Lecture Notes in Control
and Information Sciences
Edited by M. Thoma and A. Wyner
v.113, pp. 231–240

M. Iri, K. Yajima (Editors)
System Modelling and Optimization
Proceedings of the 13th IFIP Conference
Tokyo, Japan, August 31 – September 4, 1987

Springer-Verlag
Berlin Heidelberg New York London Paris Tokyo

MULTICRITERIA OPTIMIZATION IN THE DISO SYSTEM

Evtushenko Y., Mazourik V., Ratkin V.
Computing Center of the USSR Academy of Sciences
Moscow B-333, Street Vavilova 40, U.S.S.R.

(Revised version 15 January 2004)

1. INTRODUCTION

From the mathematical point of view *multicriteria optimization* (MCO) is a natural generalization of optimization problems. The need of decision making in contradictory situations makes MCO methods so interesting for us. MCO deals with one of the most sophisticated aspects of human activity which is to achieve several goals by the single act of decision making. MCO models and ordinary optimization are not very much different in task definition, giving us hope to use the similar numerical methods.

This paper gives the overview of the MCO package as one of the main parts of the DISO — *dialogue system for optimization problem solving* which was developed in the Computing Center of the USSR Academy of Sciences. Two MCO methods are described in this paper. Both methods are based on the idea of non-uniform covering technique and inclusion function approach, which was initially developed for global extremum search [1]–[5]. The complexity of MCO tasks makes it necessary to create effective numerical methods to find both a single point of *Pareto set* and an approximation of this set also. The paper describes two MCO algorithms which differ in the interpretation of the solution and as a consequence in the complexity of numerical calculations. The main features of the MCO package are described also.

2. OVERVIEW OF THE DISO SYSTEM

The basic feature of the DISO system is the integration principle. Unlike other dialogue systems of this class the DISO system includes several interconnected packages for the solution of the following tasks:

- unconstrained minimization;
- nonlinear programming;
- optimal control;
- linear programming;

- global optimization;
- multicriteria optimization;
- linear algebra;
- nonlinear algebraic equations.

For all these tasks the DISO system delivers for the user the unified set of dialogue capabilities, which includes:

- task definition and analysis in text mode;
- automatic optimization class recognition and correspondent dialogue package initiation;
- local analysis of task definition functions in a given point;
- changing of optimization method and its control parameters in the dialogue session;
- asynchronous control of the solution process which makes it possible to stop the process at any moment;
- control of the hierarchical solution process with the automatic or manual creation of the subordinate tasks from the list mentioned above;
- choice of the numerical or analytical differentiation schemes;
- control of the numerical and graphical interaction in the process of solution.

From the implementation point of view dialogue capabilities of the DISO system are based on the new approach of the multiwindow technique which includes the capability to access the values of the variables using special fields in the windows. The methods in each optimization class have different forms of graphical output in accordance with their basic mathematical schemes.

The integrated mode of the DISO system makes it really power-full instrument for the solution of different application tasks. It is essential, that the system makes it possible to change the task definition in the process of the solution and to transfer the task from one mathematical model to another. The typical example of such transformation takes place if you add a restrictions to the initial unconstrained optimization problem thus creating nonlinear programming problem. Another example: transforming all the criteria but one to the restrictions will change the MCO problem to the more simple class of nonlinear programming. It is important to note that all the numerical results achieved so far will remain accessible after these transformations.

Most optimization models have hierarchical structure. The nonlinear programming methods, for example, may reduce the task to unconstrained minimization on every iteration. Optimal control methods usually create subordinate tasks of nonlinear programming and so on. In all these cases the DISO system provides the dialogue capabilities of the correspondent class for the solution of the subordinate problem. Finishing the solution will move the user again to the level of the initial problem. This feature of the DISO system proved to be really valuable for the solution of several difficult applied problems saving time and increasing the accuracy of the solution.

The MCO dialogue package plays a central role in the DISO system, because its mathematical model can be obviously treated as a generalization of the other optimization models listed above.

3. STATEMENT OF MCO PROBLEM

The multicriteria problem that we consider has the following form:

$$\min_{x \in X} F(x), \tag{1}$$

where $x \in \mathbb{R}^n$ is a vector of decision variables, X is the feasible decision set, $F = [F^1, F^2, \dots, F^m]$ is the objective vector, $F : \mathbb{R}^n \to \mathbb{R}^m$, vector-function F is continuous on X. The decision set X is assumed to be a closed and bounded (therefore, compact). The goal is to find the efficient (Pareto) set X_* of X with respect to F, that is

$$X_* = \{x \in X : \text{ if } F(w) \le F(x) \text{ for some } w \in X, \text{ then } F(w) = F(x)\}. \tag{2}$$

We shall use the following convention: if $a, b \in \mathbb{R}^s$, then $a \le b$ if and only if $a^i \le b^i$ for all $1 \le i \le s$.

We propose two extensions of ε-optimality concept which were developed for scalar optimization problem to vector case. In the first extension we introduce ε-efficient set as follows:

$$X_*^\varepsilon = \{x \in X : F(x) \le F(x_*) + \varepsilon, \text{ where } x_* \in X_*\}, \tag{3}$$

where vector $\varepsilon \in \mathbb{R}^m$ has all positive components and is named accuracy vector.

The set W_*^ε is called ε-net of the Pareto set if:

1) for any point $x \in X_*$ there exists a point $z \in W_*^\varepsilon$ such that $F(z) \le F(x) + \varepsilon$;

2) there are no two different points x and z in W_*^ε such that $F(x) \le F(z)$.

Let X and P_i be a compact right parallelepipeds parallel to the coordinate axis (abbreviated as a box in the sequel):

$$X = \{x \in \mathbb{R}^n : a \le x \le b\},$$

$$P_i = \{x \in \mathbb{R}^n : a_i \le x \le b_i\}, \quad P_i \subset X, \ a_i \in \mathbb{R}^n, \ b_i \in \mathbb{R}^n, \quad i = 1, 2, \dots.$$

The main diagonal of the box P_i we denote as $d_i = b_i - a_i$, the midpoint of the box is $c_i = (1/2)(b_i + a_i)$.

Let's introduce the m-dimensional vector-function $Q(P)$, for which every j-th component is defined by the condition:

$$Q^j(P) = \min_{x \in P} F^j(x), \qquad P \subset X.$$

We assume that for $Q(P)$ it is possible to find vector-function $G(P)$ which is the lower estimation of $Q(P)$ on the box P. This function must satisfy two conditions:

$$G(P) \le Q(P), \tag{4}$$

$$\lim_{\|d_i\|_\infty \to 0} (G(P_i) - Q(P_i)) = 0. \tag{5}$$

Here we introduced the sequence of the boxes which satisfies the following conditions:

$$P_{i+1} \in P_i, \qquad \lim_{\|d_i\|_\infty \to 0} P_i = P_\infty \in X, \qquad i = 1, 2, \dots,$$

where P_∞ is accumulating point.

For the given vector-function $F(x)$ the vector-function $G(P)$ can be found either on the basis of interval analysis [6] or by introducing some additional hypothesis. For example, supposing that all the components of $F(x)$ on the X set satisfy the Lipschitz condition with constants $L^j = \sum_{i=1}^{n} \max_{x \in X} \left| \frac{\partial F^j(x)}{\partial x^i} \right|$, $1 \le j \le m$, we have:

$$G^j(P_i) = F^j(c_i) - \frac{1}{2} \cdot L^j \cdot \|d_i\|_\infty. \tag{6}$$

If in addition $F(x)$ satisfies Lipschitz condition with constants $M^j = \sum_{k=1}^{n}\sum_{i=1}^{n} \max_{x\in X} \left|\frac{\partial^2 F^j(x)}{\partial x^i \partial x^k}\right|$, $1 \le j \le m$, then:

$$G^j(P_i) = F^j(c_i) - \frac{1}{2}\|d_i\|_\infty \min\{L^j, |F_x^j(c_i)|_1 + \frac{1}{4}M^j\|d_i\|_\infty\}. \tag{7}$$

It is obvious that these functions $G(P)$ satisfy conditions (4), (5).

4. DESCRIPTION OF MCO METHODS

Now we are going to describe two algorithms for the approximate solution of the problem (1). During the computation process the algorithms will generate the sequence of the boxes

$$B_k = \{P_1, P_2, \dots, P_k\}, \quad \text{all} \quad P_i \subset X,$$

and the corresponding sequence of these boxes midpoints

$$N_k = \{c_1, c_2, \dots, c_k\}.$$

Let each box P_i be linked with the structure $S_i = (c_i, d_i, G_i)$, where $G_i = G(P_i)$. We call the set S for the sequence B_k the structure list

$$S = \{S_1, S_2, \dots, S_k\}.$$

Algorithms differ in the interpretation of the solution and time consuming. The W_*^ε set is obtained by one of them (second algorithm) and a single point x_r from the X_*^ε set — by another (first algorithm). The general scheme of both algorithms is described below.

Algorithm.

Initial actions:

1). Let $P_1 = X$, calculate c_1, d_1, $F(c_1)$, $G_1 = G(P_1)$ and set $B_1 = \{P_1\}$, $S_1 = (c_1, d_1, G_1)$, $S = \{S_1\}$.

Main cycle:

2). Choose the box P_s from B_k, for which $\min\limits_{1\le i\le k} \max\limits_{1\le j\le m} G_i^j$ is achieved.

3). Choose a coordinate direction t in box P_s, parallel to which P_s has an edge of maximum length, i.e. $d_s^t = \max\limits_{1\le i\le n} d_s^i$. Bisect P_s in the direction t, getting boxes P_α, P_β with midpoints c_α, c_β, and diagonals d_α, d_β, respectively.

4). Calculate $F(c_\alpha)$, $F(c_\beta)$, modify W_k (or x_r) and define the vector values $G_\alpha = G(P_\alpha)$, $G_\beta = G(P_\beta)$.

5). Remove the box P_s from the sequence B_k, i.e. remove the structure S_s from the list S. Add into the list S two structures: $S_\alpha = (c_\alpha, d_\alpha, G_\alpha)$ and $S_\beta = (c_\beta, d_\beta, G_\beta)$, assuming $S_s = S_\alpha$ and $S_{k+1} = S_\beta$.

6). For all S_i from the list S check the following condition: if $F(x_j) \le G(P_i) + \varepsilon$ for any x_j from W_k (or for x_r), then remove S_i from the list S and remove P_i from B_k. Order new list $\{S_{i_1}, S_{i_2}, \dots, S_{i_p}\}$ and give it the name $S = \{S_i\}_{1\le i\le p}$.

7). Let $k = p$. If $k \neq 0$, i.e. S is not empty, then go to 2.

Concluding operations:

8). Invoke output procedure. Stop computation.

The construction rule for W_*^ε.
Let $W_1 = \{c_1\}$.
Let we have N_k, N_{k+1}, W_k. Then
i) if there exists $x_i \in W_k$ such that $F(x_i) \leq F(x_{k+1})$, then $W_{k+1} = W_k$;
ii) otherwise $W_{k+1} = (W_k \backslash V) \cup \{x_{k+1}\}$, where $V = \{x_i \in W_k : F(x_{k+1}) \leq F(x_i)\}$.

The above (second) algorithm defines as a result the ε-net of the Pareto set (i.e. $W_q = W_*^\varepsilon$) using the finite number of F evaluations.

The rule for finding a single point from X_*^ε.
Let $x_r = c_1$.
Let c_k was obtained. Then if $F(c_k) \leq F(x_r)$, then $x_r = c_k$.

The above (first) algorithm defines the point $x_r \in X_*^\varepsilon$ using the finite number of F evaluations.

5. PROBLEM SOLVING IN MCO PACKAGE

The task below was solved in order to investigate the quality of MCO algorithms. Full definition of this task is given in [7]. The solution based on the sequence of minimization subtasks was found there. This task includes two criteria, five parameters and has the following form:

$$\min_{x \in X} F(x), \tag{8}$$

where

$$F^1(x) = 1 - \prod_{i=1}^{n} \left[1 - (1 - r^i)^{x^i+1}\right], \qquad F^2(x) = \sum_{i=1}^{n} c^i \cdot x^i,$$

$X = \{x^i \in N : 0 \leq x^i \leq 10,\ 1 \leq i \leq n\}$, $n = 5$, vectors c and r are given in the Table 1.

Two solutions were obtained using both algorithms with different accuracy vectors ε. Vector function G was defined in accordance to (6). Lipschitz constants vector was chosen to be $L = [0.7, 0.7]$. This value is good higher estimation for the given vector function. The results for the first algorithm are given in Table 2 for different accuracies. Table 3 contains the results for the second algorithm with the accuracy vector $\varepsilon = [0.1, 0.35]$.

t	1	2	3	4	5
c^i	0.13	0.13	0.15	0.14	0.15
r^i	0.90	0.75	0.65	0.80	0.85

Table 1. Coefficients for task (8).

accuracy		criteria		parameters					crit.
ε^1	ε^2	$F^1(x)$	$F^2(x)$	x^1	x^2	x^3	x^4	x^5	evaluations
0.20	0.50	0.575	0.43	0	0	0	2	1	163
0.15	0.40	0.444	0.44	0	0	1	1	1	335
0.10	0.35	0.336	0.55	1	1	1	1	0	591

Table 2. Single points from the X_*^ε set for different accuracies.

N°	$F^1(x)$	$F^2(x)$	x^1	x^2	x^3	x^4	x^5
1	0.096	1.12	1	2	2	2	1
2	0.125	0.98	1	2	2	1	1
3	0.166	0.85	1	1	2	1	1
4	0.198	0.83	1	2	1	1	1
5	0.236	0.70	1	1	1	1	1
6	0.305	0.57	0	1	1	1	1
7	0.336	0.55	1	1	1	1	0
8	0.396	0.42	0	1	1	1	0
9	0.446	0.41	1	1	1	0	0
10	0.497	0.28	0	1	1	0	0

Table 3. ε-net of the Pareto set W_*^ε for $\varepsilon = [0.1, 0.35]$.

6. MCO PACKAGE DIALOGUE CAPABILITIES

As was already mentioned above, the DISO system provides the user the unified set of dialogue capabilities to control every step of the solution process from the task definition up to the analysis of the numerical results. Let's overview briefly these capabilities.

The MCO task definition can be done using text processor which is part of the DISO system. A special language DIFALG is used for task definition. This language is very similar to the ALGOL-60. The inner form of the function representation is created as a result of the task definition compilation process. The calculation of the numerical values of the functions in a given point is based then on the interpretation of this inner form. The peculiarity of the DIFALG language is that its semantics includes the notion of the differentiation. Function evaluation may be done in parallel with the first and second derivatives of this function in a given point by the user request. The important point is that it gives the user not the numerical approximation but the exact value of the derivatives, which corresponds to the analytically evaluated value. The differentiation algorithms are based on a special highly effective approach which qualitatively increases the calculation speed.

Task definition includes comment lines. The system automatically defines the optimization class and passes the control to the correspondent dialogue monitor analyzing these comments. For example, to define MCO problem, the user marks with the special comment those functions in the DIFALG listing which he wants to be included to the set of the criteria. The other comments mark the functions to be included to the equality and inequality restriction sets. If any, the problem will be classified as nonlinear (single or multicriteria) programming model. Special comments define the mode of optimization: local or global, give initial point value, parallelepipedal restrictions in the parameter space etc. The user may return to the task definition step at any moment of the solution, make some modifications and continue the solution process.

The user initiates the dialogue session finishing the task definition. The set of control windows become available to him at this moment. Each window provides the user with some resources to control the solution process. The MCO dialogue monitor opens the access to the three windows.

The first one provides the capabilities for manual analysis of the functions local properties in a given point. It also controls the MCO method choice, initiation and termination. As was mentioned above, the DISO windows are structured in a sense that they include the different sets of fields to access the variables and control their values. The first MCO window contains the fields for the decision vector, criteria values vector, list of methods available etc. The user may move the cursor to the parameter vector and give this vector some initial value. The cursor movement to the criteria field automatically yields in the recalculation of the criteria functions in a given point with immediate output of the numerical results in this field. The field which is connected with the list of available methods plays the role of the menu. Each method has its own list of control parameters. The choice of the method automatically provides the user with the new list of fields for control parameters which appear in the window and become accessible by the ordinary routine of the cursor movement. Another field is responsible for the type of the solution: the user can choose ε-approximation of the Pareto set or single-point solution as it was mentioned above. Finally, the window has the menu field to run the method, stop it and switch to the task modification mode.

The second window provides the user with the view of the Pareto set in the process of its creation. With the help of the fields in this window the user can point out two coordinate axis to define the two-dimensional plain to create the projection of the Pareto set. The solution process will show each new Pareto point in the special graphical field of this window. Finishing the solution process the user may enter the mode of Pareto set analysis. The cursor will take the form of the arrow in the Pareto field. The user can move this arrow from one Pareto point to another, visualizing correspondent numerical values of criteria and parameter vectors. The user can store the Pareto set in a file with a given name or retrieve previously defined Pareto set and continue the solution to achieve more accurate Pareto approximation.

The third window gives the possibility to visualize the covering technique in the parameter space. Each method has its own covering strategy. The user can visually estimate the efficiency of the chosen covering scheme for the given task to be solved.

The system provides the possibility of asynchronous control which includes, for example, switching from one window to another while the optimization process continues it's progress.

7. CONCLUSION

The MCO methods described in this paper were tested using several tasks and proved to be rather effective. There exist several reasons of the methods success. First, unlike the known methods, these methods are strongly oriented on the effective use of the computer memory which decreases the amount of function evaluation. This is most valuable feature if function evaluation takes long time. Second, the interval analysis technique makes it possible to eliminate the initial Lipschitz constant estimation. Third, there exists a simple and natural way to organize parallel calculations by the feasible domain division between several processors. Fourth, the described methods permit the inclusion of local search algorithms, which may speed the calculations enormously. Fifth, the second of the proposed methods creates the Pareto set using nonunified one-way covering technique, instead of the common approach, which is based on the manifold solution of the auxiliary tasks on the feasible domain.

Dialogue MCO package, which is part of the DISO system, is a promising numerical basis for the implementation of numerous decision support systems.

References

[1] Evtushenko, Y.G. *Numerical Method for the Global Search.* Zhurnal Vychislitelnoi Matematiki i Matematicheskoi Fiziki, Vol.11, N° 6, 1971, pp. 1390–1403.

[2] Evtushenko, Y.G. *Numerical Optimization Technique.* Springer-Verlag, New-York, 1985.

[3] Evtushenko, Y.G., Potapov, M.A. *A Hondifferentiable Approach to Multlcrlterla Optimization.* Lecture Notes in Economics and Mathematical Systems. Springer-Verlag. 1984, 255: pp. 97–102.

[4] Ratschek, H. *Inclusion Functions and Global Optimization.* Mathematical Programming, 33 (3), 1985, 300–317.

[5] Evtushenko, Y.G., Ratkin, V.A. *Bisection Method for Global Optimization* (in Russian). Izvestija Akademii Nauk SSSR, Tehnicheskaya Kibernetika, N° 1, 1987, pp. 119–127.

[6] Moore, R.E. *Interval Analysis.* Prentice-Hall, Englewood Clifs, 1966.

[7] Saaty T.L. *Optimization in Integers and Related Extremal Problems.* McGraw-Hill, New York, 1970.