Interactive Evolutionary Algorithm for Multiple Objective Convex Integer Problems

Leoneed Kirilov[1] , Vassil Guliashki[2] and Krasimira Genova[3]

***Abstract* – An interactive evolutionary algorithm is proposed to solve multiple objective convex integer problems. The algorithm uses a heuristic for fast search, proposed to generate quickly a local approximate representative subset of the efficient frontier. Utility coefficients are calculated and used to support the Decision Maker (DM) to obtain a good reference point. A comparison with two other evolutionary algorithms for the same class optimization problems is done on an illustrative example.**

***Keywords* – Multiple objective integer optimization, Interactive evolutionary algorithms, Pareto-optimal front.**

I. Introduction

Multiple objective convex integer optimization problem is considered in this paper. It can be stated in the following general form:

$$\text{Maximize} \quad \boldsymbol{f}(\boldsymbol{x}) = [f_1(\boldsymbol{x}), f_2(\boldsymbol{x}), \ldots, f_k(\boldsymbol{x})]^T \qquad (1)$$

$$\text{subject to:} \quad \boldsymbol{g_j}(\boldsymbol{x}) \le 0, \qquad j = 1,2,..., m; \qquad (2)$$

$$\boldsymbol{x_i}^{(L)} \le \boldsymbol{x_i} \le \boldsymbol{x_i}^{(U)}, \quad i = 1,2,\ldots, n; \qquad (3)$$

$$\boldsymbol{x} \in \boldsymbol{Z^n}, \qquad (4)$$

where $\boldsymbol{g_j}(\boldsymbol{x})$, $j = 1,2,\ldots, m$; are convex functions and $f_i(\boldsymbol{x})$, $i = 1,2,\ldots, k$; are concave nonlinear functions.

In the text of the paper is used the term "solution" as a vector of variables and the term "point" as a corresponding vector of objectives.

A solution $\boldsymbol{x} \in \mathbf{Z}^{\boldsymbol{n}}$ is a vector of n integer decision variables: $\boldsymbol{x} = (x_1, x_2, \ldots, x_n)^T$. The value $x_i^{(L)}$ is the known lower bound and the value $x_i^{(U}$ is correspondingly the upper bound of variable x_i. The solutions satisfying the constraints (2)-(4) constitute a feasible decision variable space $\boldsymbol{V} \subset \boldsymbol{Z^n}$. The objective functions (1) constitute a k-dimensional space, called objective space $\boldsymbol{F} \subset \boldsymbol{R}^k$. For each solution $\boldsymbol{x}$ in the decision variable space, there exists a point $\boldsymbol{f} \in \mathbf{R}^k$ in the objective space, denoted by $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{f} = (f_1, f_2, \ldots, f_k)^T$.

The problem (1-4) does not posses a unique optimal solution in the objective space. Instead that a conception of Pareto optimality or non-domination is used (see [2], [4], [13], [16]).

The domination between two solutions is defined as follows (see [2, 4, 13]):

[1]Leoneed Kirilov is with the Institute of Information and Communication Technologies – BAS, "Acad. G. Bonchev" str. Bl. 2, 1113 Sofia, Bulgaria, E-mail: leomk@abv.bg
[2]Vassil Guliashki is with the Institute of Information and Communication Technologies – BAS, "Acad. G. Bonchev" str. Bl. 2, 1113 Sofia, Bulgaria, E-mail: vggul@yahoo.com
[3]Krasimira Genova is with the Institute of Information and Communication Technologies – BAS, "Acad. G. Bonchev" str. Bl. 2, 1113 Sofia, Bulgaria, E-mail: kgenova@iinf.bas.bg

Definition 1. *A solution* $\mathbf{x}^{(1)}$ *is said to dominate the other solution* $\mathbf{x}^{(2)}$, *if both the following conditions are true*:

1. *The solution* $\boldsymbol{x}^{(1)}$ *is no worse(say the operator* $\prec$ *denotes worse and the operator* $\succ$ *denotes better) than* $\boldsymbol{x}^{(2)}$ *in all objectives,* or $f_j(\boldsymbol{x}^{(1)}) \not\prec f_j(\boldsymbol{x}^{(2)})$ for $j = 1,2,..., k$;.
2. *The solution* $\boldsymbol{x}^{(1)}$ *is strictly better than* $\boldsymbol{x}^{(2)}$ *in at least one objective,* or $f_j(\boldsymbol{x}^{(1)}) \succ f_j(\boldsymbol{x}^{(2)})$ for at least one $j \in \{1,2,..., k\}$.

All points which are not dominated by any other point $\boldsymbol{f} \in \boldsymbol{F}$ form the set of non-dominated points and the set of Pareto-optimal solutions in the variable space respectively.

There are two ideal goals in the multi-objective optimization:

1. Find a set of solutions which are diverse enough to represent the entire range of the Pareto-optimal front, and
2. Find a set of Pareto-optimal solutions, which satisfy in the best way the DM's preferences.

The interactive algorithms are the most popular in solving multi-objective optimization problems (see [14, 19]. They consist of two alternate phases: 1. Interaction (dialogue) with the DM and 2. Generating solutions. Usually an appropriate single objective convex integer optimization problem is solved during the second phase. Such problems belong to the class of NP-hard problems (see for example [6, 15]). There does not exist an exact algorithm, which can solve these problems in time depending polynomially on the problem input data length or on the problem size. For this reason many researchers investigate approximate algorithms with polynomial computational complexity, which solve such kind optimization problems. For the past 20 years evolutionary multiple objective optimization (EMOO) methodologies have demonstrated their usefulness in finding a set of near Pareto-optimal solutions [2, 3, 5]. As a sequence many source codes – both commercial and free have been created and the EMOO algorithms obtained wide application.

In principle the evolutionary optimization (EO) algorithms use a population-based approach, in which the iterations are performed on a set of solutions (called population) and more than one solution is generated at each iteration. The main positive features making popular the EO algorithms are the following: (i) They do not require any derivative information; (ii) EO algorithms are relatively simple to implement; (iii) EO algorithms are flexible and robust, i.e. they perform very well on a wide spectrum of problems (see [7]); The use of a population in EO algorithms has a number of advantages (see [2]): 1) it provides an EO procedure with a parallel processing power, 2) it allows EO procedures to find multiple optimal solutions, thereby facilitating the solution of multi-modal and multi-objective optimization problems, and 3) it provides an EO algorithm with the ability to normalize decision variables (as well as objective and constraint functions) within an evolving population using the best minimum and maximum values in the population.

Some important disadvantages of EMOO algorithms are: (i) their convergence to the Pareto-optimal front could be slow and may require large number of iterations; (ii) they face difficulty in solving problems with a large number of objectives, i.e. they could obtain difficult a well representative set of Pareto-Optimal Solutions (see [5]).

The number of objectives as a convergence factor is considered in [17]. Good approach in solving problems with large number of objectives is to use the EMOO methodologies to find a preferred and smaller set of Pareto-optimal solutions, instead of the entire front [5]. In this way the DM can concentrate to explore only the regions of Pareto-optimal front, which are of interest to her/him. An accelerating technique for population based algorithms is proposed in [8]. A technique for quickly moving the population to the Pareto-optimal front is proposed in [9, 12]. Some hybrid EMOO algorithms have been recently proposed to overcome the second mentioned disadvantage (see [5, 11]). They expand the use of classical multi-objective optimization procedures (see [13]) like reference point-, reference direction- and other type methods, proposing new approaches and hybrid techniques.

An interactive population-based (evolutionary) algorithm solving the problem (1)-(4) is proposed in this paper. To evaluate and arrange the solutions in the population is used the PROMETHEE method (see [1]). Utility coefficients like those proposed in [10] are calculated and used to support the DM in the choice of a reference point. The algorithm includes a heuristic procedure for quickly moving the whole population to the reference point, according the DM preferences. This procedure leads to better convergence of the search process to Pareto-optimal front.

II. Utility coefficients and choice of reference point

Let we have a population $\boldsymbol{P}$ of solutions in the variable space and let the size of $\boldsymbol{P}$ be p.

The idea here is the algorithm to support the DM in his/her orientation where to search the desired compromise solution. The application of a scalarization process such as weighted sums or root mean square is excluded in [18] by the assumption that the different dimensions of the objectives $\boldsymbol{f}_i(x)$ in (1) are not commensurable. To avoid this obstacle the algorithm uses utility coefficients $\eta_i(x)$ for each objective $\boldsymbol{f}_i(x)$, i=1,…,k; which are computed as follows:

$$\eta_i(x) = \frac{f_i - f_{i\,\min}}{f_{i\,\max} - f_{i\,\min}} \qquad \textbf{(5)}$$

where $f_{i\ max}$ and $f_{i\ min}$ are correspondingly the maximal known and the minimal value for i-th objective. In case $f_{i\ max} = f_{i\ min}$ the denominator of (5) is set to be 1.

On the base of 10% of best members of $\boldsymbol{P}$, arranged by means of special Gaussian generalized evaluation according the method PROMETHEE (see [1]), DM gives his preferences as utility coefficients for each objective $\boldsymbol{f}_i(x)$, i =1,…,k;. Let they be $\eta^r = (\eta_1, \eta_2, \ldots, \eta_k)^T$. Let the best and the worst solution among these 10% members of $\boldsymbol{P}$ are x^b and x^w. The direction **z** is calculated as follows:

$$\mathbf{z} = x^b - x^w \qquad \textbf{(6)}$$

Steps along **z** are done starting by x^b and corresponding solutions, candidate to be chosen as reference solutions are generated: $x^{(1)}, x^{(2)}, \ldots x^{(j)}$.

Then the vectors $\eta(x) = (\eta_1(x), \eta_2(x), \ldots, \eta_k(x))^T$ for each $x^{(1)}$, $x^{(2)}$, …, $x^{(j)}$ are calculated. The Euclidean distance between those vectors and the preferred utility vector η^r is calculated:

$$d^{\,j} = \sqrt{\sum_{i=1}^{k} \left(\eta^j_i - \eta^r_i\right)^2} \qquad \textbf{(7)}$$

The solution $x^{(j)}$, where $d^{\,j}$ becomes minimal is chosen as current reference solution $\boldsymbol{x}^r$ and the corresponding point in the objective space becomes current reference point $\boldsymbol{f}^r$.

III. A fast search heuristic procedure.

The proposed algorithm uses a *heuristic procedure* to move quickly the whole population to the Pareto-optimal front. It consists of following steps:

The weight center $\boldsymbol{C}$ of the solutions $\boldsymbol{x}^i \in \boldsymbol{P}$, for i = 1,…,p; is calculated. The components $\boldsymbol{C}_i$ of $\boldsymbol{C}$ are:

$$\boldsymbol{C}_i = \frac{\sum_{j=1}^{p} x_i^j}{p}, \text{ for } i = 1,\ldots,n; \qquad \textbf{(8)}$$

A direction in the variable space for moving the whole population to the reference solution is calculated:

$$\boldsymbol{y} = \boldsymbol{x}^r - \boldsymbol{C} \qquad \textbf{(9)}$$

It is expected that the $\boldsymbol{y}$ vector is directed to the Pareto-optimal front, because the solutions x^b and x^w are evaluated by means of the Gaussian generalized criterion [1]. Then we move the whole population with step $\boldsymbol{y}$ to the reference solution $\boldsymbol{x}^r$. In case $\boldsymbol{x}^r$ is infeasible some members of the new population may become infeasible. In case x' violates some constraint in the system (2)-(3) the corresponding feasible solution is calculated by using Golden section method for line search along the segment xx' (where x is the corresponding starting solution) and by rounding the final x' to an integer solution.

IV. The proposed interactive evolutionary algorithm

The considered problem has a closed feasible domain because there are given lower and upper bounds for each variable (see constraints **(3)**). The Tchebycheff center of the feasible domain can be calculated and it can be rounded off to the closest integer feasible solution $\boldsymbol{x}^{ch}$, called below rounded Tchebycheff center. There is also a possibility to use near Tchebycheff center $\boldsymbol{x}^{nch}$ for the domain determined by the constraint system **(3)**, but only if $\boldsymbol{x}^{nch}$ is feasible for the domain determined by **(2)-(3)**.

By means of the fast search heuristic procedure the whole population is translated fast and close to the Pareto-optimal front.

Below is presented the general scheme of the new algorithm:

Scheme of the algorithm

Step 1. Set iteration counter $h = 0$. **A**round the Tchebycheff center $\boldsymbol{x^{ch}}$ of the feasible domain generate a number of p uniform distributed solutions' vectors by using deviation of ±δ, where δ is a constant or % of corresponding component (for example δmax = ±5%). Use them to create the initial population P_h.

Step 2. Evaluate the members of P_h using the Gaussian generalized criterion:

$$F_l(x^{(i)}x^{(j)}) = 1 - e^{(-d^2/2s^2)},\ l = 1,\ldots,k; \qquad \textbf{(10)}$$

where $s = 2$ and $d = \boldsymbol{f}_l(x^{(i)}) - \boldsymbol{f}_l(x^{(j)})$.

$$\pi(x^{(i)}x^{(j)}) = \sum_{l=1}^{k} F_l(x^{(i)}x^{(j)}),\ \text{for } \forall i,j = 1,\ldots,p;\ \text{and } i\neq j; \qquad \textbf{(11)}$$

$$\boldsymbol{\Phi}^{+}(x^{(j)}) = \sum_{i\neq j,\ i=1}^{p} \pi(x^{(j)}x^{(i)}),\ j = 1,\ldots,p; \qquad \textbf{(12)}$$

Arrange the solutions in P_h in descending order according their $\boldsymbol{\Phi}^{+}$-values ($\boldsymbol{\Phi}^{+}(x^{(j)})$ for $j = 1,\ldots,p$;).

Step 3. Among the best 10% of members of P_h determine the best and the worst solution according their $\boldsymbol{\Phi}^{+}$-values. Let they are x^b and x^w. Then calculate direction **z** as shown in **(6).**

Step 4. Calculate the utility vectors $\eta(x) = (\eta_1(x), \eta_2(x), \ldots, \eta_k(x))^T$ for the best 10% of members of P_h.

Step 5. Show the calculated utility vectors to DM and ask he/she to put his/her preferences in form of utility coefficients for each objective $\boldsymbol{f}_i(x)$, $i = 1,\ldots,k$; Use these coefficients as components of the utility vector η^r.

Step 6. Starting by x^b make steps along **z** (calculated at Step 3.) and generate corresponding solutions, candidate to be chosen as reference solutions: $x^{(1)}, x^{(2)}, \ldots x^{(j)}$.

Step 7. Calculate the utility vectors $\eta(x) = (\eta_1(x), \eta_2(x), \ldots, \eta_k(x))^T$ for each $x^{(1)}, x^{(2)}, \ldots, x^{(j)}$.

Step 8. Calculate the Euclidean distance between those vectors and the preferred utility vector η^r. Choose the solution having minimal Euclidean distance to η^r as current reference solution $\boldsymbol{x}^r$ and the corresponding point in the objective space as current reference point $\boldsymbol{f}^r$.

Step 9. Perform the *heuristic procedure* to move P_h towards the Pareto-optimal front:

Calculate the weight center **C** of the solutions $\boldsymbol{x}^i \in P_h$, for $i = 1,\ldots,p$; (see **(8)**). Then calculate the vector $\boldsymbol{y}$ as shown in **(9)**.

Move the population with step size α along this direction: $\{Pnew\} = \{Ph\} + \alpha.\boldsymbol{y}$. The goal is the population *Pnew* to be located as close as possible to the Pareto-optimal front.

Step 10. Some solutions in the *Pnew* may be infeasible. For each infeasible solution in *Pnew* perform the Golden section method for line search to move it to the feasible region.

Step 11. The DM evaluates all points in *Pnew* and if he/she is satisfied by one of them go to **Step 12**, otherwise set $h = h+1$, $Ph = Pnew$, and go to **Step 2**.

Step 12. STOP. (End of the algorithm)

V. ILLUSTRATIVE EXAMPLE

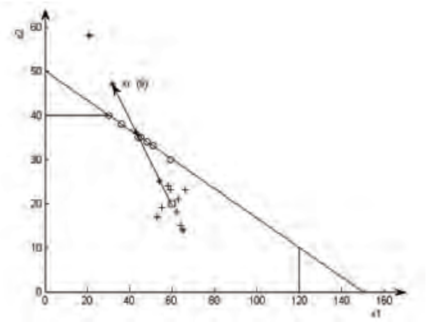
The performance of presented algorithm is illustrated on the following test example:

$$\textbf{Max } \boldsymbol{f}(x) = [\boldsymbol{f}_1(x), \boldsymbol{f}_2(x)]^T,$$
$$\boldsymbol{f}_1(x) = x_1;\ \boldsymbol{f}_2(x) = x_2;$$

subject to:

$$x_1 + 3x_2 - 150 \le 0$$
$$0 \le x_1 \le 120,$$
$$0 \le x_2 \le 40,$$
$$x_1, x_2 \text{ – integer;}$$

The performance for this example for one iteration of the algorithm is presented on Fig. 1.



Legend:

+ – initial population P_0

□ – weight center of P_0

* – solutions $x^{(j)}$ at **Step 6**

o – population *Pnew*

Fig.1. Result of algorithm after the first iteration

At Step 1. Here is used the near Tchebycheff center $\boldsymbol{x^{nch}}$ = (60, 20) because it is feasible solution. The following initial population P_0 is created: $x^{(1)}$ = (66,23), $x^{(2)}$ = (53,17), $x^{(3)}$ = (58,24), $x^{(4)}$ = (55,19), $x^{(5)}$ = (62,18), $x^{(6)}$ = (65,14), $x^{(7)}$ = (63,21), $x^{(8)}$ = (54,25), $x^{(9)}$ = (59,23), $x^{(10)}$ = (64,15);

At Step 2. The corresponding $\boldsymbol{\Phi}^{+}$-values are: 12,76215; 13,8756; 13,39125; 13,44787; 12,93763; 13,90183; 12,37131; 14,69332; 12,2522; 12,99491;

At Step 3. The best 10% of population are the solutions $x^{(8)}$ = (54,25) and $x^{(6)}$ = (65,14). The direction **z** = (-11, 11).

At Step 4. The minimal and maximal values for f_1 are (0, 120) and for f_2 are (0, 40). The utility vectors for $x^{(8)}$ and $x^{(6)}$ are: $\eta(x^{(8)})$ = (0.45; 0.625); $\eta(x^{(6)})$ = (0.54; 0.35);

At Step 5. The DM specifies as most preferable the utility vector η^r = (0.58; 0.75).

At Step 6. Starting by $x^{(8)}$ = (54,25) the following solutions are generated along z: (43, 36), (32, 47), (21, 58), (10, 69) and (0, 79).

At Step 8. The obtained reference solution $\boldsymbol{x}^r$ = (32, 47). It coincides with the corresponding reference point $\boldsymbol{f}^r$.

At Step 9. The weight center of P_0 is **C** = (60, 19.9). The vector $\boldsymbol{y}$ = (-28, 27.1).

At Step 10. The generated new population is *Pnew* = {(59, 30), (30, 40), (48, 34), (36, 38), (45, 35), (44, 35), (51, 33), (44, 35), (48, 34), (44,35)}.

At Step 11. The best obtained solutions are $x^{(2)}$ = (30,40) and $x^{(1)}$ = (59, 30). The corresponding $\boldsymbol{\Phi}^{+}$-values are: 17,22825 and 28,27333. The solution arranged on third position is: $x^{(4)}$ = (36, 38) with $\boldsymbol{\Phi}^{+}$-value 15,56581. The utility vectors are: $\eta(x^{(2)})$ = (0.25; 1); $\eta(x^{(1)})$ = (0.492; 0.75); $\eta(x^{(4)})$ = (0.3; 0.95); The DM chooses as compromise the solution $x^{(1)}$ = (59, 30) and terminates the calculations.

If DM wishes, she/he can perform a new iteration. In this case the algorithm continues by Step 2 and DM can specify new preferences in the form of utility coefficients. In this way DM could explore different parts of Pareto-optimal front.

The solution of the above example by the method SPEA (see [4], [20]) is as follows: Initial population P_0 is the same. External population is P_1^{ext} = {(65, 27), (67, 20)}. After the first iteration the current population is P_1= {(53, 17), (58, 24), (55, 19), (65, 27), (67, 20), (63, 21), (54, 25), (59, 23), (66, 23), (58, 23)}. The best obtained solutions are: $x^{(4)}$ = (65,27), $x^{(5)}$ = (67,20) and $x^{(8)}$ = (66, 23). The corresponding utility vectors are: $\eta(x^{(4)})$ = (0.542; 0.675); $\eta(x^{(5)})$ = (0.55; 0.575); $\eta(x^{(8)})$ = (0.558; 0.5);

It can be seen that SPEA generates dispersed sample along the whole Pareto-optimal front.

The same example is solved by means of algorithm MGA, proposed in [10], starting with the same initial population P_0. The following result is obtained: After the first iteration the current population is P_1= {(66, 25), (65, 25), (66, 24), (65, 24), (66, 23), (65, 23), (66, 21), (65, 21), (66, 19), (65, 19)}. The best obtained solutions are: $x^{(1)}$ = (66,25) and $x^{(2)}$ = (65, 25). The corresponding utility vectors are: $\eta(x^{(1)})$ = (0.55; 0.625); $\eta(x^{(2)})$ = (0.542; 0.625);

Obviously the algorithm proposed here has better convergence to the Pareto-optimal front than the algorithms SPEA and MGA.

VI. Conclusion

The proposed algorithm is suitable to solve real-life large size multiple objective integer optimization problems. It has the following basic characteristics:

- It is designed to find a preferred set of solutions instead of the entire Pareto-optimal set.
- It can quickly converge to the desired part of Pareto-optimal front.
- It is indifferent to the shape of Pareto-optimal front.
- It is applicable to problems with large number of objectives and large number of variables.
- It does not put great demands to the DM.
- It is an interactive evolutionary method and could generate a number of solutions in the region of interest, so that the DM would be able to find without great efforts the satisfactory non-dominated solution among them.

The algorithm is realized as module in a web-based interactive system for multiple objective optimization.

Acknowledgement

The authors gratefully acknowledge the support of Bulgarian National Science Fund, Grant No DTK02/71 “Web-Based Interactive System, Supporting the Building Models and Solving Optimization and Decision Making Problems” and IICT-BAS research project “Modeling, Optimization and Multiple Criteria Decision Making”.

References

[1] Bana e Costa, C. A. ed., *Reading In Multiple Criteria Decision Aid*, Springer Verlag, Berlin-Heidelberg, 1990.

[2] Branke J., Deb K., Miettinen K., Słowiński R., „*Multiobjective Optimization. Interactive and Evolutionary Approaches*“, Springer-Verlag Berlin Heidelberg, 2008

[3] Coello C. A. C., D. A. Van Veldhuizen, and G. Lamont *Evolutionary Algorithms for Solving Multi-Objective Problems*, Boston, MA: Kluwer Academic Publishers, 2002.

[4] Deb K., „*Multi-Objective Optimization Using Evolutionary Algorithms*”, Wiley-Interscience Series in Systems and Optimization. John Wiley & Sons, Chichester, 2001

[5] Deb K., J. Sundar, U. B. Rao N., S. Chaudhuri, “Reference Point Based Multi-Objective Optimization Using Evolutionary Algorithms”, *International Journal of Computational Intelligence Research*, Vol. 2, No. 3 (2006), pp. 273-286.

[6] Garey M. R. and Johnson D. S. *Computers Intractability: A Guide to the Theory of NP-Completeness*, W. H. Freeman, San Francisco 1979.

[7] Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading Mass, Addison Wesley, 1989.

[8] Guliashki V., „An Accelerating Technique for Population Based Algorithms”, Proceedings of Papers of IX Internatrional Conference CompSysTech’08, Gabrovo, Bulgaria, 12-13 June, 2008, pp. -IIIB.11-1- ÷ -IIIB.11-6-.

[9] Guliashki V., L. Kirilov, “SPEA-Based Method for MCDM Convex Integer Problems”, *Cybernetics and Information Technologies*, Vol. 9, No. 4 (2009), pp. 93-101.

[10] Guliashki V. “A Modified Genetic Algorithm for Multiple Objective Convex Integer Optimization”, Working paper No 43, Institute of Information Technologies, Section “Decision Support Systems”, 1997.

[11] Ivanova J, “An Interactive Hybrid Algorithm of Integer Multiobjective Optimization”, International Conf. Automatics and Informatics’08, Proc. “School for young Scientists Business Process management”, (Ed. P. Ruskov), 2008, pp.45-48.

[12] Kirilov L., V. Guliashki, “Interactive Evolutionary Method FIEM for Solving Integer Multiple Objective Problems”, *Comptes rendus de l’Academie bulgare des Sciences*, Tome 64, No2, (2011), pp. 203-212. (in the press)**.**

[13] Miettinen K., P. Neittaanmäki, M. M. Mäkelä, J. Périaux, “*Evolutionary Algorithms in Engineering and Computer Science: Recent Advances in Genetic Algorithms, Evolution Strategies, Evolutionary Programming, Genetic Programming and Industrial Applications*”, John Wiley & Sons, Chichester, 1999.

[14] Miettinen K., *Nonlinear multiobjective optimization*, Kluwer, Norwell. USA, 1999.

[15] Nemhauser G.L. and Wolsey L.A. *Integer and Combinatorial Optimization*, Wiley, New York 1988.

[16] Pareto V. “*Cours d’Economie Politique*”, Rouge, Lausanne, Schwitzerland, 1896.

[17] Praditwong, K.; Xin Yao, „How well do multi-objective evolutionary algorithms scale to large problems”, *Evolutionary Computation*, 2007. CEC 2007. *IEEE Congress on*, 25-28 Sept. (2007), pp. 3959 – 3966.

[18] Schaffer J. D. “Multiple Objective Optimization with Vector Evaluated Genetic Algorithms”, Proceedings of I. International Conference on Genetic Algorithms and their Applications, (1985), Lawrence Erlbaum Associates, Publishers, pp. 160-168.

[19] Steuer R., *Multiple Criteria Optimization: Theory, Computation and Application*, John Wiley & Sons, New York, 1986.

[20] Zitzler E., L. Thiele. In: *IEEE Transactions on Evolutionary Computation*. **3**, 1999, No 4, 257-271.