

## SOME NEW HEURISTIC ALGORITHMS FOR GLOBAL OPTIMIZATION

I. Michinev<sup>1</sup>, F. J. Keil<sup>2</sup>, S. Stoyanov<sup>1</sup>

<sup>1</sup> University of Chemical Technology and Metallurgy  
8 Kl. Ohridski, 1756 Sofia, Bulgaria  
E-mail: stoyan@uctm.edu

Received 15 September 2007  
Accepted 15 November 2007

<sup>2</sup> Hamburg University of Technology  
Eissendorfer Strasse 38, D-21071 Hamburg, Germany,  
E-mail: keil@tu-harburg.de

---

### ABSTRACT

Some new heuristic optimization algorithms are introduced and compared with known reliable algorithms for global optimization problems. The efficiency of the algorithms is estimated on the basis of the convergence, average number of function evaluations and the probability of finding the global optimum. The optimal number of starting points for some algorithms is also investigated.

*Keywords:* global optimization, heuristic algorithms, clustering analysis.

---

### INTRODUCTION

A very rapid development of new algorithms for global optimization was going on over the last decades. The main reasons are the necessity to find the global optimum in various practical problems and the rapid computer development. Heuristic algorithms [1, 2, 3, 4, 5], genetic algorithms [1, 6], algorithms based on clustering analysis [7], and their combinations are the main new directions in global optimization techniques developed over the last few years. To find a new reliable algorithm which can solve a large variety of global optimization tasks with a minimum number of function evaluations is still an open problem. The current developed investigation is a step in this direction. Three new heuristic algorithms will be described and investigated.

### METHODOLOGY OF INVESTIGATION

#### *Problem Formulation*

Let us consider the problem of determining the optimum vector  $\mathbf{x}^*$  to maximize the continuous objective function of  $n$  control variables

$$\max_{\mathbf{x}} Q(\mathbf{x}) = \max_{\mathbf{x}} [f(x_1, x_2, \dots, x_n)], \quad (1)$$

$$\text{subject to constraints of the control variables} \\ \mathbf{x} \in X^n \quad (2)$$

The investigation is carried out with 15 test functions given on Table 1 [5, 8, 9, 10] with a number of local optimums between five and several thousands, and dimensions of control variables between  $n = 2$  (8 functions),  $n = 3$  (4 functions), and  $n = 5$  (3 functions). Each of the investigated algorithms is started 10000 times from different sets of initial points, generated randomly in the feasible region of the control variables for every objective function. The probability of finding the global optimum is investigated (*PFGO*) for every set of different numbers of initial points. The *PFGO* is estimated as a ratio of solved problems with respect to the whole number of starts expressed in percent. The average number of function evaluations (*ANFE*) is calculated only for the solved tasks. The termination criteria for the investigated algorithms are the following:

$$\|\mathbf{x}_{best} - \mathbf{x}_{worst}\| < \varepsilon_x, \quad (3)$$

where  $\mathbf{x}_{best}$  and  $\mathbf{x}_{worst}$  are the vectors with coordinates of the points with the best and the worst result for the objective function respectively. In some algorithms also other termination criteria are used. The value of  $\varepsilon_x = 0.0001$  is employed for all tasks.

### Investigated algorithms for global optimization

The following algorithms for global optimization have been investigated:

- (1) Multicomplex method [11];
- (2) Complex method [11, 12];
- (3) Moving constraints method [11];
- (4) Newly proposed algorithm PARSEC-1;
- (5) Two runs complex method 2S [11];
- (6) Newly proposed algorithm PARSEC-2;
- (7) Newly proposed algorithm CLUSTER ANALYSIS;
- (8) Tunneling algorithm [13, 14];
- (9) Price algorithm [15, 16, 17].

### NEW PROPOSED ALGORITHMS FOR GLOBAL OPTIMIZATION

Two variants of new heuristic algorithms for global optimization called PARSEC 1 and PARSEC 2 (Parallel Adaptive Random Search with Estimation of Convergence) and the algorithm CLUSTER ANALYSIS are proposed in this paper. The structures of the algorithm PARSEC is given in Fig. 1, and for CLUSTER ANALYSIS in Fig. 2. The algorithms are written for searching the global maximum. The description of the parameters and their recommended values are as following:

#### Common parameters

- $n$  Number of control variables (In the block diagram  $N$ );
- $NF$  Number of initial points;
- $x(i,j)$  Vector coordinates of each point,  $i = 1, NF; j = 1, n$ ;
- $Q(i)$  Value of the objective function in each point  $i = 1, NF$ .

#### Parameters for the PARSEC algorithm

- $NF$  Number of points in the beginning of each of the iterations (Initial number of points);
- $NP$  Current number of points;

$FP$  Number of the points with best results for the objective function. Among  $FP$

the closest points are chosen ( $FP = 0.9NF$ );

$NCP$  Number of the closest best points from which  $\mathbf{x}_c$  is calculated (see eq.(4))

( $NCP = 0.9NF, \min\{NCP\} = 2$ );

$\alpha$  Coefficient determining the regional distribution of the points ( $\alpha = 1.01$ );

$\beta$  Coefficient determining the reiterative points ( $\beta = 0.95$ );

$\gamma$  Coefficient for increasing the number of points ( $\gamma = 1.00$ );

$\delta$  Coefficient for reduction of the number of points ( $\delta = 0.75$ );

$NFmax$  Maximal number of initial points;

$RO$  and  $RN$  Old  $RO$  and new  $RN$  value of the average radius to the point with the best result;

$MI$  The minimal number of points before proceeding to CLUSTER ANALYSIS ( $MI = 0.2NF$ );

$S$  Vector containing the logical values for repeating of the points.

#### Parameters for CLUSTER ANALYSIS algorithm

$NF$  Initial number of points;

$Num\ CL$  Current number of clusters;

$MNCP$  Current total number of clusters points;

$NPC$  Minimal number of points in each cluster required for starting the Complex method ( $NPC = n + 2$ );

$ITN$  Number of iterations for Complex method ( $ITN = 15$ );

$SC$  Coefficient for reduction of the number of working points ( $RC = 0.4$ ).

#### PARSEC algorithm

The new Parallel Adaptive Random Search with Estimation of Convergence (PARSEC) algorithm is heuristic and iterative. It consists of two blocks (*Block 1* and *Block 2*) given in Fig. 1. *Block 1* is a modified complex method for optimization [11] applied several times. All found maximums of each run of the complex method are stored and sorted. The number of points in the next complex used on the next iteration is estimated in *Block 2*. Two conditions are used for this procedure. The first condition is related to the so-called average radius  $RN$ .

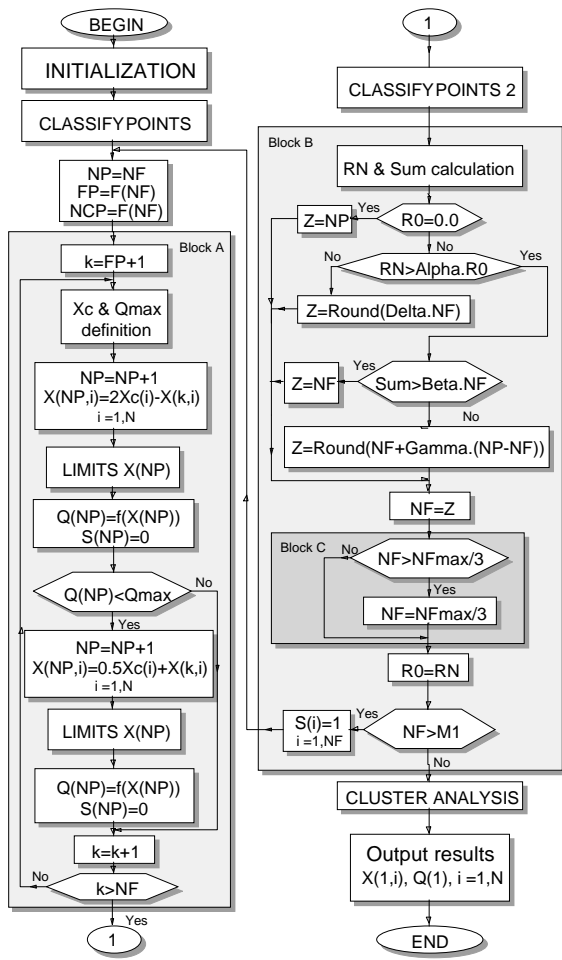


Fig. 1. PARSEC Algorithm.

$RN$  is calculated as an average distance from each point to the point with the best result of the objective function. If  $RN$  is decreasing, it means that the points of the complex are grouped around the supposed maximum and it is necessary to reduce the number of points to avoid unnecessary calculations, especially when the task is with a large number of control variables. If  $RN$  is increased it means that new points with better results are found on the last iteration but they are far off from the current maximum and it is better to increase the number of points.

A second condition is also included for increasing the number of points  $NF$  in the next iteration. If the number of points used in the previous iteration ranked in the group of the “best points” (present maximum) is less than  $\beta * NF$  (for example  $0.95 * NF$ ), then the number of points  $NF$  in the next iteration is increased. Otherwise the assumption is that the points are grouped

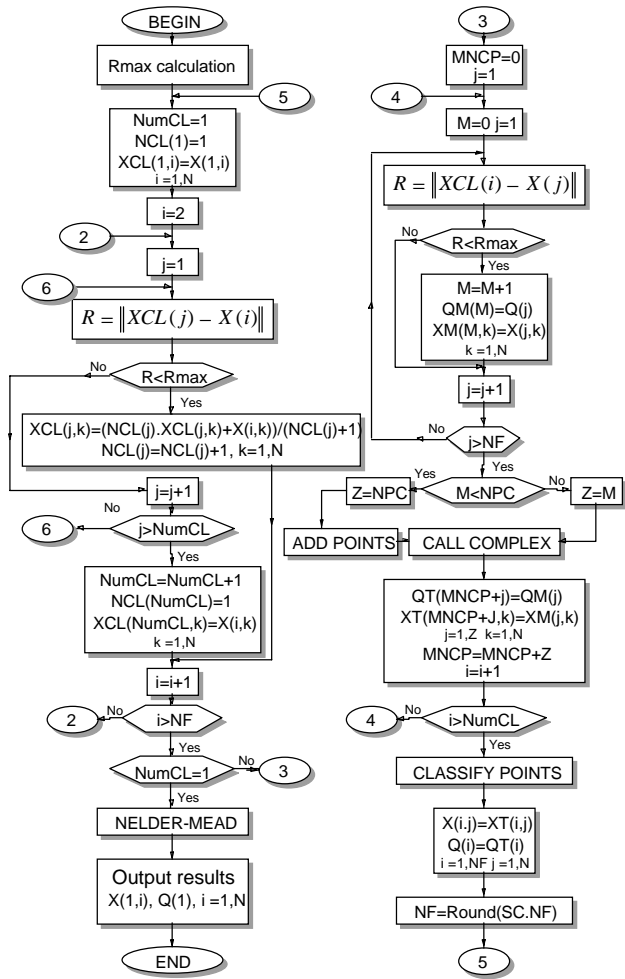


Fig. 2. CLUSTER ANALYSIS algorithm.

in the region of maximums of the function and the number of points is kept. When the current number of points is reduced below  $M1$  it means that the points are well grouped around the local maximums then the algorithm is turned to the clustering analysis procedure. The procedure CLUSTER ANALYSIS is given in Fig. 2.

The modules of the entire algorithm (Fig. 1 and Fig. 2) are as following:

1. **Initialization.**  $NF$  number of points  $x(i,j)$ ,  $i = 1, NF$ ,  $j = 1, N$  are generated randomly with uniform distribution in the feasible region of control variables  $\mathbf{x} \in X^n$ . The objective function  $Q(i)$ ,  $i = 1, NF$  is evaluated at each point. Set  $S(i)=1$ ,  $i = 1, NF$  and  $RO = 0.0$ . Calculate  $M1$  ( $M1 = 0.2NF$ ).  $M1$  will be constant during the whole algorithm.

2. **Classify points.** The whole number  $NF$  of points  $x(i,j)$ ,  $i = 1, NCP$ ,  $j = 1, N$  are sorted in decreasing order according the results of the objective function  $Q(i)$ .

3. **Classify points 2.** The same as **Classify points** but for the current number of points  $NP$ .

4. **Xc and Qmax definition.** Selecting the point  $\mathbf{x}(k)$  with the best value for the objective function. Selection of  $NCP$  number of points  $x(i,j)$ ,  $i=1,NCP$ ,  $j=1,N$  among the best points  $FP$  ( $2 \leq NCP \leq NPF-1$ ), which are the nearest to  $\mathbf{x}(k)$ . Selecting  $Qmax = \max\{Q(i)\}$ ,  $i = 1,NCP$  where  $Q(i)$  are the values of the objective function at the nearest  $NCP$  points to  $\mathbf{x}(k)$ . Calculate the weight center  $\mathbf{x}_c$

$$x_c(j) = \frac{1}{NCP} \sum_{m=1}^{NCP} x(m, j) \quad j = 1, n. \quad (4)$$

5. **Limits X(i).** This procedure checks the boundary conditions for  $\mathbf{x}(i)$ . If  $\mathbf{x}(i) < LowLimit$ , then  $\mathbf{x}(i) = LowLimit$ . If  $\mathbf{x}(i) > HighLimit$ , then  $\mathbf{x}(i) = HighLimit$ ,  $i = 1, NF$ .

6. **RN and Sum calculation.**  $\mathbf{x}(1)$  is the "best point".

$$RN = \frac{1}{NF-1} \sum_{k=2}^{NF} \|x(1) - x(k)\|, \quad (5)$$

$$Sum = \sum_{k=1}^{NF} S(i). \quad (6)$$

The *Block C* is included only in case of necessity to reduce the number of current points when the computer memory is insufficient. The above described procedure is concerned to PARSEC 2. The version PARSEC 1 differs from PARSEC 2 in the following items: The *Block B* is replaced with 2 terminated criteria, the first condition is eq. (1) and the second one is "if  $Sum = NF$  then start CLUSTER ANALYSIS procedure", where  $Sum$  is calculated by eq. (6). In this version the number of points is constant.

#### CLUSTER ANALYSIS procedure

The algorithm is iterative, and is based on the clustering analysis. The centers of the clusters  $XCL$  with radius  $Rmax$  are evaluated in each iteration. The complex method is started for each cluster with points belonging to this cluster with the number of  $ITN$  iterations. If necessary, some additional points are generated. All new points are sorted according to their coordinates in a cluster and values. When only one cluster remains, the optimization procedure of Nelder–Mead

[18] is started for precise localization of the maximum. The basic modules of the algorithm are as follows:

#### 1. Rmax calculation.

$$R_{max} = 0.2 \sqrt{\sum_{j=1}^n (HighLimit(j) - LowLimit(j))^2}. \quad (7)$$

2. **Nelder–Mead.** Generate randomly  $n + 1$  points in the region with a radius of 500 and center the best current point  $\mathbf{x}(1)$ . Start the Nelder–Mead [18] procedure for localization of the maximum with accuracy.

3. **Add points.** New ( $NPC-M$ ) points are added, generated by a uniform random distribution in the region with radius  $Rmax$  and center  $XCL(1)$ .

4. **Call COMPLEX.** Starting the procedure of the complex – method [11] for  $ITN$  iterations from the initial points  $XM(j)$ , (Fig. 2). The current results are kept in the same array.

5. **Classify points.** Sorting  $MNCP$  number of points  $XT(j)$  in decreasing order of the value of the objective function  $QT(j)$ .

The described procedure can be used also as an independent algorithm for global optimization. For this purpose the procedures **Initialization** and **Classify points** (Fig. 1) have to be implemented in the beginning.

## RESULTS AND DISCUSSION

A new performance index is introduced called Probability to Find Global Optimum ( $PFGO$ ). This index gives reliable statistical information about ability of the investigated procedure to find the global optimum, and is defined as the ratio of total number of successful runs  $NSR$ , and total number of runs (equal to 10000). Calculate  $PFGO$  for  $l$ -th test function ( $l=1, M$ ) and average number of function evaluations  $NFE$ :

$$PFGO_l = \frac{NSR}{10000}, \quad (8)$$

$$NFE_l = \frac{TNFE}{NSR}. \quad (9)$$

Calculate the average value for  $PFGO$  and average value for  $NFE$  for all test functions:

$$PFGO = \frac{1}{M} \sum_{l=1}^M PFGO_l, \quad (10)$$

Table 1. Test functions for global optimization.

No	Function description			Limits		Maximum	
		N	i	$x_{imin}$	$x_{imax}$	$Q_{max}$	$\mathbf{x}^*$
1.	$F = -\sum_{i=1}^N (x_i^2 - \cos(18x_i^2))$	2	1	-0.625	0.125	2.0000	$x_1^* = 0.0000$
			2	-0.250	0.500		$x_2^* = 0.0000$
2.	$F = \sum_{i=1}^N x_i^2 + 40 \prod_{i=1}^N \sin(x_i)$	2	1	0.000	7.000	86.8057	$x_1^* = 4.9728$
			2	-3.000	6.000		$x_2^* = 4.9728$
3.	$F = \prod_{i=1}^N x_i \sin(x_i)$	2	1	6.000	16.000	200.8562	$x_1^* = 14.2074$
			2	6.000	16.000		$x_2^* = 14.2074$
4.	$F = abs((-30x_2^4 + 64x_2^3 - 43.8x_2^2 + 10.8x_2 + 0.12) \cdot 1000 \cdot \frac{\sin(5\pi \cdot x_1)}{1 + 0.1x_1^2})$	2	1	0.000	0.500	991.0098	$x_1^* = 0.1000$
			2	0.000	0.500		$x_2^* = 0.2000$
5.	$F = abs((-30x_2^4 + 64x_2^3 - 43.8x_2^2 + 10.8x_2 + 0.12) \cdot 1000 \cdot \frac{\sin(5\pi \cdot x_1)}{1 + 2.724\sqrt{x_1 + 0.001}})$	2	1	0.000	1.000	723.7987	$x_1^* = 0.0900$
			2	0.000	1.000		$x_2^* = 0.9000$
6.	$F = abs((-3x_2^4 + 64x_2^3 - 438x_2^2 + 1080x_2 + 120) \cdot \frac{\sin(5\pi \cdot x_1)}{1 + 0.1x_1^2})$	2	1	0.000	1.000	822.1778	$x_1^* = 0.1000$
			2	0.000	1.000		$x_2^* = 1.0000$
7.	$F = \prod_{i=1}^N \sum_{j=1}^5 j \cdot \cos[(j+1)x_i + j]$	2	1	0.000	10.000	210.4820	$x_1^* = 5.4827$
			2	0.000	10.000		$x_2^* = 5.4827$
8.	$F = -\prod_{i=1}^N \sum_{j=1}^5 j \cdot \cos[(j+1)x_i + j]$	2	1	0.000	10.000	186.7313*	$x_1^* = 5.4827$
			2	0.000	10.000		$x_2^* = 4.8581$
9.	$F = -\sum_{i=1}^N (x_i^2 - \cos(18x_i^2))$	3	1	-0.625	0.125	3.0000	$x_1^* = 0.0000$
			2	-0.250	0.500		$x_2^* = 0.0000$
			3	-0.300	0.450		$x_3^* = 0.0000$
10.	$F = \prod_{i=1}^N \sum_{j=1}^5 j \cdot \cos[(j+1)x_i + j]$	3	1	0.000	10.000	3053.6724	$x_1^* = 5.4827$
			2	0.000	10.000		$x_2^* = 5.4827$
			3	0.000	10.000		$x_3^* = 5.4827$
11.	$F = \prod_{i=1}^N x_i \sin(x_i)$	3	1	6.000	16.000	2846.6093	$x_1^* = 14.2076$
			2	6.000	16.000		$x_2^* = 14.2076$
			3	6.000	16.000		$x_3^* = 14.2076$
12.	$F = \sum_{i=1}^N x_i^2 + 40\sin(x_1)\sin(x_2) + 40\sin(x_1)\sin(x_3) + 40\sin(x_2)\sin(x_3)$	3	1	0.000	7.000	188.3367	$x_1^* = 4.8344$
			2	-3.000	6.000		$x_2^* = 4.8344$
			3	-4.000	5.000		$x_3^* = 4.8344$
13.	$F = -\sum_{i=1}^N (x_i^2 - \cos(18x_i^2))$	5	$x_i$	-1.000	1.000	5.0000	$x_i^* = 0.0000$
				$i=1,\dots,5$			$i = 1,\dots,5$
14.	$F = 0.001 \cdot \prod_{i=1}^N \sum_{j=1}^5 j \cdot \cos[(j+1)x_i + j]$	5	$x_i$	0.000	10.000	642.7431	$x_i^* = 5.4827$
				$i=1,\dots,5$			$i = 1,\dots,5$
15.	$F = 0.001 \cdot \prod_{i=1}^N x_i \sin(x_i)$	5	$x_i$	6.000	16.000	571.7590	$x_i^* = 14.2076$
				$i=1,\dots,5$			$i = 1,\dots,5$

Note: \* - Two equivalent global maximums

$$NFE = \frac{1}{M} \sum_{l=1}^M NFE_l, \quad (11)$$

where  $M$  is the number of test functions ( $M = 15$ , Table 1).

The probability of finding the global optimum ( $PFGO$ ) and the average number of function evaluations ( $ANFE$ ) are investigated only for the solved problems with 15 test functions (Table 1) using the above new algorithms for global optimization and known algorithms. The results obtained are given in Fig. 3 for  $n = 2$ , Fig. 4 for  $n = 3$  and Fig. 5 for  $n = 5$ . It is noticeable that the newly proposed algorithms have a 2 to 7 times less average number of function evaluations ( $NFE$ ) than the other algorithms for equal levels for the probability of finding the global optimum ( $PFGO$ ).

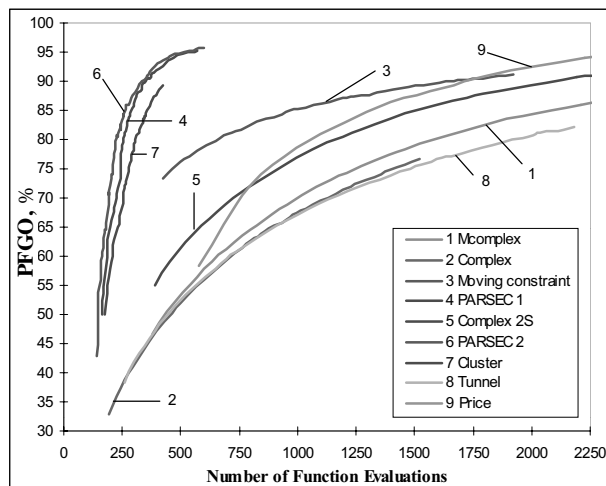


Fig. 3. The  $PFGO$  for test functions with  $n = 2$ .

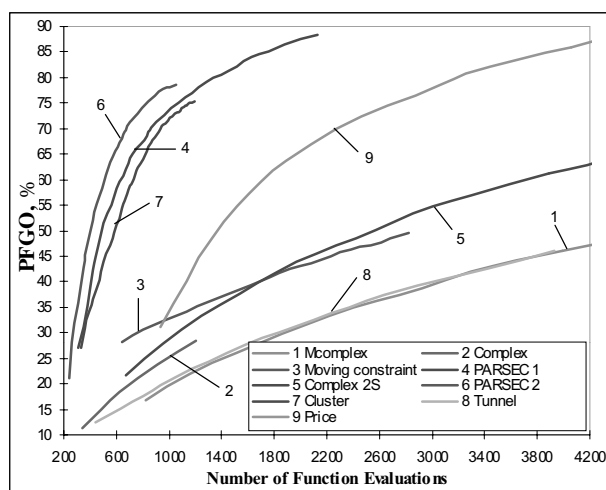


Fig. 4. The  $PFGO$  for test functions with  $n = 3$ .

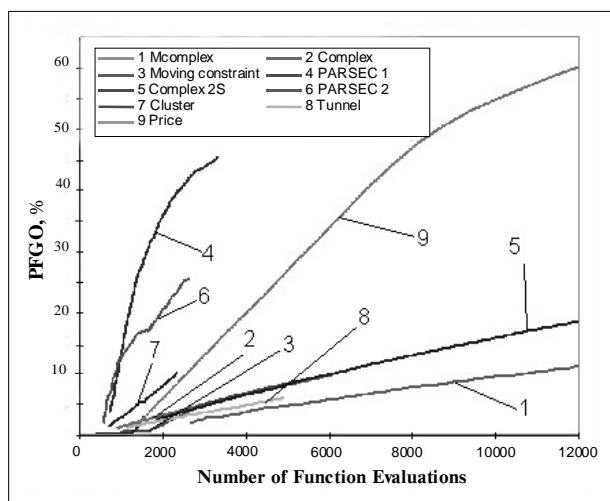


Fig. 5. The  $PFGO$  for test functions with  $n = 5$ .

The optimum number of starting points for some of the algorithms is also investigated. The following function of usefulness ( $U$ ) is used in order to estimate the efficiency of the algorithms for different numbers of starting points  $N_s$ :

$$U = 0.6 \frac{(PFGO_{N_s} - PFGO_{min}^{N_s})}{(PFGO_{max}^{N_s} - PFGO_{min}^{N_s})} + 0.4 \frac{(NFE_{max}^{N_s} - NFE_{N_s})}{(NFE_{max}^{N_s} - NFE_{min}^{N_s})} \quad (12)$$

where  $PFGO_{max}$  and  $PFGO_{min}$  are the maximal and the minimal value for  $PFGO$  calculated by eq. (10), and  $NFE_{max}$  and  $NFE_{min}$  are the maximal and the minimal value for calculated by eq. (11). The accepted weight coefficients for  $PFGO$  and  $NFE$  are 0.6 and 0.4 respectively.

The maximum value  $U_{max}$  for  $U$  is achieved with optimum number of starting points  $N_s$ .

The graphical presentation of the function of usefulness  $U$  is given in Fig. 6. The maximum of  $U$  corresponds to the optimum number of starting points. The recommended value of number of starting points and expected values of  $PFGO$  and  $ANFE$  for the algorithms are given in Table 2. The curves for the PARSEC algorithm shown in Fig. 6 are not smooth (curves 4, 6 and 7). This is due to the manner of calculating of  $NCP$  and  $FP$  as functions of  $NF$ .  $NF$  is estimated as an integer parameter using rounded functions (see Fig. 1). The formulas for estimation of  $NCP$  and  $FP$  are empirically defined on the basis of the results for two and three dimensional objective functions. The optimum values

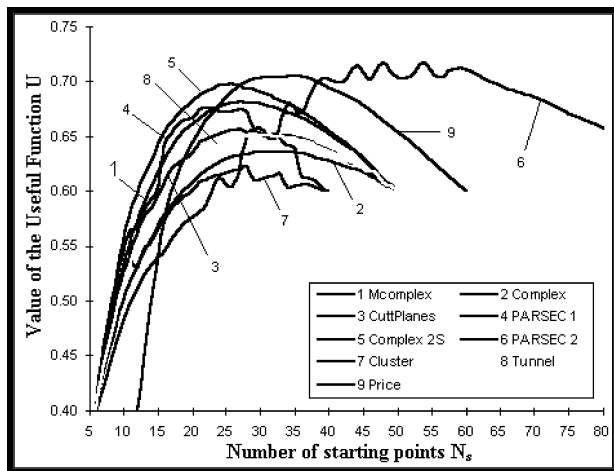


Fig. 6. Evaluation of the optimal number of starting points  $N_s$  for  $n = 2$ .

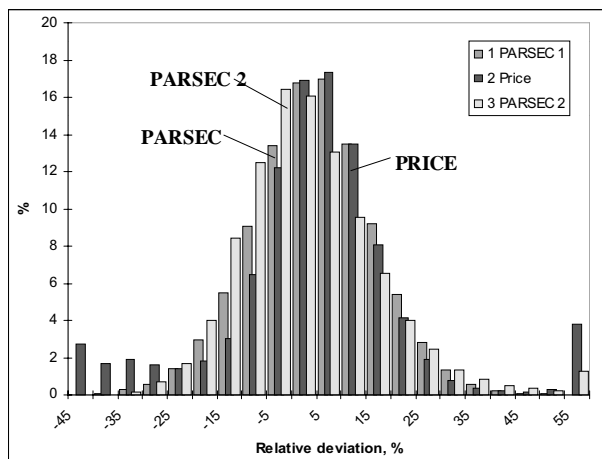


Fig. 7. Distribution of the deviations of ANFE.

for *FP* and *NCP* strongly depend of the surface of the objective function and finding an adaptive algorithm for their estimation will significantly improve the speed of convergence and the efficiency of the new proposed heuristic algorithms.

The distribution of the deviations of ANFE from the total average in percent is given in Fig. 7, which is showing the statistical significance of the obtained results. For PARSEC 1 and PARSEC 2 more than 70 % of the deviations are grouped in the interval -25 % +5 %. It means that the real tasks can be solved with expected less ANFE. For the Price algorithm some results are dispersed on a small scale less than - 35 % and above 45 %.

Table 2. Probability of finding the global optimum (*PFGO*), average number of function evaluations (*ANFE*) and recommended number of starting points (*RNSP*) for the investigated algorithms.

Algorithm	$n$	RNSP	PFGO	ANFE
Multicomplex	2	28	82.2	1769
Complex	2	35	70.8	1162
Complex 2S	2	26	87.2	1724
Moving constraints	2	27	86.9	1172
Cluster analysis	2	28	80.0	308
Price	2	36	91.0	1810
	3	36	80.8	3252
Tunneling	2	29	72.0	1262
PARSEC 1	2	27	87.3	310
	3	55	65.0	725
PARSEC 2	2	60	86.0	267
	3	100	63.4	552

## CONCLUSIONS

From the obtained results is possible to make the following conclusions:

- The proposed new algorithms PARSEC 1, PARSEC 2 and CLUSTER ANALYSIS are more reliable than the rest of investigated algorithms and they have a lower number of function evaluations. Good results are also shown for the Price algorithm and for moving constraints. Non satisfying results are obtained for Complex, Multicomplex and for Tunneling algorithms.

- The idea to use a changeable number of the current group of points for PARSEC 2 (*Block B*, Fig. 1), improves the convergence and can be suggested also for other algorithms.

- The present disadvantages of the algorithms PARSEC 1 and PARSEC 2 are the heuristic parameters and some constants in the algorithm, which have to be chosen according to the most reliable global optimum.

## REFERENCES

1. E. Kenneth, *Advances in Genetic Programming*, Cambridge, Mass. :MIT Press, 1994.
2. L. Davis, *Handbook of Genetic Algorithms*, New York, Van Nostrand, Rainhold, 1991.
3. I. P. Anroulakis, V. Venkatasubramanian, *A Genetic Algorithmic Framework for Process Design and*

- Optimization, *Comput. Chem. Engng.*, **15**, 5, 1991, p. 217-228.
4. I. Mishinev, S. Stoyanov, F. Keil, Investigation and Modification of the Luus - Jaakola Global Optimization Algorithm, *Hungarian Journal of Industrial Chemistry*, **28**, 3, 2000, p. 231-239.
  5. F. Keil, S. Stoyanov, S. Yordanov, Improved Heuristic Rules for the Price Algorithm of Global Optimization, *J. of App. Comp. Science*, **1**, 2007, (in print).
  6. V. K. Chichinadze, Random Search to Determine the Extremum of the Functions of Several Variables, *Eng. Cybernetics*, **1**, 1967, p. 115-123.
  7. B. S. Everitt, Cluster analysis. New York, Halsted Press, 1993.
  8. J. K. Tellalyan, Optimization Methods and Algorithms with Complex Objective Functions, PhD Thesis, HICHT, Sofia, 1985.
  9. C. A. Floudas, P.M.Pardalos, A Collection of Test Problems for Constrained Global Optimization Algorithms, *Lecture Notes in Computer Science 455*, Springer Verlag Heidelberg, 1990.
  10. B. Shubert, A Sequential Method Seeking the Global Maximum of a Function, *SIAM J. of Num. Anal.*, **9**, 3, 1972, p. 379-388.
  11. S. K. Stoyanov, *Methods and Algorithms for Optimization*, Technika, Sofia, 1990, (in Bulgarian).
  12. M. J. Box, A New Method of Constrained Optimization and a Comparison with other Methods, *Computer J.*, **8**, 1965, p. 42-52.
  13. A. V. Levy, S. Gomez, The Tunneling Algorithm for Global Minimization of Functions *SIAM J. Sci. Stat. Comput.*, 1985.
  14. S. Gomez, A Method for Solving the Global Constrained Minimization Problems, PhD Thesis, I.C. London, 1980.
  15. W. L. Price, A Controlled Random Search Procedure for Global Optimization, Report 75-23, University of Leicester, Eng. Dept., Nov., 1975.
  16. P. Brachetti, M. De Felice Ciccoli, G. Di Pillo, S. Lucidi, A New Version of the Price's Algorithm for Global Optimization, *J. Glob. Optim.*, **10**, 1997, p. 165-184.
  17. Y.-C Jiao, C. Dang, Y. Leung, Y. Hao, A modification of the New Version of the Price's Algorithm for Continuous Global Optimization Problems, *J. Glob. Optim.*, **36**, 2006, p. 609-626.
  18. J. A. Nelder, R. Mead, A Simplex Method for Function Minimization, *Comp. J.*, **7**, 1964, p. 308-313.