization Algorithms Optimization and Engineering 18 815-48

(Almaty: Nur-Print) p 50)