Software of Irregular Multiextremal Multidimensional Mathematical Functions Generation for Testing the Evolutionary Optimization Algorithms

Rudolf Neydorf, Ivan Chernogorov
Department of Software Computer Technology and Automated Systems
Don State Technical University
Rostov-on-Don, Russia
Email: ran_pro@mail.ru, hintaivr@gmail.com

Dean Vucinic
Vesalius College
Vrije Universiteit Brussel (VUB)
Faculty of Electrical Engineering, Computer Science and Information Technology (FERIT)
Josip Juraj Strossmayer University of Osijek
Email: dean.vucinic@vub.ac.be, dean.vucinic@ferit.hr

***Abstract* - The paper studies the actual task of developing and setting up the algorithms for multi-extreme objects search optimization. To solve such problems, the heuristic methods are effectively used, in particular applying the swarming particles method. The mathematical base for the modified swarming particles method, which is oriented to solve the multi-extreme search problems, is developed and described in detail. The modified algorithm is applied to the irregular multi-extreme "Epsilon" test functions, considered to be a very difficult test case. The functionality of developed software "EpsilonFunction", which is created to control the generation, editing and investigation of multidimensional multi-extreme test functions is described. Epsilon function is a multiplicative function developed by R. Neydorf, which has fundamental extremes, multidimensionality and isolation in the factor space, which makes possible to completely exclude the influence of the results. The use of these fundamentally new test functions made possible to show that such modified method of swarming particles is suitable for solving the rather complex multi-extreme search problems. The developed software tool "EpsilonFunction" has a wide range of possibilities for developing and researching the test functions that are being created for other related applications. Epsilon-functions have found application in the method of "Cut-Glue" approximation of experimental data, which is at the researhing stage, but has already proved itself as an innovative and effective tool of data approximation.**

Keywords - search optimization; multi-extreme; method of swarming particles; test functions; irregularity; software.

I. INTRODUCTION

Many modern technical and scientific problems are complex, as they need to solve optimization problems [1][2]. Today, most of the known search engine optimization methods are designed and used to find one optimum, which is often the global one. However, the goal is not always to find only the global optimal solution. In many cases, there are many suboptimal and close to the global optimal extreme solutions, which are quite acceptable. To study such problems and find solutions applying the Multi-Extreme (ME) optimization, subject-oriented methods, as well as tools for testing and evaluation, are required.

When making decisions regarding ME, it is necessary to take into account that the deterministic search methods are usually very sensitive to their essential nonlinear continuum dependencies (in particular to discontinuity of their derivatives and variables). When searching the discrete quotient spaces, ME problems are often NP-complete [2]. In this regard, to solve complex (multidimensional and ME) optimization problems, more and more often various effective heuristic methods are applied.

The most important advantages of the heuristic algorithms, over other types of optimization algorithms, are in analogies that generated them. They consider the adaptation processes found in living and inanimate nature. Methodologically, they are based on processes found in the knowledge areas as decision-making theory, fuzzy logic, neural networks, evolutionary-genetic mechanisms, fleece behavior, etc. These processes partially repeat and in many ways supplement each other [3][4]. The disadvantages of these methods are that they are not adaptable to analytical research and evaluation.

Today, heuristic methods are used to solve problems of high computational complexity. One of the most promising representatives of such methods is the Method of Swarming Particles (MSP) [3]. However, the peculiarity of research and practical development of ME optimization algorithms are coming with their own complexity, cumbersomeness and significant development times, when a large number of extremes in the factor space of the related problem has to be solved.

The impossibility of a theoretical study of qualitative properties and numerical settings of heuristic algorithms implies that their performance and efficiency are most often checked with so-called Test functions (TFs) [5][6]. When algorithms for investigating ME objects are in development, the selection of effective testing tools is the problem. It is well known that TFs have either one global extreme, or they have a regular character with respect to the location of extremes, and the magnitude of their amplitudes [7]-[11]. Thus, for a more effective testing, the irregular multidimensional ME functions are needed.

The most famous and widely used ME optimization TFs are: Rosenbrock [8], Rastrigin [9], Himmelblau [10], De Jong [8], Griewank [8], Schwefel [8], etc. In addition, many papers describe other variants of TFs that generates ME functions [11]. They ensure a good verification of the ME optimization algorithms for the quality of the structural and

2018, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

parametric setup for the study of the factor space. In this context, a structural evaluation means the determination of the number of extremes and their spatial arrangement (coordinates). The parametric estimation means the determination of the extremes magnitudes (taking into account their signs).

The disadvantage of most TFs is their regular and analytical character. The absence of no differentiable or poorly differentiable areas greatly facilitates the work on the algorithm, by evaluating the surface structure under investigation. The real search is made difficult due to the fact that their coordinates are usually close to each other. The presence of a noticeable surface curvature at a respective extreme distance facilitates its search. Therefore, the TF extreme should be as close as possible to the impulse form, as in such case its neighborhood is minimally curved. A sufficiently developed adaptive algorithm can easily identify the period of the extremes alternation.

In Section II, the problem described in this paper is formulated. Section III contains a description of the Multiplicatively Allocating Function (MAF) and its characteristics. Section IV describes actual application of Epsilon-function. Section V illustrates the features of the developed special software (SW) for MAF building. Section VI describes the mathematical model (MM) of modified MSP for ME search. Section VII shows the result of experiments on the generated Epsilon TFs. Section VIII demonstrates the experimental results of MAF parameters influence optimization. Section IX contains the conclusion of the conducted research and future work.

II. PROBLEM FORMULATION

Following the above described issues, the goal of this paper is to develop and study the MSP modification, aiming to solve different ME search problems. For testing and setting a highly efficient solution to treat these problems, it is necessary to test the MSP on TFs, which are coming with disadvantages, as described in this paper introduction. To do this, it is necessary to implement algorithmically and programmatically the TF generator, which is theoretically presented in [5], and to conduct and process statistically representative experiments when setting up the modified MSP. In addition, the real implementation of developed TF and solve the ME searching problem by modified heuristic MSP is necessary to describe and demonstrate experiments result.

III. SCALABLE MAF FOR EXTREME FORMING

R. Neydorf et al. developed the general principles for constructing the universal irregular ME TFs, based on the application of MAF constructed to approximate problems [5]-[7].

MM of such MAF for N-dimensional ME TF, with a number of K extremes, has the form:

$$E(\vec{x}) = \sum_{k=1}^{K} [a_k \prod_{i=1}^{N} E_{x_i k}(x_i, x_{ik}, \Delta x_{ik}, \overrightarrow{\varepsilon_{ik}})] \qquad (1)$$

where: x − is an argument; α_k − is a coefficient specifying the extreme value; $\overrightarrow{\varepsilon_{ik}}$ - are the edge steepness parameters.

Figure 1 demonstrates the modeling of 3 ε-functions maxima in 2-dimensional space having different edge steepness of pulse fronts (2).

$$\begin{aligned} &E_{x_i k}(x_i, x_{ik}, \Delta x_{ik}, \overrightarrow{\varepsilon_{ik}}) = \\ &= [x_i - x_{ik}^L + \sqrt{(x_i - x_{ik}^L)^2 + (\varepsilon_{ik}^L)^2}\,]* \\ &*[x_{ik}^R - x_i + \sqrt{(x_{ik}^R - x_i)^2 + (\varepsilon_{ik}^R)^2}\,]/ \\ &/(4\sqrt{[(x_i - x_{ik}^L)^2 + (\varepsilon_{ik}^L)^2]\cdot[(x_{ik}^R - x_i)^2 + (\varepsilon_{ik}^R)^2]}) \end{aligned} \qquad (2)$$

where: $\{x_{ik}, \Delta x_{ik}, \overrightarrow{\varepsilon_{ik}}\}$ – is the set of TF parameters; $x_{ik}^L = x_{ik} - \Delta x_{ik}$, $x_{ik}^R = x_{ik} + \Delta x_{ik}$ – are the initial and final coordinates of extreme pulse for x argument; ε_{ik}^L and ε_{ik}^R – are the edge steepness parameters.

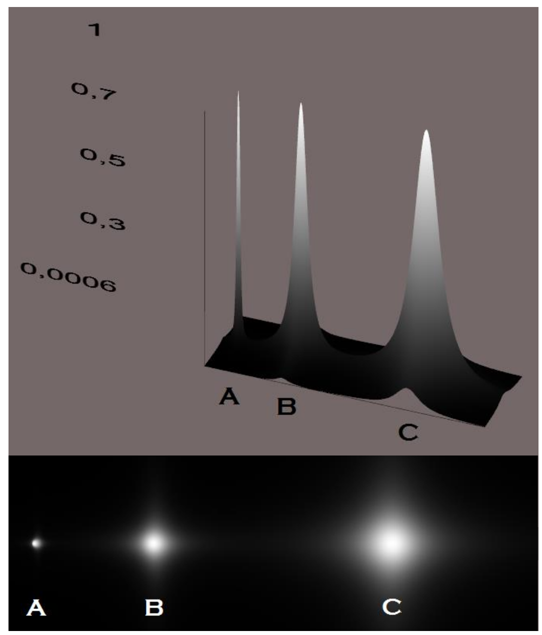

Figure 1. Demonstration of different steepness of pulse fronts of ε-function extrema

Variant A is impulse extreme ($\overrightarrow{\varepsilon_{ik}}$ =0.1), B is intermediate variant ($\overrightarrow{\varepsilon_{ik}}$ =0.5), C is shelving extreme ($\overrightarrow{\varepsilon_{ik}}$ =1). The graphs are constructed from (1) and (2).

IV. MAF APPLICATION

In the case of the first developed technical devices, technological processes and installations, which are largely created on the basis of heuristic representations of designers, analytical modeling may be generally inaccessible or give very unsatisfactory accuracy, as it is determined by many difficult factors to be taken into account. In this case, it is expedient to build models on the basis of their experimentally removed characteristics. Often such

2018, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

characteristics turn out to be essentially nonlinear. In many cases, when approximating the static characteristics, a piecewise approximation is sufficient. However, in dynamic models such a solution creates considerable difficulties in solving models, and with their mathematical transformations.

A "Cut-Glue Approximation" (CGA) method for constructing mathematical models of essentially nonlinear dependencies, applicable to fragmented EDs, was proposed in [3][6] and in a number of intermediate publications. The algorithm for implementing the CGA method consists of two relatively separate stages: the preparatory - directed on the fragmentation of the initial array of experimental data and the mathematical description of the resulting experimental data fragments (locally-approximating functions) and the multiplicative-additive stage - realizing the construction of mathematical model based on the received locally-approximating functions. Mathematically, this corresponds to cutting a single experimental data array into several fragments while preserving common boundaries. The boundaries are chosen from the condition of the required accuracy of the fragment description with the help of locally-approximating functions. In fact, the CGA method is oriented at narrowing the domains of the definition of analytic functions that approximate each of the point subsets - fragments. Feedback effects of the results of the subsequent stages of the CGA algorithm on previous ones are of great importance for the optimization obtained by the final mathematical model, but in this paper they are not considered. The most significant stage of the CGA, connected directly with the construction of the mathematical model, which is carried out using two operations, is investigated. The first of them is multiplicative ("Cut the fragments"), provides the formation of so-called interval-isolated functions (IIF), approximating the fragments within their boundaries. The IIF are formed by multiplying the locally-approximating functions by MAFs, the mathematical structure of, which provides interval isolation of the IIF while preserving its approximating properties within the fragment provided by the locally-approximating function. The second operation - the additive ("Glue the fragments"), makes the addition of IIF, multiplicatively approximating the fragments. The result of this additive operation is a smooth function that approximates the piecewise dependence with the required accuracy.

CGA method is based on the multiplicative "cutting out" of well-approximated sections of the modeled dependence and the additive "gluing" them together into a single analytic function [12]. The MAF is used for "cutting out", which determines the analytical properties of the final expression. The latter is the main distinctive feature and advantage of the method. allows not only numerical, but also analytical transformations of the obtained model.

"Cut the fragments" is a process in the CGA method that realizes the "cutting out" of a fragment that approximates some part of the experimental dependence on the boundaries of the selected fragment. Mathematically, this corresponds to cutting a single matrix of experimental data into several fragments while preserving common boundaries. The boundaries are determined by the condition for the accuracy of the description of the section by the approximating analytic function. To do this, we use MAF or so-called. the epsilon function of the steepness of the pulse fronts, which enter into the composition of the function, is ε. In previous works, the author gives the condition for the most effective variant of using this parameter. But such efficiency is conditioned only by the convenience of programming, and, as studies show, it is effective only if the boundaries of the fragment are approximated by a sufficiently accurate reproduction.

"Glue the fragments" is a process in the CGA method, which provides a single analytic function, describing the investigated area of the object's characteristics. Combining the fragments, i.e. Their gluing after the operation Cut the fragments is carried out by algebraic summation. After a series of preliminary experiments, the authors found that varying the values of ε for different coordinates in a multidimensional space can improve the approximation result for the Glue the fragments procedure.

However, to date, the effectiveness of the proposed solution of the approximation problem is theoretically justified and is practically confirmed only for one-dimensional and two-dimensional dependencies [12]. B The possibility and prospects of applying the proposed approach for approximating the dependencies of arbitrary dimension are justified. This significantly expands the scope of the method and its significance in the relevant field of science and practice.

Cut the fragments operation is described by the following general expression for the multiplicative transformation:

$$\forall i = \overline{1, N} \rightarrow f_i^n(\vec{x}) = \phi_i^n(\vec{x}) \cdot E_i^n(x_i, x_{ik}, \Delta x_{ik}, \overrightarrow{\varepsilon_{ik}}) \qquad (3)$$

where: $f_i(\vec{x}_i)$ - the IIF of the *i*-th experimental data fragment; n – is the index of the factor dimension of the experimental data; N – is number of fragments and their IIFs; $\phi_i(\vec{x}_i)$ - *i*-th locally approximating function; E_i^n - multidimensional MAF for the *i*-th locally approximating function, whose dimension n is determined by the factor dimension of the experimental data.

Glue the fragment operation is carried out by simple summation of all N IIFs (3):

$$F(\vec{x}) = \sum_{i=1}^{N} f_i^n(\vec{x}_i) \qquad (4)$$

MAF perform the "cutting out" operation from the *i*-th locally approximating function with minimal distortion of its fragment within the intervals specified for each coordinate of the quotient space of the arguments allocated to the experimental data by the fragment. For an exact multiplicative realization, this should be done by multiplying the internal locally approximating function data by one, and external ones by zero. Since it is impossible to do this in the framework of the postulated property of analyticity of the result of mathematical modeling, and, hence, the MAF, it is impossible to impose restrictions on the structure and

2018, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

parameters of these allocating functions associated with permissible locally approximating function distortions approximating the fragments. Based on this, the MAF is endowed with important for the solved task of "cutting out", properties that have been partially investigated by R. Neidorf in previous works [12] and a number of other earlier ones. They identified the main characteristics of MAF, allowing it to perform CGA postulated properties with respect to the internal point data of the fragments. However, additional studies by CGA on optimizing the MAF adjustment parameters and showed the need for a more thorough study of MAF boundary properties that affect the curvature of the boundaries of the cutting functions [12], and, through them, the error in approximating the obtained mathematical models both at the internal boundaries of the fragments closure and external boundaries of the whole experimental data array.

V. SOFTWARE IMPLEMENTATION OF *E*-GENERATOR

The SW for TFs creation is developed with C# programming language. It is a MAF research tool. The SW is a desktop application with third-party library for visualization. This library is a part of the executable file to simplify its execution.

The "Epsiolon Function" features are:

- Russian and English interface languages;
- Create / load / save / delete the test. The test is saved in the XML format. This feature allows user to use the resulting TF to effectively check the optimization algorithm within the third-party program without the use of additional technologies;
- Multidimensionality;
- Adding (editing) extremes in 2 modes: 1st - manual input and 2nd - pseudo-random generation of parameter values in the specified ranges;
- Display and save the resulting TF equation in analytical form;
- Validation of all input data;
- Visualization of the TF graph with additional setting the cut-off points to display multidimensional TF. 2-display modes: 2D and 3D graph.

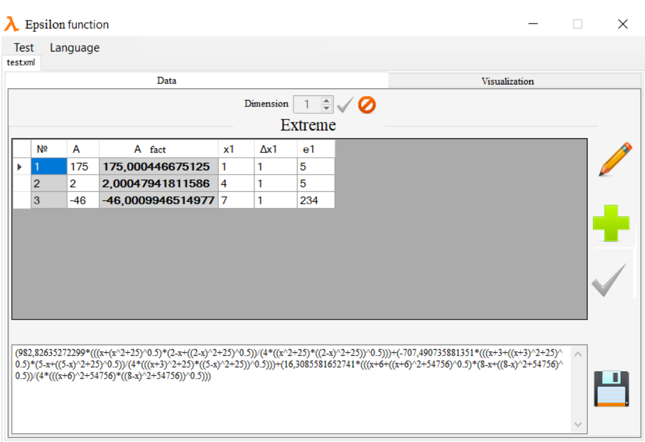

Figure 2. Input extreme software screen

Figure 2 illustrates the screen after creation/loading of test file. It is responsible for input main data – the extreme. The user can choose Dimension count and set up the extreme parameters.

Next sub-screens are necessary for adding the extreme. A user can choose the type of new extrema. In Empty mode the extreme will not have any data, as shown in Figure 3. In Custom mode the extreme will obtain pseudo-random data, as shown in Figure 4. The user chooses low and high bounds of randomization. In addition, user can simplify a process and check the specific check boxes, if he need same values for more than one parameters.

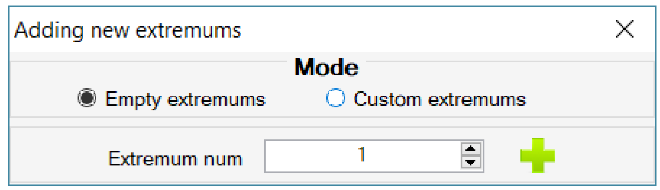

Figure 3. Add new empty extrema(e) software screen

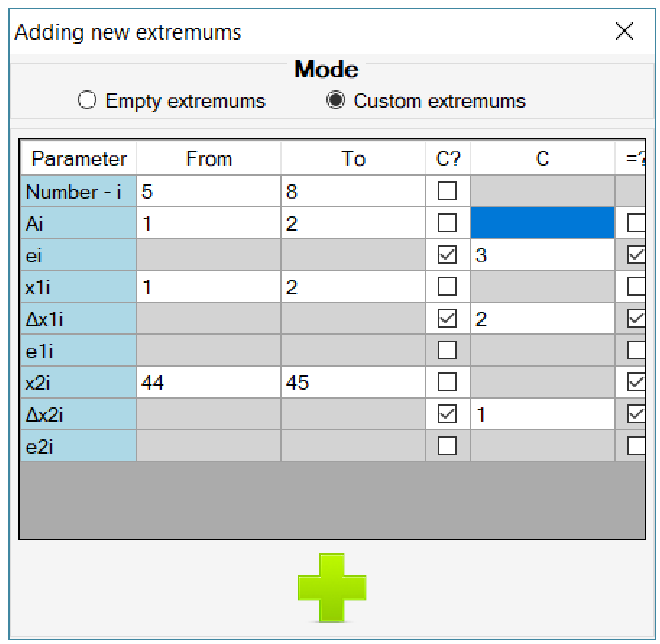

Figure 4. Add new pseudo-random extrema(e) software screen

Figure 5 demonstrates the next screen. It shows the 2-D graph of MAF and bounds of visibility. Extreme information shown only for convenience of research analytical studies. Any MAF in software can be shown as 2-D function, if the user turns on only one of the "Displayed?" parameters. For 3-D graph two of "Displayed?" parameters must be switched.

Figure 6 illustrates the previous screen, but MAF and displayed function has 2 dimensions (3-D graph). On this screen the user can rotate and scale the graph using mouse and special controller on the bottom-right side of the window.

Figures 7 and 8 illustrate the SW capabilities (on 2D and 3D models). Figure 7 shows a user function with 50 maxima (equal in magnitude of amplitudes, increments and steepness

2018, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

of pulse fronts) constructed by the SW. Figure 8 demonstrates a generated function with 31 maxima (different amplitudes, increments and steepness of the pulse fronts).

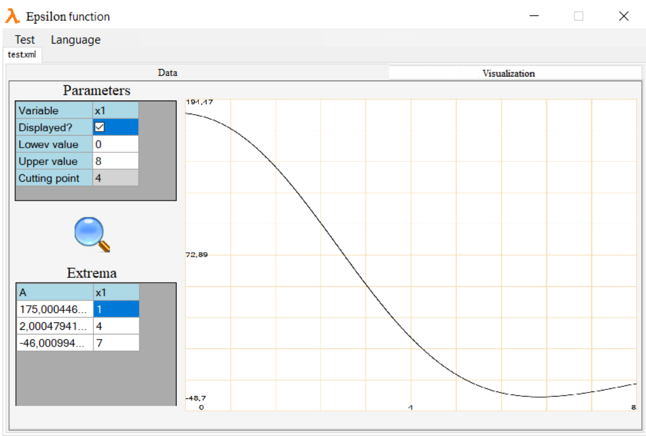

Figure 5. Software screen for analytical function analysis (2-D variant)

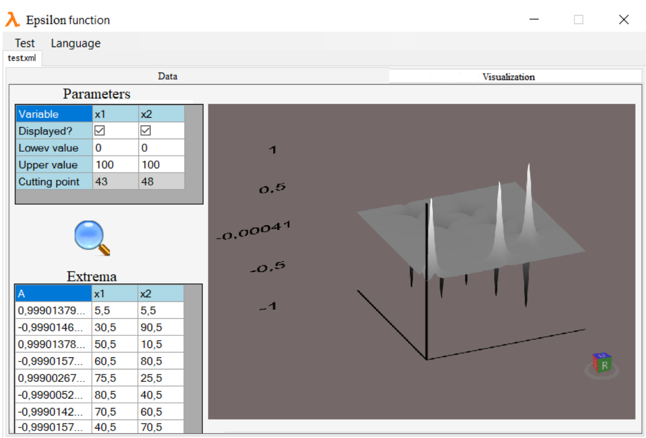

Figure 6. Software screen for analytical function analysis (3-D variant)

VI. MSP MM, MODIFIED FOR ME SEARCH

The essence and grounds for using MSP in search optimization problems are well known [2][3]. The classical MSP algorithm imitates the real group behavioral insects, birds, fish, many protozoa, etc. However, ME of optimization object requires some specific algorithm properties. Therefore, the canonical MSP version has been significantly revised and modified by the authors [3][4]. The hybrid algorithm includes basic algorithm laws of mechanics, dynamics, gravitation and stochastic "blurring" of the method parameters, which used in swarm prototype. In particular, its modification has been developed for solving ME problems in multidimensional spaces.

MSP MM is constructed on the basic equations of kinematic motion of a material point for particle position and velocity:

$$\vec{X}_{ti} = \vec{X}_{(t-\Delta t)i} + \vec{V}_{(t-\Delta t)i} \cdot \Delta t \qquad (5)$$

$$\vec{V}_{ti} = \vec{V}_{(t-\Delta t)i} + \vec{A}_{(t-\Delta t)i} \cdot \Delta t \qquad (6)$$

where: $\vec{X}_{(t-\Delta t)i}$ - is previous particle position; $\vec{V}_{(t-\Delta t)i}$ - is previous particle velocity; Δt - is time interval (iteration); $\vec{A}_{(t-\Delta t)i}$ - is particle acceleration at previous iteration, where:

$$\vec{A}_i = \sum \frac{\hat{D}_i^Q g^Q m_i^Q}{(r_i^Q)^2 + (\varepsilon^Q)^2} - \\ -\mu_{vis} \vec{V}_{(t-\Delta t)i} - \mu_{tur} \left|\vec{V}_{(t-\Delta t)i}\right| \vec{V}_{(t-\Delta t)i} \qquad (7)$$

where: $\sum \frac{\hat{D}_i^Q g^Q m_i^Q}{(r_i^Q)^2 + (\varepsilon^Q)^2}$ - is the acceleration caused by the bio-analog of particles gravitational attraction to the extreme point, $Q \in \{G, L_i\}$, G - is the particle attraction to the global swarm extreme; L_i - the best found position by particle for all time; $\hat{D}_i^Q$ - is the unit director vector towards the point of attraction; g^Q - is the gravitational constant prototype; m_i^Q - is the gravity center mass; r_i^Q - is the distance between particle position and diffuse position of the attraction target point; ε^Q - is a natural acceleration limiter that excludes the passage of any material point at $\Delta X<\varepsilon$ distance; $-\mu_{vis}\vec{V}_{(t-\Delta t)i}$ - is the viscosity friction; $-\mu_{tur}\left|\vec{V}_{(t-\Delta t)i}\right|\vec{V}_{(t-\Delta t)i}$ - is the turbulent friction; μ_{vis}, μ_{tur}- are the coefficients of viscosity and turbulent friction, respectively.

To take into account the MM stochastic behavioral components, the equation of parameters random fluctuation (distortion) is included:

$$\lambda^{\xi}(\varphi) = \lambda \cdot (1 + 2\varphi \cdot (rnd(1) - 0.5)) \qquad (8)$$

where: ε - is the nominal value of fluctuating parameter; φ - is the coefficient of parameter distortion, relative to the nominal value; $rnd(1)$ - is the random float number in [0;1] range. This law applies to the following collective parameters of a swarm and particles:

- Prototypes of gravitational constants - g^Q;
- Coefficients of viscosity and turbulent friction - μ_{vis} and μ_{tur};
- Dissipation coefficient - μ_{dis}.

VII. MSP MODIFICATION FOR E TFS APPLICATION

To study and adjust the ME modification of MSP, 3 demonstration Epsilon TFs are generated using the "Epsilon Function" SW; see Figures 9(a), 9(b) and 9(c). In addition, to test the MSP modification, an appropriate "Modified MSP" SW was developed. For its development, the C # programming language was used.

For all experiments, the same particle number (P) and iteration (I) settings were used, to obtain a more general picture of MSP operation on various generated functions. At

2018, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

the same time, the dynamics parameters were settings dynamically, with respect to the region under consideration.

Figures 9(d), 9(e) and 9(f) show Epsilon functions and localized MSP regions (red squares) and extremes (blue dots), which are found and evaluated. Each function has a specific feature that allows you to identify the positive aspects and disadvantages of the optimization algorithm being developed.

Figures 9(a) and 9(d) show a TF with 5 minima and 3 maxima. The complexity of the extremes search for a given function can be characterized as an average. The functions are steep near extremes and moderately canopies at the bases, and being located at a considerable distance from each other. However, the amplitude of the extremes is not high (-1), and it is not easy to identify the whole set of extremes from the first pass.

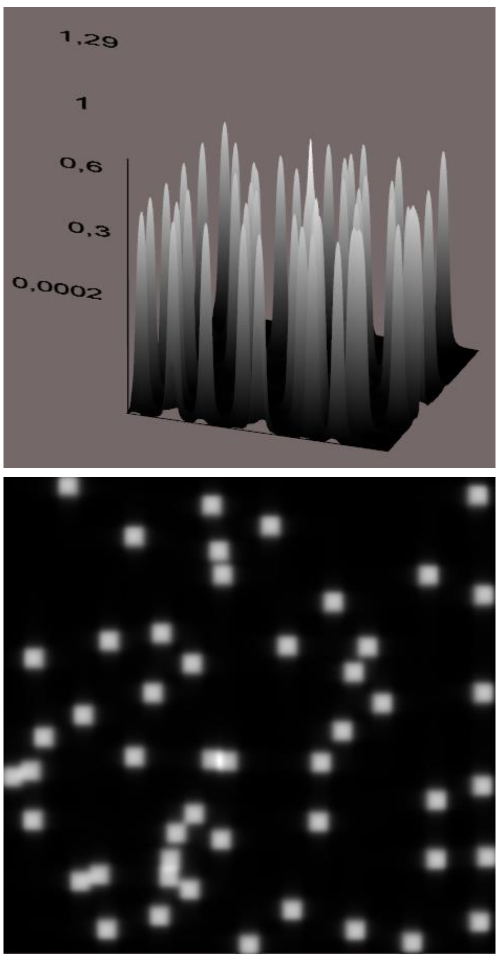

Figure 7. Demonstration of different steepness of pulse fronts of ε-function extrema

Figures 9(c) and 9(f) show the Epsilon function, which has 8 minima and 8 maxima to identify and estimate the minima. The shape of this function is similar to the bends of "peaks" and "gorges", which can be smooth, but may have sharp cliffs. An additional complication in this function is the large difference in the extremes amplitudes. By localizing one of the extremes, the multi-agent system is not exploring the rest of the search space. However, this does not happen in the modified MSP.

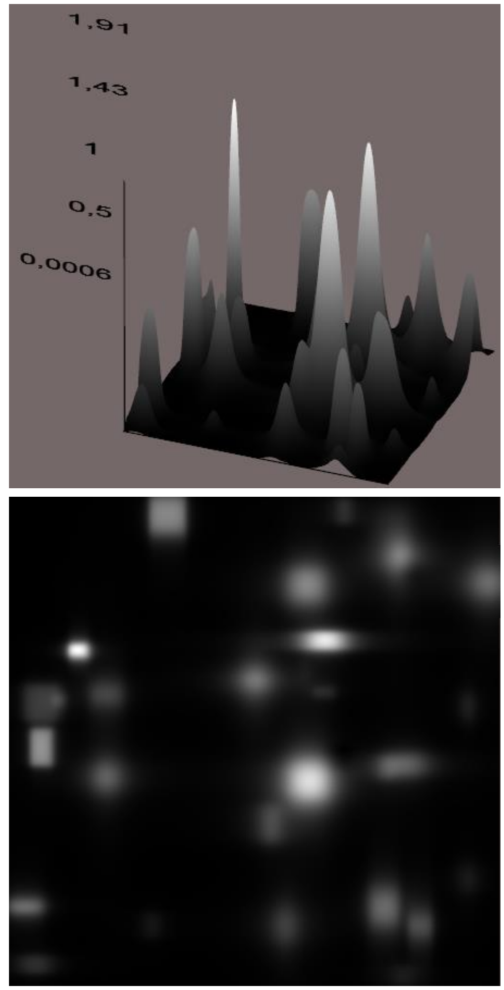

Figure 8. Example of generated ME TF with impulses of different steepness

Tables I-III show the experiments results of a successful search for the modified-MSP. These results were obtained from the basic MM motion of the swarm (preceding the MM clustering mechanism [13], which divided the search space into subspaces and found in each an extreme and, which was replaced by the dynamic clustering caused by the behavioral model of the swarm itself). This made it possible to approximate the MM method to the real prototype of the agent's interaction (insects, birds, fish, etc.) in the swarm. The agents localize extreme areas, under the influence of attraction forces (not only global, but also local). The increase in the influence of local attraction is caused, in particular, by the introduction of a turbulent deceleration in the MM. The removal of the non-dynamic clustering mechanism from MM also enabled to exclude the "cluster" attraction of the swarm particles to the closest previously created clusters, which allowed the particles to behave in a more similar way to the real prototype.

2018, © Copyright by authors, Published under agreement with IARIA - www.iaria.org

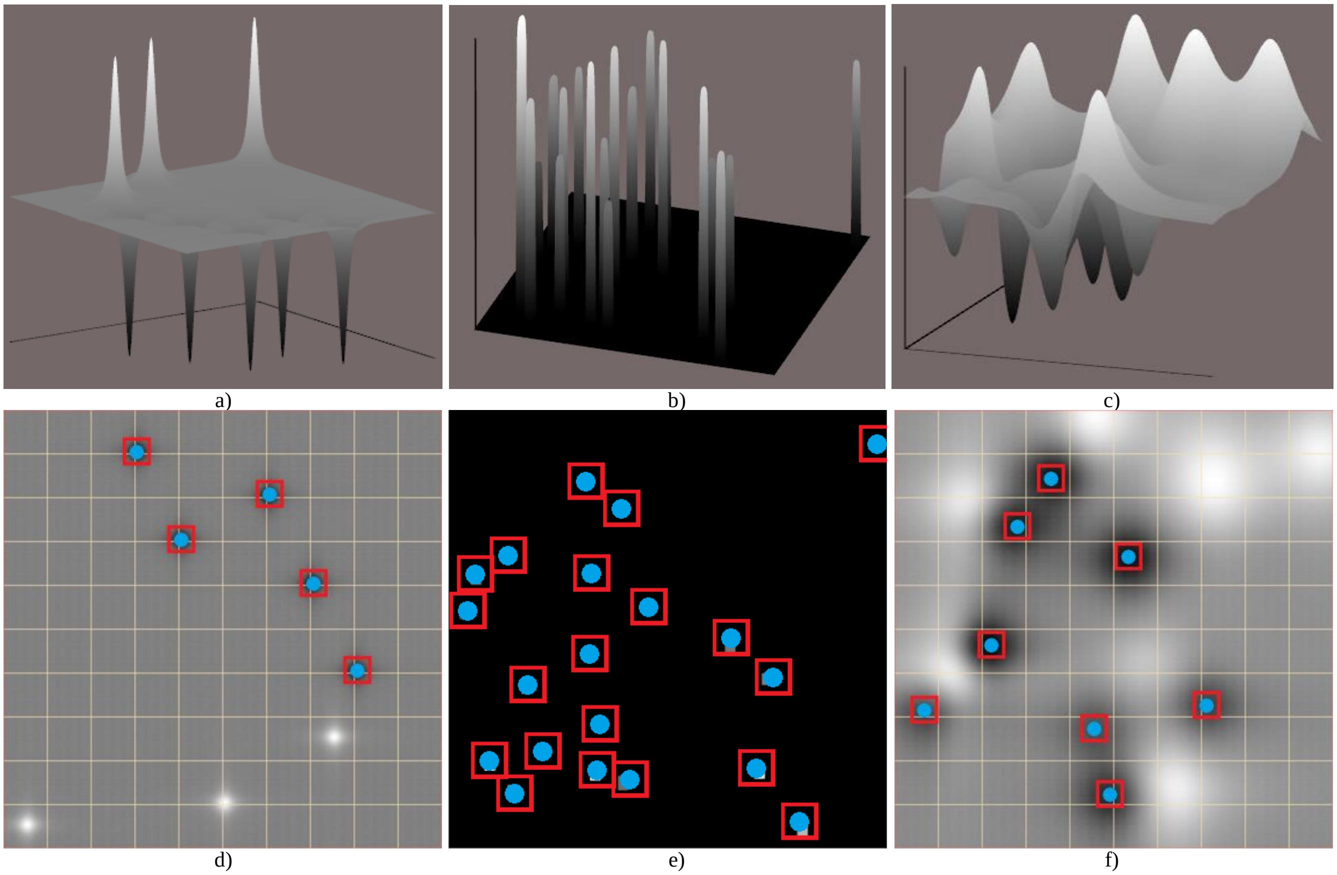

Figure 9. Generated TF graphs and MSP result on different scenes: *a* – I on 3D, *b* – II on 3D, *c* – III on 3D, *d* – I on 2D, *e* – II on 2D, *f* – III on 2D

As a result, the minimum error of the obtained approximation in experiment 1 (see Table I), relative to the standard, was ~0.01%, average ~0.03%. MSP successfully isolated the extreme regions and obtained the described results due to the smooth motion of the particles to the extreme values found at the moment (based on (3)-(5)). This allowed the particles not to jump through the extremes.

The minimum error of the obtained approximation in experiment 2 (see Table II), relative to the standard, turned out to be ~0.001%, mean ~0.01%. The parameter of the slope of the pulse fronts of a given Epsilon function for all extremes is 0.01, which implies the complexity in finding them. However, since the number of extremes is 20, the particles interact with each other and receive an additional opportunity to study the neighboring extremes. This effect is due to the fact that, when the particle is found close to a extreme, then, in the next step, it will get a large acceleration (see (5)), which will allow the particle to escape from this extreme attraction zone and visit the extreme region of the neighboring one. The minimum error of the approximation obtained in experiment 3 (see Table III), relative to the standard, was ~0.09%, average ~3%.

The complexity of this experiment consists in mixing maxima and minima. This means that the particles will be located more often in positions that may be worse than their previous ones. However, the method is also effective in such a case.

TABLE I. I *E* TF STANDARD AND MSP RESULT

Standard			**MSP**		
x	*y*	*f(x, y)*	*x*	*y*	*f(x, y)*
70.5	60.5	-1	70.4811	60.4904	-0.9998
30.5	90.5	-1	30.5206	90.5171	-0.9998
60.5	80.5	-1	60.4901	80.5104	-0.9999
80.5	40.5	-1	80.5486	40.5173	-0.9993
40.5	70.5	-1	40.5158	70.4641	-0.9996

The complexity of this experiment consists in mixing maxima and minima. This means that the particles will be located more often in positions that may be worse than their previous ones. However, the method is also effective in such a case.

The attraction of particles to the global extreme allows improving the result of the whole swarm, even in a situation where the best position of the particle itself is not a local extreme (which forces the particle to swarm in the pseudo-local area). With additional sub-optimization of the parameters of the swarm and particles, the error can be significantly reduced [3][4].

The problem of finding the set of MAF extrema is important in the CGA and at the moment there are no unambiguous results on the benefits of searching and using only global MAF extremum. In one of CGA operation each MAF affects to distortion in neighboring fragments by varying measures.

TABLE II. II *E* TF STANDARD AND MSP RESULT

Standard			MSP		
x	*y*	*f(x, y)*	*x*	*y*	*f(x, y)*
6.4254	4.6182	0.4543	6.4253	4.6182	0.4543
3.2322	4.4678	0.7859	3.2322	4.4698	0.7859
9.7602	9.2187	0.6206	9.7505	9.2172	0.6206
1.3463	6.6313	0.5903	1.3412	6.6291	0.5903
3.9888	1.4936	0.4183	3.99	1.4981	0.4183
…					
4.5307	5.4641	0.7796	4.5327	5.4604	0.7796
2.1397	2.2475	0.5101	2.1423	2.2536	0.5101
3.3574	1.6594	0.8659	3.3518	1.6565	0.8659
0.9593	1.9634	0.9677	0.9597	1.9561	0.9677
1.4741	1.2706	0.7321	1.473	1.2718	0.7321
7.2637	3.8593	0.5123	7.2631	3.8621	0.5123

TABLE III. III *E* TF STANDARD AND MSP RESULT

Standard			MSP		
x	*y*	*f(x, y)*	*x*	*y*	*f(x, y)*
3.6721	8.491	-0.5598	3.6181	8.451	-0.5651
5.3615	6.6256	-0.5653	5.343	6.6296	-0.5658
0.7982	3.1601	-0.2936	0.7953	3.1288	-0.2901
4.9938	1.1940	-0.4426	4.8912	1.1833	-0.455
4.5671	2.7411	-0.2833	4.5894	2.6884	-0.2849
2.025	4.5505	-0.5831	2.1631	4.6302	-0.6375
2.6129	7.4111	-0.4821	2.769	7.3498	-0.5187
7.0786	3.2646	-0.3418	7.1416	3.2602	-0.3451

Finally, to demonstrate the implementation of MAF and modified under the search MSP in a more complex task an example of experimental data approximation is further described.

VIII. EXPERIMENTAL RESEARCH OF MAF PARAMETERS INFLUENCE ON THE APPROXIMATION QUALITY IN IIF GLUING PROCESS

The initial data for pilot research were generated by the authors. As a result, the parameters of conditional experiment were chosen: the equation of local approximation functions in form of a 3-degree polynomial with two variables and the corresponding coefficients for them:

$$F(x,y) = b_0 + b_1 x + b_2 y + b_{11} x^2 + b_{12} xy + b_{22} y^2 + \\ + b_{111} x^3 + b_{112} x^2 y + b_{122} x y^2 + b_{222} y^3 \qquad (9)$$

where $b_0 = 200$, $b_1 = 8$, $b_2 = -12$, $b_{11} = 1.95$, $b_{12} = -0.18$, $b_{22} = 1.72$, $b_{111} = 0.08$, $b_{112} = 0.023$, $b_{122} = -0.08$, $b_{222} = 0.1$ and vector *x* of dimension 10, a vector *y* of dimension 8.

Resulting matrix is divided into 4 adjacent areas (fragments) with generic boundaries. Figure 10 shows the initial experimental data. The rows correspond to the values from *x* vector, the columns correspond to the values from *y* vector. Figure 11 illustrates graphs of the source data in two variations.

For each fragment, regression equations of 2-degree with the corresponding coefficients *b* are obtained (see Table IV).

$$\mathbf{y} = \begin{pmatrix} -5.2 & -3.6 & -2 & -0.4 & 1.2 & 2.8 & 4.4 & 6 \end{pmatrix}$$

$$\mathbf{F(x,y)} = \begin{pmatrix} 380.824 & 342.145 & 310.674 & 288.87 & 279.19 & 284.092 & 306.033 & 347.471 \\ 343.891 & 305.568 & 273.716 & 250.794 & 239.259 & 241.568 & 260.18 & 297.551 \\ 315.786 & 278.058 & 246.064 & 222.262 & 209.11 & 209.066 & 224.586 & 258.128 \\ 297.209 & 260.314 & 228.417 & 203.974 & 189.444 & 187.283 & 199.95 & 229.903 \\ 288.86 & 253.037 & 221.475 & 196.629 & 180.959 & 176.921 & 186.974 & 213.574 \\ 291.439 & 256.927 & 225.937 & 200.928 & 184.356 & 178.68 & 186.356 & 209.843 \\ 305.646 & 272.682 & 242.504 & 217.569 & 200.334 & 193.258 & 198.797 & 219.409 \\ 332.18 & 301.004 & 271.876 & 247.254 & 229.594 & 221.356 & 224.996 & 242.972 \\ 371.742 & 342.591 & 314.752 & 290.681 & 272.836 & 263.674 & 265.653 & 281.231 \\ 425.031 & 398.145 & 371.832 & 348.551 & 330.758 & 320.911 & 321.469 & 334.887 \end{pmatrix} \qquad \mathbf{x} = \begin{pmatrix} -9.6 \\ -7.8 \\ -6 \\ -4.2 \\ -2.4 \\ -0.6 \\ 1.2 \\ 3 \\ 4.8 \\ 6.6 \end{pmatrix}$$

1 2 3 4

Figure 10. Matrix of experimental data

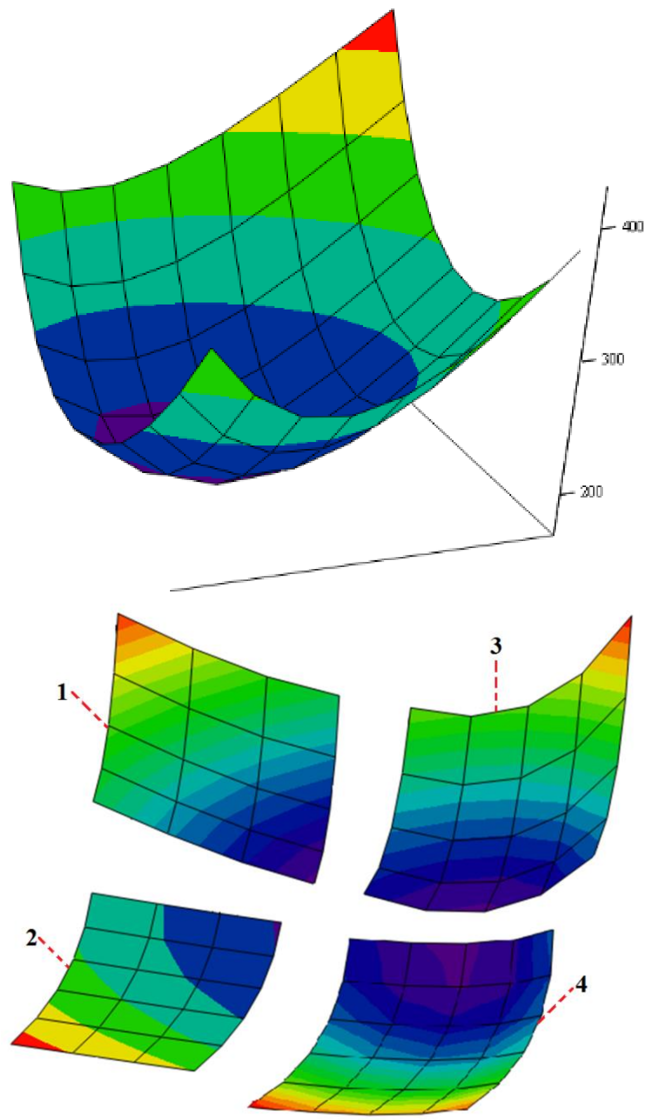

Figure 11. Full and piecewise experimental data representation

In accordance with the formulation of the problem, (ε_{xi}, ε_{yi}) ∈ (0, 1], i ∈ {1,2,3,4} are selected for the experiments, number of iterations - 300, number of particles - 100. The rest parameters of MSP MM were set up under the task. The criterion for estimating and minimizing the errors of approximation of entire dependence and fragment in the preliminary analysis are the mean-square deviation (MSD) and the maximum error in absolute error matrix. Different estimates, which can be several resulting mathematical constructions obtained as a result of CGA. Firstly, these include the whole matrix of experimental data. Secondly, it applied without boundaries values. Third, the common faces of the “glued” fragments not including the border values of whole matrix (these elements are most affected on ε values). Fourth and fifth, separately considered common edges of “glued” fragments (vertical and horizontal).

The agents of MSP search in 8-dimensional space. The results of experiments on two selected criteria and five allocated areas in the matrix of absolute errors are displayed in Table V.

Total computing time of the algorithm spent on conducting experiments with 300 iterations is ~1.1 sec. Computing time of the algorithm spent on finding the extrema described in Table V on ~160 iterations is ~0.7 sec. As can be seen from Table V obtained values of MSD, Max, ε_x and ε_y are extremely dependent on the criterion under consideration and the selected range of values in absolute error matrix.

The results shown in Table VI obtained by optimizing ε for each fragment in the pilot experiments and optimizing ε for all fragments in current studies.

Total computing time of the algorithm spent on conducting experiments on 300 iterations is ~1.1 sec. The running time of the algorithm spent on finding the extrema described in Table VI on ~160 iterations is ~0.7 sec. As can be seen from Table VI, the obtained values of MSD, Max, ε_x and ε_y are extremely dependent on the criterion and selected range of values in the absolute error matrix too.

Comparison of data showed following results: total computing time of the algorithm spent on conducting experiments on 300 iterations decreased by ~50%, MSD and maximum error in the absolute error matrix decreased by ~17% and ~8.3%, respectively.

TABLE IV. RESULTING COEFFICIENTS OF REGRESSION EQUATIONS

b^1	b^2	b^3	b^4
b_0	-0.008	-0.904	0.3646
b_1	1.36	3.040	0.712
b_2	-11.82	-16.84856	-14.65208
b_{11}	196.6	199.966688	198.317312
b_{12}	5.6587	7.05072	8.70432
b_{22}	1.5256	1.6544	2.0116

TABLE V. MAF INFLUENCE OPTIMIZATION RESULT

Values range in the absolute errors matrix	Criteria	MSD,%	Max,%	εx1	εy1	εx2	εy2	εx3	εy3	εx4	εy4
All	MSD	9,14	26,93	$1*10^{-6}$	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	$1*10^{-6}$	0,99	0,99
	Max	9,39	26,75	$1*10^{-6}$	0,99	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	0,99	$1*10^{-6}$
All except boundaries	MSD	5,62	26,87	$1*10^{-6}$	0,99	0,7	0,44	$1*10^{-6}$	8*10-6	0,99	0,74
	Max	5,77	26,75	$1*10^{-6}$	0,99	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	0,99	$1*10^{-6}$
Fragments boundaries	MSD	2,98	28,51	$1*10^{-6}$	0,81	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	0,47	$1*10^{-6}$
	Max	3,1	26,75	$1*10^{-6}$	0,99	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	0,99	$1*10^{-6}$
Fragments boundaries horizontal	MSD	3,65	29,16	0,99	0,99	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	0,99	$1*10^{-6}$
	Max	3,71	26,77	$1*10^{-6}$	0,99	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	0,99	$1*10^{-6}$
Fragments boundaries vertical	MSD	0,39	35,49	$1*10^{-6}$	0,99	0,996	$1*10^{-6}$	0,25	0,99	0,99	$1*10^{-6}$
	Max	1,06	26,75	$1*10^{-6}$	0,99	0,99	0,99	$1*10^{-6}$	$1*10^{-6}$	0,99	$1*10^{-6}$

TABLE VI. COMPARING OF MAF INFLUENCE OPTIMIZATION EXPERIMENTS

			Experiment			
			Pilot	Current	Pilot	Current
		Criteria	*MSD,%*		*Max,%*	
Values range in the absolute errors matrix	*All*	*MSD*	11,08	9,14	32,35	26,93
		Max	11,57	9,39	29,7	26,75
	All except boundaries	*MSD*	7,65	5,62	31,42	26,87
		Max	7,88	5,77	29,71	26,75
	Fragments boundaries	*MSD*	3,74	2,98	30,07	28,51
		Max	3,81	3,1	29,7	26,75
	Fragments boundaries horizontal	*MSD*	4,32	3,65	30,07	29,16
		Max	4,36	3,71	29,7	26,77
	Fragments boundaries vertical	*MSD*	0,54	0,39	33,49	35,49
		Max	0,9	1,06	29,7	26,75

IX. Conclusion and Future Work

The Epsilon Function SW developed in this paper has proven to be an effective tool for the generation of the irregular multi-dimensional ME TFs. The easy-to use and convenient interface to access the multi-functional SW allows the fast generation and qualitative research of TFs. The SW functions do not have an obvious regular and analytical character, like the set of the existing ME optimization TFs.

Experiments carried out on TFs showed that the developed MSP modification allows to localizing the extreme areas of the nonstandard irregular ME Epsilon function, having the approximation error from ~0.001% to ~3%.

Experimentally obtained results allow the validation of the developed MSP modification, and prove to be an effective tool in searching the extremes of heterogeneous generated TFs.

When optimizing ε for each fragment the MSD and maximum error in absolute error matrix of the solved examples decreased by ~17%, and ~8.3%, respectively, relative to the results of general optimization ε for the entire set of fragments obtained in pilot research.

Our main research task is to create a modification of the heuristic method of swarming particles and use it during one of the stages of the author's Cut-Glue approximation for highly nonlinear dependencies.

The development of this generator, in addition to the presented advantages is associated with the possibility of creating irregular multidimensional ME TFs on, which the MAF modification is processed. It helps to further investigate the properties of TFs, when applied in different domains, thus allowing more accurate picture and better results for the main study of the overall "Cut-Glue" approximations approach.

Acknowledgment

The research supported by the Russian Foundation of Fundamental Research, project No. 18-08-01178/18 A.

References

[1] R. Neydorf, I. Chernogorov, and D. Vucinic, “Search Opportunities of Swarming Particles Methods in Irregular Multi-Extreme Environments,” The 11th International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP2017) IARIA, Barcelona, Spain, 2017, pp. 7-12, ISSN: 2308-4499, ISBN: 978-1-61208-599-9.

[2] R. Shreves, “Drupal search engine optimization,” Birmingham: Packt Publishing LTD, 2012.

[3] R. Neydorf, I. Chernogorov, V. Polyakh, O Yarakhmedov, Y. Goncharova et al., “Study of Search Optimization Opportunities of Heuristic Algorithms for Solving Multi-Extremal Problems,” The 10th International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP2016) IARIA, Venice, Italy, 2016. pp. 44-51, ISSN: 2308-4499, ISBN: 978-1-61208-506-7.

[4] R. Neydorf, I. Chernogorov, V. Polyakh, O. Yarakhmedov, J. Goncharova et al., “Formal Characterization and Optimization of Algorithm for the Modelling of Strongly Nonlinear Dependencies Using the Method "Cut-Glue" Approximation of Experimental Data,” SAE Technical Paper, 2016, doi:10.4271/2016-01-2033.

[5] R Neydorf, “Advanced test function for studying the multi-extremal problems and solutions,” The 6th International Seminar "System analysis, Management and Information processing", Rostov-on-Don, Russia, 2015, pp. 6-14, ISBN: 978-5-7890-1077-8.

[6] R. Neydorf, “Bivariate "Cut-Glue" Approximation of Strongly Nonlinear Mathematical Models Based on Experimental Data,” SAE Int. J. Aerosp. 8(1), 2015, pp. 47-54, doi:10.4271/2015-01-2394.

[7] R. Neydorf and I. Chernogorov, “Universal Generator of Irregular Multidimensional Multiextremal Test Functions,” The 30th International Scientific Conference Mathematical methods in technique and technologies (MMTT-30), vol. 2, 2017, pp. 138-143, ISSN: 2587-9049, ISBN: 978-5-7422-5772-1.

[8] M. Molga and C. Smutnicki, “Test functions for optimization needs,” Technical Report, Institute of Computer Engineering Control and Robotics, Wroclaw University of Technology, Wroclaw, Poland, 2005.

[9] L. A. Rastrigin, “Systems of extremal control,” Nauka Publishing House, Moscow, Russia, 1974.

[10] D. Himmelblau, “Applied Nonlinear Programming,” McGraw-Hill, New York, USA, 1972, ISBN: 978-0-0702-8921-5.

[11] M. Laguna and R. Marti, “Experimental Testing of Advanced Scatter Search Designs for Global Optimization of Multimodal Functions,” J. of Global Optimization archive, vol. 33, issue 2, 2005, pp. 235-255, doi:10.1007/s10898-004-1936-z.

[12] R. A. Neydorf and I. V. Chernogorov, “Study of the multiplicatively separating function boundary properties in the Cut-Glue approximation problem,” The 31th International conference “Mathemathical methods in technique and technology” (MMTT-31), Russia, Saint-Petersburg, vol 5, 2018, pp. 31-41, ISSN: 2587-9049, ISBN: 978-1-5386-5824-6.

[13] Roman. *Clustering. Algorithm A-quasiequivalence*. [Online]. Available from: https://habrahabr.ru/post/124978/ 2018.11.12.

2018, © Copyright by authors, Published under agreement with IARIA - www.iaria.org