Linear Multicriteria Decision Support System

K. Genova, M. Vassileva, V. Vassilev, F. Andonov

Abstract: The paper presents a multicriteria decision support system, designed to model and solve linear problems of multicriteria optimization. The system is developed on the basis of an interactive classification-based algorithm, which allows the decision makers describe their local preferences with the help of desired and acceptable levels, directions and intervals of change in the values of a part or of all the criteria. The structure and the user's interface of the system are described.

Key words: Multicriteria optimization, Interactive algorithms, Multicriteria decision support systems.

INTRODUCTION

The multicriteria decision support systems (MDSS) are designed to assist the decision maker (DM) in solving not well-formalized problems of multicriteria optimization (MO). The MDSS developed [8] are classified in three groups: commercial, teaching and experimental (for new methods testing). Sometimes it is difficult to make a clear distinction between these groups. Some experimental MDSS can be successfully applied for research and teaching purposes. The software realization and the documentation of many teaching MDSS is very well accomplished, but offered free, without any commercial purpose. The status of the multicriteria decision support systems until 1996 is discussed in [8]. Among the well-known MDSS, which solve problems of MO, the systems VIG, DIDAS, ADBASE, MOLP-16, MONP-16, MOIP, NIMBUS can be pointed out. New experimental systems [6] or versions of old systems modified to work in Internet environment [4] have been developed in recent years.

Each MDSS solving problems of MO realizes different multicriteria and single criterion algorithms. In most of the MDSS developed up to now, basic attention is paid to the interactive multicriteria algorithms. Two types of interactive multicriteria algorithms are included in these MDSS: The first type comprises interactive algorithms of the reference point and of the reference direction [2]. These are MDSS systems such as DIDAS, VIG. The second type of interactive algorithms includes classification-based algorithms [4, 6]. These interactive algorithms are built in MDSS systems NIMBUS, MOLP-16, MONP-16, MOIP.

An experimental MDSS, called MOLIP, is described in this paper. It is designed to solve linear problems of MO. The system operates under MS Windows operating system and realizes an interactive classification-based multicriteria algorithm [7]. The interactive multicriteria algorithm allows the DM define not only the aspiration levels at each iteration, as it is in most of the interactive algorithms known up to now, but also set aspiration intervals and aspiration directions of change in the values of the separate criteria. In this way the DM can describe his/her preferences with greater precision, flexibility and reliability.

THE CLASSIFICATION-BASED INTERACTIVE ALGORITHM

The classification-based algorithm, called GAMA-L, which is included in MOLIP system, is designed to solve linear problems of multicriteria optimization.

The linear problem of MO (denoted by LMK), can be formulated as follows:

\[
\begin{align*}
\text{max} & \{ f_k(x) \}, \quad k \in K, \\
\text{under constraints:} & \\
\sum_{j \in N} a_{ij} x_j & \leq b_i, \quad i \in M, \\
0 & \leq x_j \leq d_j, \quad j \in N,
\end{align*}
\]
where the symbol “max” means that all the objective functions should be simultaneously maximized. \( K = \{1, 2, ..., p\} \), \( M = \{1, 2, ..., m\} \), and \( N = \{1, 2, ..., n\} \) are the indices sets of the linear criteria, of the linear constraints and of the variables respectively; \( f_k(x), \quad k \in K \), are the linear criteria; \( f_k(x) = \sum_{j=1}^{n} c_{jk} x_j, \quad x = (x_1, x_2, ..., x_j, ..., x_n) \) is the vector of variables. The constraints (2)-(4) define the feasible set of the variables. \( X \) will denote this set.

The scalarization problem used is a basic part of each modern interactive algorithm of multicriteria optimization. The classification-based scalarizing problems [7] decrease the requirements towards the DM when comparing and evaluating the new solutions obtained. With respect to the information, required from the DM in the search of new solutions, these scalarizing problems are relatively near to the scalarizing problems of the reference point [2], but unlike them, here the DM is not obliged to determine the desired or acceptable levels for all the criteria. In the scalarizing problems, suggested in this chapter, the DM can represent his/her local preferences not only by desired or acceptable levels, but also by desired or acceptable directions and intervals of change in the values of separate criteria. In this way he/she can describe his/her local preferences with greater flexibility, accuracy and reliability. Depending on these preferences, the set of the criteria at each iteration can be indirectly divided into seven or less than seven classes, denoted as follows: \( K^+, K^-, K^\geq, K^\leq, K^>, K^< \) and \( K^0 \). Each criterion \( f_k(x), \quad k \in K \) may belong to one of these classes, as given below: \( k \in K^+ \), in case the DM wishes the criterion \( f_k(x) \) to be improved; \( k \in K^- \), if the DM wants the criterion \( f_k(x) \) to be improved by a desired value \( \Delta_k > 0 \); \( k \in K\geq \), in case the DM wishes that the current value of the criterion \( f_k(x) \) is not deteriorated; \( k \in K\leq \), in case the DM agrees the criterion \( f_k(x) \) to be deteriorated; \( k \in K^> \), if the DM wishes the criterion \( f_k(x) \) to be deteriorated by an acceptable value \( \delta_k > 0 \); \( k \in K^< \), if the DM wishes the criterion \( f_k(x) \) not to be altered beyond the limits of a given interval, determined as: \( f_k - t^-_k \leq f_k(x) \leq f_k + t^+_k ; \quad k \in K^0 \), in case the DM agrees the criterion \( f_k(x) \) to change freely at this iteration.

In order to obtain a solution, which is better than the current Pareto optimal solution of the linear problem of MO, the following scalarizing problem \( L1 \) can be used on the basis of the implicit criteria classification done by the DM:

Minimize:

\[
\min \left\{ \alpha + \beta + \rho \sum_{k \in K} y_k \right\}
\]

under constraints:

\[
\alpha \geq (\overline{f}_k - f_k(x))/\overline{f}_k, \quad k \in K^+ \]

\[
\alpha \geq (\underline{f}_k - f_k(x))/\underline{f}_k, \quad k \in K^\geq \cup K^\leq \]

\[
\beta \geq (\overline{f}_k - f_k(x))/\overline{f}_k, \quad k \in K^-, \]

\[
f_k(x) \geq f_k, \quad k \in K^\geq \cup K^\leq \cup K^> \]

\[
f_k(x) \geq f_k - \delta_k, \quad k \in K^\leq \]

\[
f_k(x) \geq f_k - t^-_k, \quad k \in K^> \]

\[
f_k(x) \geq f_k + t^+_k, \quad k \in K^< \]
\[ f_k(x) = y_k, \quad k \in K^+, \quad (13) \]

\[ f_k(x) = y_k, \quad k \in K^+ \cup K^-, \quad (14) \]

\[ f_k(x) - f_k = y_k, \quad k \in K^-, \quad (15) \]

\[ \alpha, \beta - \text{arbitrary}, \quad (16) \]

\[ x \in X, \quad (17) \]

where \( f_k \) is the value of the criterion \( f_k(x) \) in the current preferred Pareto optimal solution;

\[ \tilde{f}_k = f_k + \Delta_k \] is the desired level of the criterion \( f_k(x) \);

\( \rho \) is a scaling coefficient, defined as follows:

\[
f_k = \begin{cases} 
\varepsilon, & \text{if } |f_k| \leq \varepsilon \\
 f_k, & \text{if } |f_k| > \varepsilon 
\end{cases}, \quad (18) \]

\( \rho \) and \( \varepsilon \) are small positive numbers.

Problem L1 is a linear programming problem [5]. A Pareto optimal solution of LMK problem is obtained with the help of this problem. In order to obtain more Pareto optimal solutions of LMK problem, some problems could be used, that are parametric extensions of problem L1. One parametric extension of problem L1, called L1P, is obtained after the replacement of constraints (6)-(8) of problem L1 by the following constraints:

\[ f_k(x) + f_k \alpha \geq \tilde{f}_k + \Delta f_k \tau, \quad k \in K^+, \quad (19) \]

\[ f_k(x) + f_k \alpha \geq f_k - \Delta f_k \tau, \quad k \in K^+ \cup K^-, \quad (20) \]

\[ f_k(x) + f_k \beta \geq f_k + \Delta f_k \tau, \quad k \in K^-, \quad (21) \]

\[ t \geq 0 \quad (22) \]

where \( \Delta f_k \) is a parameter.

GAMMA-L interactive algorithm is developed on the basis of scalarizing problems L1 and L1P. It is an interactive algorithm oriented towards learning [1], which means that the existence of an implicit DM’s utility function is not presumed. The DM can seek freely Pareto optimal solutions in the set of the Pareto optimal solutions, evaluating on his/her own whether the current solution found is the most preferred or is the final solution of the multicriteria problem.

The main steps of GAMMA-L interactive algorithm are as follows:

Step 1. Finding an initial Pareto optimal solution of the multicriteria problem by setting \( f_k = 1 \), \( k \in K \) and \( \tilde{f}_k = 2 \), \( k \in K \), and solving problem L1.

Step 2. Representing of the current Pareto optimal solution obtained to the DM for evaluation. If the DM considers, that this Pareto optimal solution satisfies his/her global preferences, Step 6 is executed, otherwise – Step 3.

Step 3. A request to the DM to determine his/her local preferences for improving the current Pareto optimal solution found by defining desired or acceptable levels, directions and intervals of change of a part or of all the criteria.

Step 4. A requirement towards the DM to estimate whether he/she wishes to take into account one or more new Pareto optimal solutions in the evaluation. In the first case scalarizing problem L1 is solved and Step 2 is executed, and in the second case – Step 5 is accomplished.

Step 5. A question to the DM to determine the maximal number \( s \) of new Pareto optimal solutions that he/she wishes to obtain. Solution of scalarizing problem L1P and representing of less or equal to \( s \) new Pareto optimal solutions for evaluation and for
choice of a current preferred solution. In case the DM decides that this Pareto optimal solution satisfies his/her global preferences, Step 6 is executed, otherwise – Step 3.

Step 6. Stop of the process of the linear multicriteria problem solving.

LINEAR MULTICRITERIA DECISION SUPPORT SYSTEM MOLIP

MOLIP system consists of the following three main modules: a control program, interface modules and optimization modules.

The control program is an integrated software environment for creating, processing and saving of files associated with MOLIP system (ending by “.mlp” extension) and also for linking and executing different types of software modules. The basic functional possibilities of the control program can be divided in three groups. The first group includes possibilities to use the standard for MS Windows applications menus and system functions – “File”, “Edit”, “View”, “Window”, “Help” and others in MDSS own environment. The second group of control program facilities covers the control of the interaction between the modules realizing:
- creating, modification and saving of “.mlp” files associated with MOLIP system, which contain input data and data concerning the process and the results from solving MO linear problems;
- interactive solution of the linear MO problems, which have been entered;
- localization and identification of errors occurring during MDSS operation.

The third group of control program functional features consists of possibilities for visualization of important information concerning the DM and the system operation as a whole.

The interface modules accomplish the dialogue between the DM and MDSS during the entry and correction of the input data of the multiobjective problems being solved, during the interactive process of these problems solution, and also in dynamic digital and graphic visualization of the process main parameters. The editing module enables the entry, alteration and storing of the descriptions of the criteria, the constraints, and also the variables type and limits of alteration. Two types of graphic representation of the information about the criteria values at different steps and some possibilities for comparison, are provided by another interface module. Dynamic Help is also available, which shows information about the purpose and way of use of each one of the fields and radio buttons.

GAMA-L classification-based algorithm is built in the optimization modules of MOLIP system. In addition to it, the optimization modules include also a software realization of a single criterion algorithm of linear programming [5], designed to solve scalarizing problems L1 and L1P. This program realization is generated by LINDO Callable Library [3].

The entry and correction of the problem criteria and constraints is realized in “MOLIP Editor” window. Every criterion and every constraint is entered separately in the respective text field for edition. Syntax check is accomplished when they are added to the data already entered. The syntax accepted is similar to the mathematic record of this class of optimization problems. Fig. 1 shows an illustrative example with two objective functions and three constraints.

In order to generate an initial feasible solution the DM is given two alternatives (Fig. 1): “Auto generated” or “Entered by user”. Pressing Next opens a window, which sets the initial values of the criteria, then with the help of Accept button these values are saved and after that “MOLIP Solving” window (Fig. 2) is activated. This window is divided into several zones. Its upper part contains a band with buttons that realize the main functions of the process for interactive solution of MO linear problems. These are the buttons:

Solve - for starting the optimization module in order to find a new current solution of MOLIP, solving the scalarizing problem generated at this iteration;
Info - for visualization of the variables values at the current solution considered in a separate window;

![MOLIP - Editor](image)

**Graphic** - for opening the window for graphic comparison of the results obtained at separate steps.

**Back** and **Forward** - buttons for navigation. They allow the DM go back to preceding steps and reconsider the solutions found. In case the DM wishes, he/she can change his/her own preferences concerning the criteria alteration at any of the previous steps and start the process for better solution search from there on;

**Options** - for opening different system setups: of the data file, which is active at the moment - it can be associated with “.mlp” extension; changing the names of the system variables if “alfa” and “beta” have another user’s meaning in the problem being solved; changing the values of the default parameters of the scalarizing problems solved;

**Help** - for output of help information with basic directions about entry, editing and solving of MO linear problems in MDSS environment;

**About** - for providing information about the team and system information about the computer system used;

**Exit** - for MOLIP system exiting with or without storing in a file the data and the results from the recent operation.

The next field of MOLIP Solver window contains radio buttons for setup of the MOLIP solution looked for: continuous Pareto optimal. The other radio buttons - Exact integer, Nearest integer, Heuristic integer, as well as Weak Pareto optimal are included for expansion of MOLIP system in solving linear integer multicriteria problems as well.

Two text fields follow. The first one outputs successively the values of the criteria obtained at the current step. It is an operating field where DM’s preferences relating to the search of the next solution are set. After marking each one of the criteria, a context field is opened with the help of the mouse right button, where the DM sets the desired or acceptable alteration in the value of this criterion at a following iteration. In case the selection is connected with the necessity to enter a particular value, MOLIP system opens an additional dialogue window and waits for the entry of the corresponding digital information. The scalarizing problem constructed by the algorithm, as a result of which the solution at the current step is found, is visualized in the second text window. This information could be useful for people learning and working in MO area.
A window containing a text field is opened in the operating area with the help of “View ⇒ “Hint Window” commands (Fig. 3). When the cursor moves to any object of the operating windows of MOLIP system, an explanation text appears in this field. This is dynamic Help, which facilitates the operation with the system.

When interactive algorithm is used for MO problems solving, it is an advantage to present information not only about the last solution found, but also about the process of
search, about all the previous steps. Since some significant solutions are made on the basis of these results, it is important for the DM to be able to “testify” how he/she has reached this solution. That is why the information about the interactive process of MO problem regarded, which consists of the problem input data, the solutions obtained at each step, the preferences set by the DM for a new search and the constructed scalarizing problems, which are saved in *.mlp files associated with MOLIP system, serves not only for restarting an interrupted solution process, but also for documentation. “Print” command from the main menu can be used for selective print of information chosen by the DM.

CONCLUSIONS
The MDSS MOLIP is an experimental software system, developed at the Institute of Information Technologies of the Bulgarian Academy of Sciences. This system is designed for interactive solution of continuous problems of multicriteria optimization with different number of the criteria, of the variables and of the constraints. MOLIP system is developed on the basis of an innovative classification-based interactive algorithm and also on the basis of an original concept for user-friendly interface in such type of software systems.

REFERENCES

ABOUT THE AUTHOR
Assoc.Prof. Vasil Vassilev, PhD, E-mail: vasko@iinf.bas.bg
Krassimira Genova, E-mail: krasi@iinf.bas.bg
Mariyana Vassileva, PhD student, E-mail: mari@iinf.bas.bg
Filip Andonov, PhD student, E-mail: vonodna@yahoo.com,
Institute of Information Technologies, Bulgarian Academy of Sciences,
Phone: +359 2 870 04 29.