An Exact Interactive Method for Dolving Multiple Objective Integer Problems

Vassil Guliashki[1] and Leoneed Kirilov[2]

Abstract – **An interactive population-based method is presented in the paper. It is designed to solve multiple objective convex integer optimization problems. A heuristic procedure is used to speed up the search process. The method finds exact non-dominated solutions. The properties of this method are proven theoretically.**

Keywords – **Exact methods, interactive population-based approach, multiple-objective optimization.**

I. Introduction

The evolutionary multi-objective optimization (EMOO) is a popular field for research and development of methods, which perform well on a wide spectrum of problems [3, 7, 8]. The Evolutionary Optimization (EO) methods apply an approach, in which the iterations are performed on a set of solutions (called population). The shortcoming, when a population of solutions is used, is the increase of computational cost and of the memory needed for the execution of one iteration. To overcome this shortcoming the research efforts are focused in the following two directions:

– speeding up the moving of the whole population, keeping the dispersal in the same time with the aim the whole non-dominated set to be investigated.

– speeding up the choice of one compromise solution by the DM moving quickly the population to solutions, which are interesting (desired) and acceptable for him/her.

In the development of the method, presented in this paper, the authors applied the second approach.

The problem considered in this paper belongs to the class of NP-hard optimization problems [6]. There does not exist an exact algorithm, which is able to solve these problems in time, depending polynomially on the problem input data length or on the problem size.

II. Preliminary considerations

The integer multi-objective convex optimization problem can be stated as follows:

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

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

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

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

[1]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

[2]Leoneed Kirilov is with the Institute of Information and Communication Technologies – BAS, "Acad. G. Bonchev" Str. Bl. 2, 1113 Sofia, Bulgaria, E-mail: lkirilov@iinf.bas.bg

where $g_j(x)$, $j = 1,2,\ldots, m$; are convex functions and $f_i(x)$, $i = 1,2,\ldots, k$; are convex functions; $x_i^{(L)}$ and $x_i^{(U)}$, $i = 1, 2,\ldots, n$ are the known lower and upper bound of the variable x_i respectively.

The solution $x \in Z^n$ denotes a vector of n decision variables: $x = (x_1, x_2,\ldots, x_n)^T$. The constraints (2)-(4) constitute a feasible decision domain $V \subset Z^n$.

$S = f(V) = \{ s = f(x), x \in V\}$ is a k-dimensional objectives' region, $S \subset R^k$.

We shall use the term "solution" as a vector of variables in the decision space and the term "point" as a vector of the criteria values in the objectives' space.

Definition: A solution $x^{(1)}$ is said to dominate the solution $x^{(2)}$, if the following two conditions are true:

1. The solution $x^{(1)}$ is not worse than $x^{(2)}$ in all the objectives. Thus, the solutions are compared based on their objective function values.

2. The solution $x^{(1)}$ is strictly better than $x^{(2)}$ in at least one objective.

All the points which are not dominated by any other point $\boldsymbol{s} \in S$, are called Pareto-optimal points. They constitute together the Pareto-optimal front [1, 3] in the objectives' space.

There are two basic approaches for solving the problem (1)-(4): The first one is to choose one "compromise / final" non-dominated solution among many others. Person called Decision Maker (DM) evaluates the solutions obtained during the search process. A number of methods realizing this approach exist [1, 2, 3, 8]. The second approach is to find the whole set of non-dominated points (efficient frontier). This problem is solved completely only for linear case [4, 5].

Evolutionary methods seems to be very suitable to apply the second approach, namely to find an approximation for the whole non-dominated set (see for example [1, 10]).

On the other hand, if the population cardinality is too large this leads to computational difficulties, such like calculation time, dispersion of the population, large memory used etc.

Here an evolutionary method is proposed, which applies the first approach. It performs with limited population, but large enough to approximate locally the efficient frontier driven by the DM's preferences. The process is repeated until a final solution is found. Thus we exploit the advantages of EO approach to generate a good approximation of efficient frontier. Note that when using traditional scalarizing methods for MO problems the question arises – how to support the DM in setting his/her preferences. Some of those methods use trade-off, other use search in a reference direction, or generate a number of additional points [12, 13, 15, 16, 17]. The aim of the method proposed here is to support the DM in the presenting his/her preferences as reference points.

The most popular EO algorithms for mulicriterial problems are NSGA, NSGA-II, SPEA, SPEA2, but they have some disadvantages (see [18]):

1) The non-dominated sorting of NSGA-II algorithm requires population, twice larger in size in comparison to the other evolutionary algorithms, like SPEA and SPEA2.

2) The number of objectives as a convergence factor is considered in [18]. The results showed that the performance of NSGA-II and SPEA2 deteriorates substantially as the number of objectives increases, but SPEA 2 seems to have better performance than NSGA-II in higher dimensional objective spaces [19]. NSGA-II, for example doesn't have good convergence for problems with six or more objectives.

To overcome the above mentioned disadvantages and to increase the efficiency of the proposed method in finding out Pareto-optimal solutions, which are close to the DM's preferences, here are proposed the following improvements:

- We use a heuristic procedure to accelerate the moving the whole population towards the Pareto front. It is similar to those, described in [9]. In this way we improve the speed performance of the method.
- We include an interaction step, where the DM sets a reference point f^r in the objectives' space like in [14]. Our method suggests several reference points to DM, who has the possibility periodically to choose one of them or to input another one in dependence of his/her preferences, to change his/her preferences and to replace the former reference point by a new one. This step ensures the convergence of the proposed method to a desired non-dominated solution.

III. THE NEW EXACT INTERACTIVE METHOD

We use an internal population *P* of *N* solutions and an explicitly defined external population *Pe*. The population *Pe* contains the best *k* non-dominated solutions found during the search (*k* is the number of objectives in (1)).

We propose a *heuristic procedure* to move quickly the internal population to the Pareto-optimal front. For this purpose we calculate the direction $\boldsymbol{y} = Ce - Ci$, where *Ce* is the weight center of *Pe* and *Ci* is the weight center of last 10% of solutions in *P*, relevant to the points with worst objectives values. The points in *P* are ordered in an ascending order according to the number of solutions in P, dominated by each point. The $\boldsymbol{y}$ vector is directed to the Pareto-optimal front, because the members of *Pe* dominate all members of *P*. Then we move the population as close as possible to the Pareto-optimal front (the movement of the population in the solution space is limited by the boundaries of the feasible domain, defined by the system (2)-(3)). We perform consecutive steps calculating solution $x' = x + \alpha.\boldsymbol{y}$, where α is the step length. In both cases: 1) when x' violates any constraint in the system (2)-(3) and 2) when the current step in $\boldsymbol{y}$ – direction leads to deteriorating the sum of criteria values, the corresponding feasible solution is calculated using the Golden section method. In this way a line search along the segment *xx*' is performed and the found solution is rounded off to an integer solution.

There are two possibilities in regard to the location of Pareto-optimal front in the solution space:

1) The Pareto-optimal front is located on the boundary of the feasible domain.

2) The Pareto-optimal front is located inside the feasible domain.

We present below the scheme of heuristic procedure for moving the solutions of population to reach the Pareto-optimal front in both cases:

Scheme of the new moving heuristic procedure MHP

Step 1. Calculate the function $\eta(x) = \sum_{i=1}^{k} \eta_i(x)$,

where $\eta_i(x) = \left(f_{i,\max} - f_i(x)\right) / \left(f_{i,\max} - f_{i,\min}\right)$. Here $f_{i,\max}$ and $f_{i,\min}$ are the maximal and the minimal objective value, and $f_i(x)$ is current value for the *i*-th objective, $i = 1,\ldots, k$.

Step 2. Find the maximal value x* of the function $\eta(x)$ over the rays defined by each population solution belonging to *Pe* and the vector $\boldsymbol{y}$. The Golden section method is used for this calculation in both cases: 1) when violating a constraint of system (2)-(3) occurs or 2) when the sum of criteria values gets worse.

The above heuristic is based on the following prerequisites:

1) The direction **y** is an improving direction by its construction. This means that between every two different solutions x_1 and x_2 lying on a ray $\vec{y}$ with starting solution *Ce* the following relations are satisfied: $f(x_1) \le f(x_2)$ or $f(x_1) \ge f(x_2)$, but the solutions x_1 and x_2 are not incomparable.

2) The function $\eta(x)$ obtains its maximum at a point which is located on the Pareto optimal front.

3) $0 \le \eta_i(x) \le 1$, for $i= 1, \ldots, k$.

4) $1 \le \eta(x) \le k$ where *k* is the number of objectives.

The following results are proven:

Lemma 1: $0 \le \eta_i(x) \le 1$, for all $i= 1, \ldots, k$.

Proof: It follows from the construction of $\eta_i(x)$ for all $i= 1, \ldots, k$. □

Lemma 2: The function values of $\eta(x) = \sum_{i=1}^{k} \eta_i(x)$ are within interval [1, k] for convex multiple objective problems.

Proof:
It follows from the construction of $\eta(x)$ and from the properties of efficient frontier of convex multiple objective problems. □

Theorem 1: The function $\eta(x) = \sum_{i=1}^{k} \eta_i(x)$ is monotonously increasing function over $\Omega_d = d \cap X$, where d is a direction in the space of decision variables such that $f(d) = \{ f_1(d), f_2(d), \ldots, f_k(d) \}$ intersects the efficient frontier in $f(X) \subset R^k$.

Proof: It follows from the convexity of the separate objectives, the convexity of vector function $f(x) = \{ f_1(x), f_2(x), \ldots, f_k(x)\}$. □

Remark: Such directions we will call directions of improving.

Corrolary: The maximum of the function $\eta(x)$, where

$\eta(x) = \sum_{i=1}^{k} \eta_i(x)$, belongs to the efficient frontier over each improving direction.

Theorem 2: The directions used in the heuristic are the directions of improving.

Proof: It follows from the construction of the direction d in the heuristic procedure. Namely if we assume the contradiction that the direction is not improving, i.e. there exists two points $x^{(1)}$ and $x^{(2)}$ from d such that:

1) $x^{(2)} = x^{(1)} + \rho.d^{(1)}$, where ρ is positive number, $d^{(1)}$ is the defining vector along the direction d with $\|d^{(1)}\| = 1$.

2) $\eta(x^{(2)}) \le \eta(x^{(1)})$

But this contradicts to the way of construction of improving directions ("upper - right" for the "max" optimization problems) in the heuristic procedure.

A hybrid method is presented in [11]. The basic differences in comparison to the method presented here are as follows:

1) The population ranking here is not based on a scalarization fitness function. Instead the Euclidean distance to the reference point is used.

2) The improving direction is determined in a different way. Namely, here it is defined as the difference between the weight centers corresponding to the best 10% and to the worst 10% of solutions in the current population.

Scheme of the proposed exact interactive method

Step 1. Set the iteration counter $h = 0$. Generate $N + k$ uniform distributed solutions' vectors around the Chebyshev center *Ch* of the feasible domain by using a deviation of $\pm\delta$, where δ is a % of the corresponding component variation (for example, δmax = ±5%). Use *N* of them to create the initial population P_h and *k* of them to create the external popul. *Pe*.

Step 2. Perform the heuristic procedure MHP to move P_h towards the Pareto-optimal front.

Step 3. Arrange the solutions in P_h according to their Euclidean distance to a candidate reference point $f^{\,r}$ set by DM. Compute the weight centers C^{best} and C^{worst} correspondingly of first 10% and of last 10% of solutions in P_h, relevant to the reordered points. (Another possibility is, DM to make a choice of up to 5 best solutions and up to 5 worst solutions in P_h. Then C^{best} and C^{worst} are calculated correspondingly.) Form the moving direction $d = C^{\text{best}} - C^{\text{worst}}$. Compute a series of solutions $t^{\,l} = C^{\text{best}} + l{*}d$, $l = 1,2,\ldots$; and present the corresponding points $f(t^{\,l})$ to the DM as possible reference points. The DM chooses one of them or sets another one candidate ref. point, according to his/her preferences.

Step 4. Form a direction with an origin – Chebyshev center *Ch* and an end – the reference solution x^r, corresponding to $f^{\,r}$. Then move this reference solution as close as possible to the Pareto-optimal front (reaching the boundaries of the system (2)-(3) if necessary) along this direction by using the Golden section method for line search as in the heuristic procedure MHP. The solution $x^r{*}$ is obtained. Let us denote the corresponding reference point by $f^{\,r*}$.

Step 5. Compute the weight center C^i of population P_h and the vector $d^{\,r}$ with an origin C^{i} and an end – the solution, corresponding to $f^{\,r*}$. Move the population P_h with a step size 1 along this direction: $\{Pnew\} = \{P_h\} + 1.d^{\,r}$.

Step 6. Some solutions in the *Pnew* may be infeasible. For each infeasible solution x^i in *Pnew* perform the procedure in *Step 4* along x^iCh to move it to the feasible domain. For each feasible solution perform also the procedure in *Step 4* to move it as close as possible to the Pareto-optimal front.

Step 7. Arrange all the points corresponding to the solutions in *Pnew* according to their Euclidean distance to the reference point $f^{\,r*}$. The first ten points are shown to the DM and if he/she is satisfied by one of them, go to *Step 8*, otherwise set h=h+1, *Ph*=*Pnew*, and go to *Step 3*.

Step 8. End.

IV. Illustrative Example

We consider the following problem:

$$\text{Min} \quad f_1 = 1/(x_1+1),$$
$$\text{Min} \quad f_2 = 1/(x_2+1),$$

subject to:
$$x_1^2 + 100x_2^2 \le 10^6;$$
$$0 \le x_1 \le 1000;$$
$$0 \le x_2 \le 100;$$
$$x_1, x_2 \in Z.$$

The search process of one iteration is presented on Fig.1.

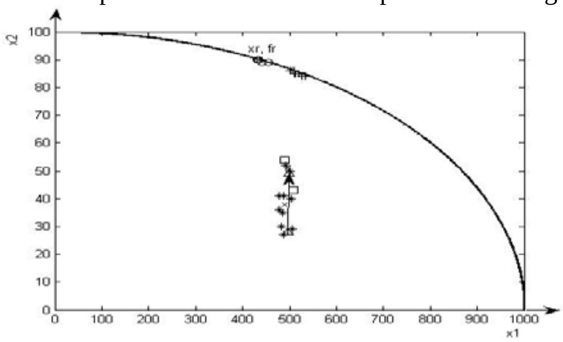

Fig. 1. Performance of one iteration

Legend:

* – initial population P_0; □ – members of Pe_0

Δ – weight centers *Ci* and *Ce* x – solutions $t^{\,j}$ at *Step 3*

\+ – P_h at the end of *Step* 2; ✡ – ref. solution $x^r{*}$

o – members of final population

We denote values $s^{\text{i}}_{\text{r}} = s^{\text{i}}.10^3$. The initial internal population at *Step* 1 is: x^1 = (500; 50), s^1_{r} = (1.996; 19.608), x^2 = (506; 29), s^2_{r} = (1.972; 33.333), x^3 = (482; 30), s^3_{r} = (2.070; 32.258), x^4 = (493; 52), s^4_{r} = (2.024; 18.868), x^5 = (477; 41), s^5_{r} = (2.092; 23.810), x^6 = (485; 35), s^6_{r} = (2.058; 27.778), x^7 = (504; 40), s^7_{r} = (1.980; 24.390), x^8 = (487; 41), s^8_{r} = (2.049; 23.810), x^9 = (488; 27), s^9_{r} = (2.045; 35.714), x^{10} =(479; 36), s^{10}_{r} =(2.083; 27.027);. Its weight center is *Ci* = (497; 28). The initial external population is x^1=(508; 43), s^1_{r}=(1.9646; 22.7272), x^2=(489; 54), s^2_{r}=(2.0408; 18.1818);. Its weight center is *Ce* = (498.5; 48.5). Worst solutions are x^2 and x^9. *Ci* = (497; 28). The direction *y* at *Step* 2 is *y* = [1.5; 20.5]. The obtained weight centers at *Step* 3 are: C^{best} = (492.5; 87) and – C^{worst} = (496; 86). Direction *d* = (–3.5; 1). The following solutions: t^1=(489; 88), t^2=(485; 89), t^3=(481; 90) are computed at *Step* 3. The corrected reference solution and the reference point calculated at *Step* 4 are x^r= (435; 90), f^r = (2.2936; 10.989). At *Step* 5 the weight center C^i = (493.5; 86.6). The final population after the first iteration is as follows: $x^1 = x^2 = x^7$ = (455; 89), $s^1_r = s^2_r = s^7_r$ = (2.1930; 11.1111), $x^3 = x^4 = x^8 = x^9 = x^{10}$ = (435; 90), $s^3_r = s^4_r = s^8_r = s^9_r = s^{10}_r$ = (2.2936; 10.989), x^5 = (433; 90), s^5_r = (2.3041; 10.989), x^6 = (442; 89), s^6_r = (2.2573; 11.1111);

The DM is satisfied with the found solution x^3 and the procedure terminates.

For comparison at the same starting conditions the obtained population after the first SPEA iteration is:

x^1 = (500; 50), s^1_r = (1.996; 19.608), x^2 = (506; 29), s^2_r = (1.972; 33.333), x^3 = (482; 30), s^3_r = (2.070; 32.258), x^4 = (493; 52), s^4_r = (2.024; 18.868), x^5 = (477; 41), s^5_r = (2.092; 23.810), x^6 = (508; 54), s^6_r = (1.964; 22.727), x^7 = (504; 40), s^7_r = (1.98; 24.39), x^8 = (489; 43), s^8_r = (2.04; 22.73), x^9=(493; 40), s^9_r =(2.024; 24.39), x^{10} =(504; 52), s^{10}_r =(1.980; 18.868);

V. Conclusions

The proposed exact interactive evolutionary method has the following advantages:

It is an interactive method. The Decision Maker is supported in the choice of a suitable reference point, so that he/she can easily direct the search process to the desired region.

The method does not generate an approximation of whole efficient frontier. The DM has the possibility to investigate different possibly small parts from efficient frontier according to his/her preferences.

The method generates exact Pareto efficient solutions. The population movement to the Pareto frontier is realized by an accelerated heuristic procedure.

Increasing the number of objectives does not affect the performance of the search procedure.

The proposed method can be used for solving linear and nonlinear multiple objective optimization problems, having continuous or integer variables.

Acknowledgement

This study is partially supported by the project № BG161PO003-1.1.06-0083 to the EU operative program „Development of Bulgarian economy competitiveness" and by the project AComIn "Advanced Computing for Innovation", grant 316087, funded by the FP7 Capacity Programme (Research Potential of Convergence Regions).

References

[1] J. Branke, K. Deb, K. Miettinen, R. Slowiń ski, *Multiobjective Optimization. Interactive and Evolutionary Approaches.* (Berlin Heidelberg, Springer-Verlag, 2008).

[2] C. A. C. Coello & G. B. Lamont, *Applications of Multi-Objective Evolutionary Algorithms*, (World Scientific, Singapore, 2004).

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

[4] J.G. Ecker, N. S. Hegner and I.A. Kouada, „Generating All Maximal Efficient Faces for Multiple Objective Linear Programs", *Journal of Optimization Theory and Applications*, Vol. 30, No 3, (1980) pp. 353-381.

[5] T. Gal, "A General Method for Determining the Set of All Efficient Solutions to a Linear Vectormaximum Problem", *European J.l of Operat. Res.*, Vol. 1, No 5, (1977) pp. 307–322.

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

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

[8] V. Guliashki, H. Toshev, C. Korsemov, "Survey of Evolutionary Algorithms Used in Multiobjective Optimization", *Problems of Engineering Cybernetics and Robotics*, v. 60, (2009), pp. 42-54.

[9] V. Guliashki, L. Kirilov, "SPEA-Based Method for MCDM Convex Integer Problems", *Cybernetics and Information Technologies – Journal of Bulgarian Academy of Sciences*, Vol.9, No4, (2009) pp. 93-101,

[10] V. Guliashki, L. Kirilov, K. Genova, "An evolutionary algorithm for integer multicriteria optimization (EVALIMCO)", In: *Proceedings of the 10-th International FLINS Conference, Uncertainty Modeling in Knowledge Engineering and Decision Making*, 26-29.August, 2012, Istanbul, Turkey, pp. 118-123.

[11] V. Guliashki, Genova K., Kirilov L., (2013) "Hybrid Evolutionary Algorithm for Integer Multiple-Objective Optimization Problems", In: *Proceedings of Papers of the XLVIII International Scientific Conference on Information, Communication and Energy Systems and Technologies ICEST2013*, (Editor Prof. Dr. Mitrovski, C.), June 26 – 29, 2013, Ohrid, Macedonia, vol. I, pp. 253-256.

[12] A. Jaszkiewicz and R. Slowinski, "The Light Beam Search approach – an overview of methodology and applications", *European J. of Operational Research*, 113(2):300-314, 1999.

[13] Korhonen, P., Wallenius, J., VIG – A Visual and Dynamic Decision Support System for Multiple Objective Linear Programming, *Multiple Criteria Decision Making and Risk Analysis Using Microcomputers*, Edited by B. Karpak, S. Zionts, Springer-Verlag, Berlin Heidelberg, (1989) pp. 251-281.

[14] K. Miettinen, L. Kirilov, "Interactive Reference Direction Approach Using Implicit Parametrization for Nonlinear Multiobjective Optimization", *Journal of Multi-Criteria Decision Analysis*, vol.13, (2005) pp. 115-123.

[15] H. Nakayama, Aspiration Level Approach to Interactive Multi-Objective Programming and Its Applications, *Advances in Multicriteria Analysis*, Kluwer Academic Publishers, Dordrecht, (1995), pp. 147-174.

[16] S.C. Narula, Kirilov L., Vassilev V., Reference Direction Approach for Solving Multiple Objective Nonlinear Programming Problems, *IEEE Transactions on Systems, Man, and Cybernetics*, vol.24, No5, (1994) pp. 804-806.

[17] S. C. Narula, and V. Vassilev , An Interactive Algorithm for Solving Multiple Objective Integer Linear Programming Problems. *European Journal of Operational Research*, 79, (1994) pp. 443-450.

[18] K. Praditwong, Xin Yao, „How well do multi-objective evolutionary algorithms scale to large problems", In: *Proceedings of IEEE Congress on Evolutionary Computation*, CEC 2007. Singapore, 25-28 September 2007, pp. 3959 – 3966.

[19] E. Zitzler, M. Laumanns and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm for multiple objective optimization", In: *Proceedings of EUROGEN2001 Evolutionary methods for design, optimization and control with applications to industrial problems*, (2001), pp. 95-100.