

A Prototype of Interactive Decision Support System with Automatic Prompter

Andrejs ZUJEVS

Latvia University of Agriculture, Liela 2, LV-3001, Jelgava, Latvia

andrejs.zujevs@gmail.com

Abstract. Engineering optimization problems mostly has more than one contradictory criteria. For solving such problems mathematically one can use multiple criteria decision making methods that provide finding compromise solution. This work contributes with nonlinear multiobjective optimization methods, where interactive methods comparing to others is more informative. Using the interactive methods contributes Interactive Decision Support Systems which provide user friendly interface and helps to make decisions. In the paper are described and approved opportunities then Interactive Decision Support System assist Decision Maker in preference information defining. This realized with an Automatic Prompter which determining preference information of the interactive method for the next iteration. The Automatic Prompter adaptable for the different interactive methods and has independence from strategy how to provide preference information. Using the Automatic Prompter the Decision Maker can verify whether it is possible to achieve a desired solution and what should be preference information. Determined preference information by the Automatic Prompter provides more information about a problem and the Decision Maker learning process is more effective. The Automatic Prompter was implemented and approbated as part of a prototype of the Interactive Decision Support System. Experiments with using the Automatic Prompter demonstrate that the Decision Maker has obtained more effective solutions.

Keywords. Interactive Decision Support System, multiobjective optimization, interactive methods, preference information determination, Decision Maker formal model

1. Introduction

Multiobjective optimization solves problems with two or more criteria which are optimized simultaneously. Usually the criteria are conflicting. To solve such problems multiple criteria decision making (MCDM) methods are used, but this paper focuses on the nonlinear interactive optimization methods (Miettinen, 1999). The multiobjective optimization problem (MOP) defined as:

$$\begin{aligned} \min f(x), \\ g(x) \leq 0, \\ h(x) = 0, \\ x: \mathcal{R}^n \in X, f(x): \mathcal{R}^k \in D, \end{aligned} \tag{1}$$

where $x \in \mathcal{R}^n$, $g(x) \in \mathcal{R}^m$, $h(x) \in \mathcal{R}^p$; m , p – inequality and equality count; n – variable count; k – criteria count; x – solution.

Objective vectors consist of objective function values $f(x) = (f_1(x), f_2(x), \dots, f_k(x))$ ^T and belongs to feasible objective region D. Decision (variable) vector is $x = (x_1, x_2, \dots, x_n)$ ^T and belongs to nonempty and compact feasible region $X \subset \mathbb{R}^n$.

In continuous problems an infinite number of solutions typically exist and preference information given by a human domain expert, called a Decision Maker (DM), used to direct the search toward the most preferred solution called final solution. Accordingly to Branke *et al.*, 2008 and Miettinen, 1999 the multiobjective optimization methods are divided into such classes: (i) non-preference methods without involving DM; (ii) apriori methods where DM gives her/his preference information before finding Pareto optimal solution; (iii) posteriori methods where a presentation of the set of Pareto optimal solutions is first generated and then DM chooses one of them; (iv) interactive methods where DM interacts with the interactive method, giving her/his preference information at each iteration until preferred final solution is obtained. The DM is participating in problem solving in an active way by using interactive methods. The DM solves the problem (1) by the interactive method in which she/he accumulate knowledge about the problem (Miettinen, Mäkelä, 2006) and learn about the criteria interaction (Xiao *et al.*, 2007). The role of the DM learning process is crucial for effective solutions obtaining (Miettinen, 1999, Belton *et al.*, 2008, Korhonen, Wallenius, 1996). Furthermore, the DM doesn't need apriori knowledge or information about the problem (Brandon *et al.*, 2009). Using interactive methods the DM can define her/his preferences more precisely and obtain more effective solutions.

A user-friendly interface provides more opportunities for the DM to collaborate with the interactive method. The information exchange between the DM and the interactive method using a user interface plays a crucial role in the decision making process (Tarkkanen *et al.*, 2013). Usually such opportunities are embedded in the Interactive Decision Support Systems (IDSS). There are a few publications about IDSS in multiobjective optimization for nonlinear problems. For example IDSS working with the implemented NIMBUS method is described in (Miettinen, Mäkelä, 2006) and whose commercial version IND-NIMBUS that integrates the Pareto Navigator method (Eskelinen *et al.*, 2010), and the NIMBUS and PAINT method (Hartikainen *et al.*, 2012). GANetXLS tool (Dragan *et al.*, 2011) based on Microsoft Excel and implemented for water management problem solving and may be modified to solve other model-driven problems. Obtained solutions visual representation also provide more information about problem and as result DM may obtain more effective final solution (Tarkkanen *et al.*, 2013, Miettinen, 1999).

The problem (1) solving using the interactive method usually occurs according to the interactive optimization procedure (IOP) (Miettinen, 1999). Solving optimization problem iteratively means that DM provides preference information for method during iterations (usually 3 – 8) and decide whether obtained solution is the final solution $f(x^*)$ or solving should be continued. The final solution choosing is subjective and depends on DM knowledge about problem (Miettinen, 1999). The problem solving starts with the initial solution $x^{(0)}$, then the DM specifies her/his preferences and method find the next Pareto optimal solution. Foremost preference information depends on the interactive method. For example, the GDF method (Geofrion *et al.*, 1972) uses marginal rates of criteria functions; the NIMBUS method (Miettinen, Mäkelä, 1997) uses criteria classification and a reference point. Depending on the DM's experience and situation, the DM may use different types of preference information; this is realized by using different interactive methods (Luque *et al.*, 2011).

Sometimes obtained solutions does not correspond to DM preferences. For example, DM observe that decisions and solutions are not consequent or convergence to the desired solution do not happen. In such cases it is hard for DM to choose final solution, and also to define preferences for the next iteration to continue solving the problem. Also sometimes DM has not experience with interactive method or preference information type is too difficult for him/her.

In the paper are described and approved opportunities then IDSS support DM in preference information defining. This is realized with the Automatic Prompter (AP) which determines preference information for the next iteration. It's lets DM better understand problem (learning is more efficient) and determine/produce the preference information for further iterations. DM trying to solve the problem (1) with minimal iterations number. If iteration number is large then DM get tired and decision making process become complicated to her/him (Korhonen, Wallenius, 1996). The AP determining preference information may take into account this observation. Furthermore the next iteration solution depends on the current solution $x^{(i)}$ and newly defined/provided preference information.

The second part of the paper describe concept of the AP. Also mathematical explanation of the AP is given, where the AP realized as a DM formal model. The third part of the paper describe experiments and results of validation of the DM formal model and using the AP in three testing optimization problem solving by using classical interactive methods: GDF (Geofrion *et al.*, 1972), STEM (Benayoun *et al.*, 1971) and GUESS (Buchanan, Corner, 1997) with different preference information type.

2. Concept

A problem solving process with the interactive method and involving a DM usually runs according to the IOP (Ravindra *et al.*, 2006). Problem solving starts with the initial solution $x^{(0)}$ (see Figure 1), if it is unknown to the DM, then IDSS obtains it using one of the special optimization methods. At the next step the DM provides her/his preferences in method preference information parameter's value way. Then IDSS transforms the preference information into a single optimization problem in conformity with the interactive method problem definition. At the next step optimization problem will be solved. An obtained solution will be presented to the DM in a graphical or numerical way for further interpretations and decisions. The DM analyses the obtained solution and decides to either continue to solve the problem and go to the next iteration or choose the final solution and stops optimization. If the DM decide continue, then new preferences are given and process repeats.

Unfortunately, problem solving with the interactive method usually relates to major difficulties for the DM: (i) how to transform the DM's preferences according to the method preference information type; (ii) how to interpret the obtained solution(s); (iii) how to get effective decisions faster and more focused; (iv) how to choose a more preferred solution. In practice where additional difficulties exist in problem solving using the interactive methods: (i) the opposite of the expectation is derived from the obtained solutions and decisions (no consequence between decisions and obtained solutions); (ii) no or a relatively small convergence to a desired solution is observed; (iii) DM doesn't have experience with the interactive method. Thus it is crucial to make decisions and choose a final solution.

The IOP was modified and that include the additional feedback from the IDSS to the DM. The feedback was implemented as the AP which determine preference information for the next iteration. In using the AP from DM necessary define desired solution called as goal solution which is a vector of desired criteria functions` values. Accordingly to it the AP determines preference information for the next iteration where more close solution to the desired will obtained.

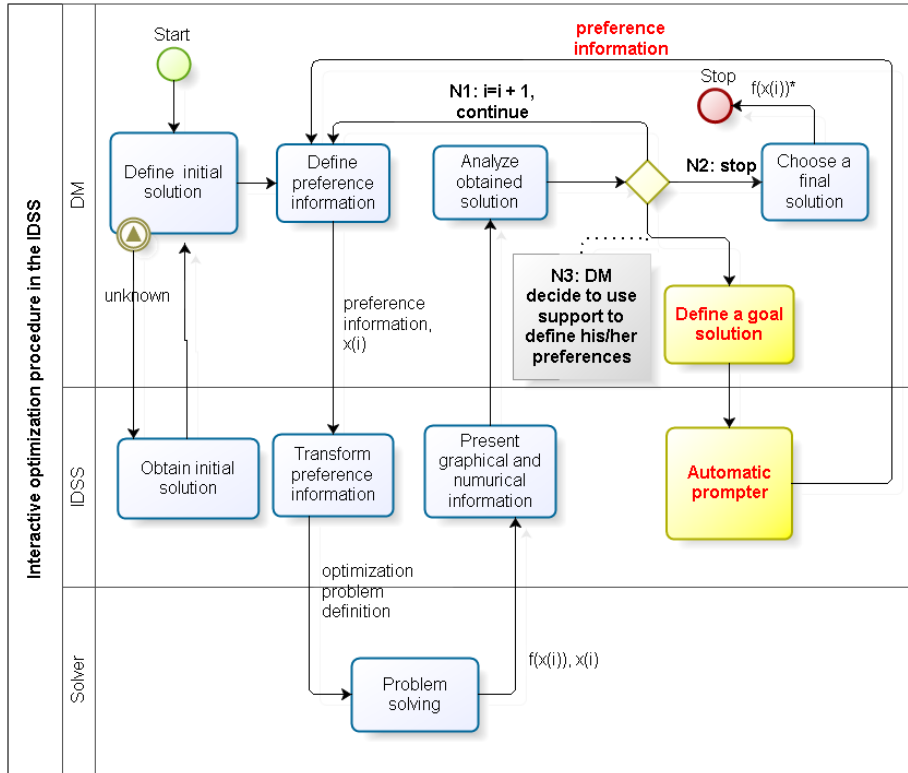


Fig. 1. Modified IOP. Diagram designed accordingly to BPMN2.0 (WEB, a) using BIZAGI modeler software (WEB, b). The diamond is the decision gateway block with outputs N1, N2 and N3. The outcome of N3 depicts by the AP determined preference information for the next iteration.

DM formal model (DMFM) called ZuMo (Zujevs, Eiduks, 2011) was adapted for the AP implementation. DMFM is defined as two criteria optimization problem (Zujevs, Eiduks, 2011). First criterion evaluates Euclidean distance from the obtained solution $\mu \in R^k$ to the goal solution $z^* \in R^k$ both are vectors in criteria values space. The solution μ will be obtained according to the generated preference information that is the solution of the problem (2). The second criterion assesses the preference information's correctness and conformity to the interactive method requirements. Definition of the adapted ZuMo is:

$$\min \left\{ p_1 \left(x^{(h)}, z^*, \varphi_p^{(h+1)} \right), p_2 \left(\varphi_p^{(h+1)} \right) \right\} \quad (2)$$

$$x^{(h)}: R^n \in S, z^*: R^k \in D, \varphi_p^{(h+1)} \in S_\varphi,$$

where $x^{(h)}$ – MOP current solution of iteration h in variable values space, z^* – the goal solution defined by DM in criteria values space, $\varphi_p^{(h+1)}$ – preference information for the next iteration $h+1$ (variable of problem (2)); S – feasible region (set) of original MOP in variables space, D – feasible region (set) of original MOP in criteria values space, P – original MOP which solves DM during IOP, S_φ – feasible region (set) of preference information accordingly to the interactive method.

As mentioned before the first criterion p_1 is Euclidean distance from the goal solution z^* to obtained μ accordingly to the $\varphi_p^{(h+1)}$ and current solution $x^{(h)}$. Second criterion p_2 is assessment function of correctness of $\varphi_p^{(h+1)}$ accordingly to the interactive method requirements. Inherently the second criterion is penalty function that value depends on $\varphi_p^{(h+1)}$ preference information contradiction.

The definition of the first criterion is:

$$p_1(x^{(h)}, z^*, \varphi_p^{(h+1)}) = \sqrt{\sum_{j=1}^k (z_j^* - \mu_j)^2}, \quad (3)$$

$$x^{(h)}: R^n \in S; u, z^*: R^k \in D,$$

where, k – criteria count of MOP, n – number of variables in original problem P , μ_j – value of j criterion which is obtained by solving P – original MOP with the interactive method accordingly to the current solution $x^{(h)}$ and generated preference information $\varphi_p^{(h+1)}$. The second criterion p_2 definition depends on interactive method and IDSS developer. Anyway the second criterion p_2 always equal to zero if there are no contradictions with interactive method preference information. Otherwise the value of criterion increase.

Solution of the problem (2) is the preference information $\varphi_p^{(h+1)}$ for the next iteration of IOP.

The paper's author suggests solving the problem (2) by using multiobjective evolutionary algorithms. Hence they are concerned with the global optimization and may find solutions for a difficult preference information structure. Preference information determining is difficult task for the classical search algorithms, because structure of the method's preference information may vary from simple (for example, criterion weight's values) to complicated (for example, criteria classification and reference point). The first and second criteria normalization of the problem (2) depends on evolutionary algorithm.

3. Experiments and results

For the experiments were selected three classical interactive methods: GDF (Geofrion *et al.*, 1972), STEM (Benayoun *et al.*, 1971) and GUESS (Buchanan, Corner, 1997).

All the methods have a different preference information type. The GDF method uses marginal rates of objective functions. The STEM method uses objective function classification: $I^<$ (criteria functions whose values will be improved) and $I^>$ (criteria functions whose values will be relaxed and $\varepsilon_i \geq f_i, f_i \in I^>$). The GUESS method uses ideal vector z^* , nadir vector z_i^{nad} for i -th criterion function and reference point vector \bar{z}^h for iteration h . Such interactive methods are chosen from the view of the preference

information difference. The DM using GDF method will be faultless accurate and usually the DM uses pre-calculation which is inconveniently.

Implemented prototype of IDSS with the proposed AP (see Figure 2) available in the internet (WEB, c). The prototype of IDSS provides work with previously mentioned classical interactive methods. In the prototype are implemented testing MOP with two and three criteria. Also the DM supported with different diagrams for the interpretation of the obtained solutions. The DM will use a progress diagram, broken line diagram, spider diagram, horizontal and vertical bar diagram, Pareto front diagram and Euclidean distance diagram.

Prototype of IDSS provides assessment for different metrics of Pareto front (Van Veldhuizen, 1999): Error Ratio, Generational Distance, Standart Deviation from the General Distance, Maximal Pareto Front Error, Overall Nondominated Vector Generation, Overall Nondominated Vector Generation Ratio, Progress metric, Hyper Plane metric, Hyper Plane Ratio, Zitler metric (Zitler, Thiele, 1998). For each testing MOP theoretical Pareto front was generated by using brute force technic. Pareto front metrics aimed to assess how the obtained solution(s) belongs to the theoretical Pareto front.

Experiments with the IDSS prototype were executed for the two aims: (i) validate the AP determining preference information which provide convergence to the goal (desired) solution; (ii) verify that the AP gives for the DM opportunity to get more effective solutions.

In the experiments for the first aim involved six testing optimization problems (see Table 1). Problems DOWNING and HANNE are solved with the GDF method, because they are corresponding to the requirements of the GDF method: the criteria functions are continuously differentiable and concave or continuously differentiable and linear. For the all problems initial and goal solution were predefined and they are Pareto optimal solutions.

Tab. 1. Solving testing optimization problems with the interactive methods

Interactive method			MOP	Type of Pareto front	Obj. count	Var. count
GDF	STEM	GUESS				
-	✓	✓	BINH1 (Binh, Ulrich, 1996)	Continuous and concave	2	2
-	✓	✓	FONSECA1 (Fonseca, Fleming, 1995)	Continuous and convex	2	3
-	✓	✓	TAPPETA (Tappeta <i>et al.</i> , 2000)	Asymmetric and continuous	3	3
-	✓	✓	VINNET (Vinnnet, 1996)	Symmetric and concave	3	2
✓	✓	✓	DOWNING (Downing, Ringuest, 1996)	Asymmetric and continuous	3	5
✓	✓	✓	HANNE (Collette, Siarry, 2003)	continuous and convex	2	2

The problem (2) was solved using evolutionary algorithm MOGA (Fonseca, Fleming, 1995) and NSGA-II (Deb *et al.*, 2002) with 50 and 100 generations and a fixed population size of 50 individuals. The MOGA algorithm's mutation rate was used 0.23 and *LambdaShare* 0.9. As low-level optimization method/algorithm used Matlab2008b (WEB, d) *Optimization toolbox* function *fmincon* which best fits to the interactive methods (GDF, STEM and GUESS) problem definition. Low-level optimization method solves single optimization problem which are result of multiobjective problem transformation by the interactive method. As *fmincon* integrates different algorithms, where only *Active-Set* (AS) and *Interior-Point* (IP) algorithm were used. Since they are provides using inequality constraints in optimization problem.

At each run of EA may produce a little bit different results. Since evolutionary algorithms have a stochastic type, then all the experiments were executed two times and the best result was selected using Euclidean Distance metric value. In total 112 experiments were executed: for each interactive method and problem eight experiments with MOGA and eight with NSGA-II were necessary.

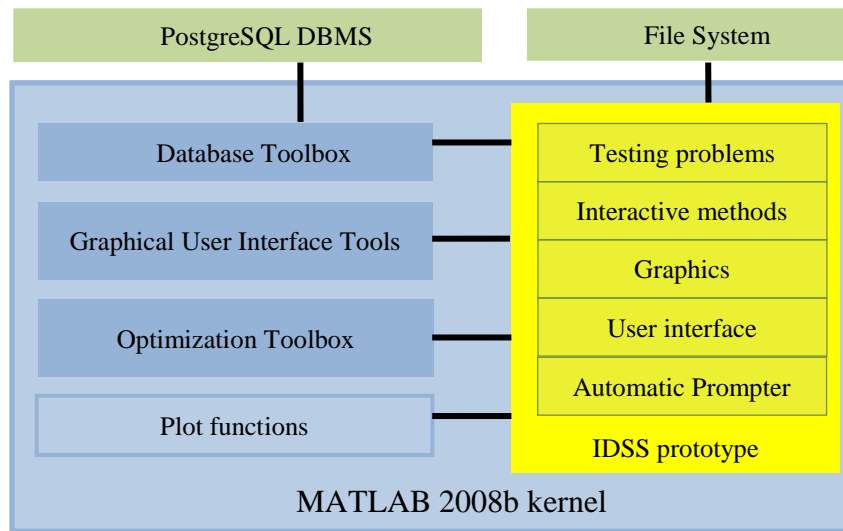


Fig. 2. Architecture of the implemented IDSS prototype with the AP. The implemented IDSS include twelve testing optimizations problems, three classical multiobjective interactive methods, and different graphics for results interpretations from the DM, also assessment of different Pareto front metrics available. With the yellow color marked author's work.

For experiments of the first aim different metrics also were assessed: Error Rate (ER), Maximal Pareto Front Error (MPE), Euclidean Distance (ED) and problem solving termination reason (PTR). ER and MPE metrics are used to assess how the obtained solution belongs to the theoretical Pareto front. ED assesses the distance from the initial solution to the goal solution in Euclidean space. PTR metrics assess the result of solving problem (2): the goal solution obtained (with defined accuracy); the goal solution not obtained (with defined accuracy) and if it belongs to the Pareto front; the goal solution not obtained and it does not belongs to the Pareto front; preference information not correctly generated.

After, all experiments the results were described using descriptive statistics. Variance and standard deviation was calculated for the ED and MPE metrics grouping data by PTR metric's values and the frequency was calculated for the PTR metric.

The experiment's results show that AP for the GDF, STEM and GUESS interactive method can determine preference information to obtain a goal solution, but not for all problems (see Figure 4). Therefore, the results depend on low-level method work, preference information type and problem as such. More precise results (by ED and MPE metric) were obtained by solving the problem (2) with NSGA-II and 100 generations.

For the four testing problems: HANNE, DOWNING, FONSECA1 and VINNET the AP determined preference information and a goal solution were obtained: for HANNE problem with the GDF and GUESS, for DOWNING only with GDF, for FONSECA1 only with GUESS, for VINNET with STEM and GUESS. For BINH1 and TAPPETA a goal solution was not obtained, but obtained solutions move near a goal solution, in the case with TAPPETA problem obtained solutions stay on hold with relatively small changes of initial Euclidean distance to a goal solution. Thus explained with TAPPETA problem Pareto set difficulty (in three dimensional space) and selected initial solution. For the BINH1 problem obtained solution converge to the middle point (middle point between the initial and goal solution) of Pareto front in concavity zone (see Figure 3). Thus explained with type of Pareto front concave or convex.

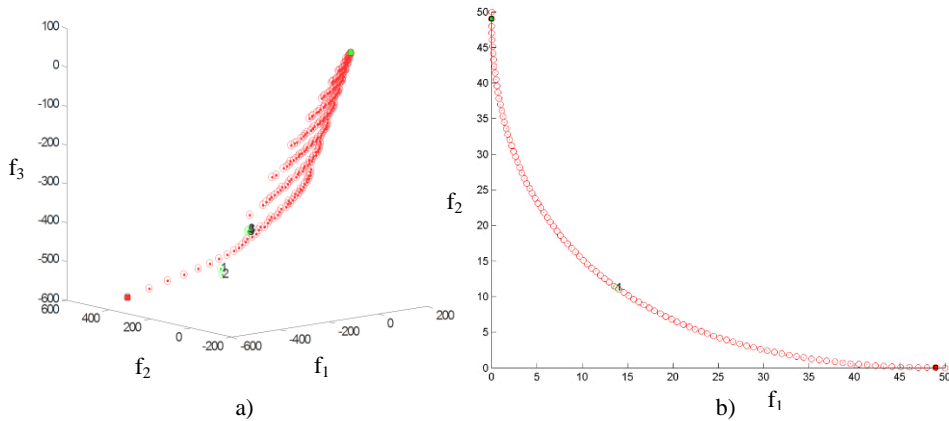


Fig. 3. Theoretical Pareto front with obtained solutions of TAPPETA (a) and BINH1 (b) problem. Where BINH1's theoretical Pareto front is concave and TAPPETA's theoretical Pareto front is asymmetric.

For the second aim the problem HANNE was solved with GDF, the problem TAPPETA with STEM method, and the problem BINH1 with GUESS method. In all experiments the DM used the AP.

Solving HANNE problem with GDF method the DM uses the AP in 5th iteration where obtained solutions move near the goal solution (not Pareto optimal) which the DM specifies in 4-th iteration is $z^* = \{2.0, 0.5\} \in D$ (see Figure 5). The reason of using support from IDSS is the DM observation that previously obtained solutions move away from the goal solution and the DM can't understand how to provide preference information for convergence to the desired solution. Solving TAPPETA problem with STEM method, the DM uses the AP in the 4th iteration where obtained solutions move

near the goal solution (not Pareto optimal) which the DM specifies after 3-th iteration is $z^* = \{-110, 70, -260\} \in D$ (see Figure 6). The reason for using the AP is divergence from the goal solution and the DM wants to improve the results.

Solving BINH1 problem with GUESS method, the DM wants to make sure whether it is possible to significantly improve the value of the criterion, however, the allowed, offered preference information only achieved a feasible solution (not Pareto optimal). This is explained by GUESS methods properties that it does not guarantee optimal Pareto solutions. In this case the DM makes sure it is possible to obtain another solution and what preference information corresponds to it.

The AP use has been justified in cases when the DM has observed obtained solution inconsistency or the DM wants to make sure whether it is possible to significantly improve the value of the criteria, as well as if the DM can't understand problems as such or has no experience with interactive method.

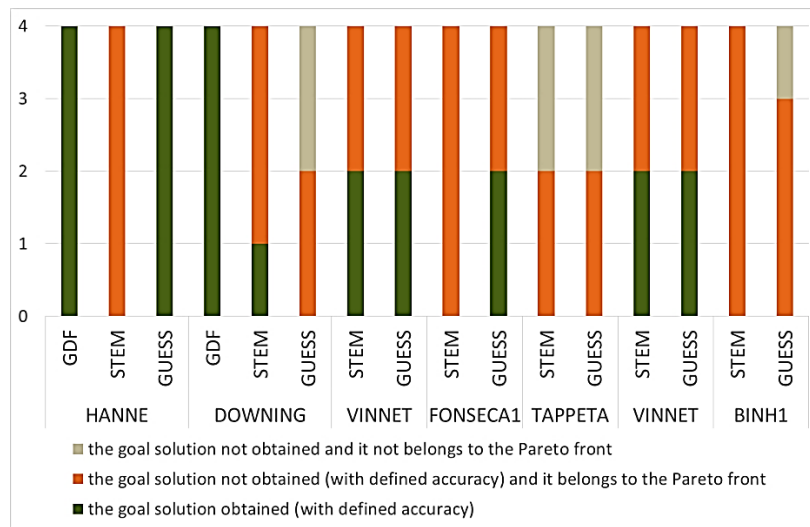


Fig. 4. Frequency of PTR metric. Obtaining a goal solution in a single iteration for the different testing problems and interactive methods: GDF, STEM and GUESS. In the vertical axis number is provided (selected by ED metric best value) of experiments runs executed with adapted ZuMo using NSGA-II and 100 generations.

4. Conclusions

First time defined and implemented IDSS with the AP which was realized as the DMFM, that the results foremost depend on evolutionary algorithm, generation count and from low level optimization method (solver).

The AP may be adapted to the interactive methods with a different preference type and that may be integrated in other IDSS. The AP is useful in cases when the DM observes an inconsistency of obtained solutions and decisions and no convergence to the desired solution. Using the AP the DM can verify whether it is possible to significantly improve the values of the criteria and by what preference information it is possible.

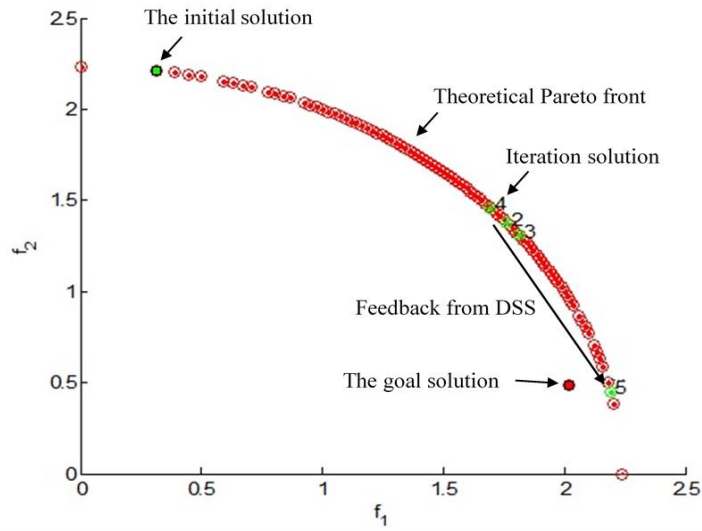


Fig. 5. DM solves HANNE problem with GDF method and uses the AP in 5th iteration.

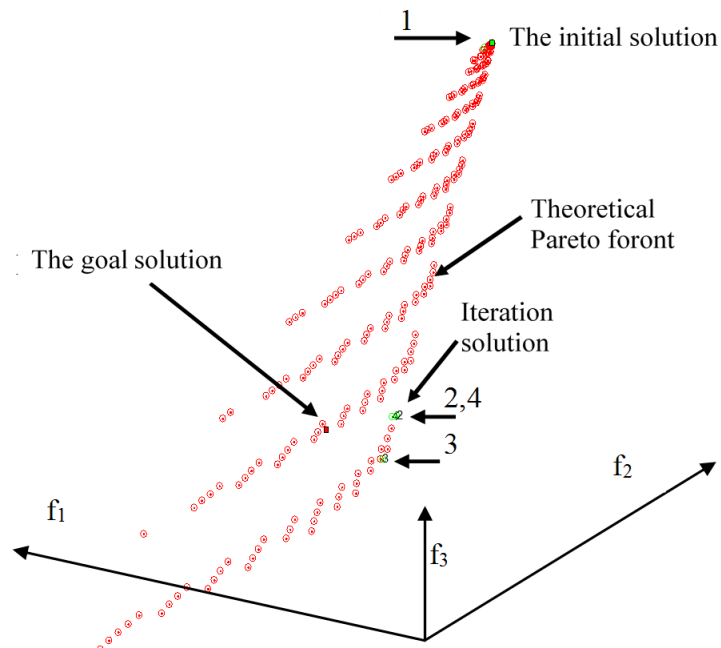


Fig. 6. DM solves TAPPETA problem with STEM method and uses the AP in 4th iteration.

The AP improves the DM learning process and as a result the DM may get a more effective solutions. The preference information provided from the AP can be used for DM learning process and for further decisions.

Optimization problem solving is more efficient if IDSS provides the DM with:

- a) different graphics for obtained solutions visual interpretation;
- b) the possibility to return to the previous iteration solution;
- c) the possibility to choose a low level optimization method (solver);
- d) different interactive methods;
- e) AP as a support from IDSS, which determines preference information for the next iteration.

Theoretically classical interactive methods can obtain a goal solution by a single iteration, but in practise the low level optimization method may not always converge to it.

Solving real-based problems the DMFM does not guaranty that a goal solution will be obtained, but a close solution can always be obtained. It is not mandatory that the goal solution will be a final solution of a problem.

References

- Benayoun, R., DeMontgolfier, J., Tergny, J., Laritchev O. (1971) Linear Programming with Multiple Objective Functions: STEP-method (STEM). *Mathematical Programming* 1, p. 366–375.
- Binh, T. T., Ulrich, K. (1996) An Evolutionary Strategy for the Multiobjective Optimization. *Proceedings of the Second International Conference on Genetic Algorithms*, p. 23-28.
- Brandon, Wong Y.H., Mariano, L., Yang, J. (2009) Using Interactive Multiobjective Methods to Solve DEA Problems with Value Judgements. *Computers & Operations Research* 36(2), p. 623-636.
- Branke, J., Deb, K., Miettinen, K., & Slowinski, R. (Eds.). (2008). *Multiobjective optimization: Interactive and evolutionary approaches*. Springer-Verlag, p 470.
- Buchanan, J.T., Corner, J.L. (1997) The Effects of Anchoring in Interactive MCDM Solution Methods. *Computers & Operations Research*, 24(10), p. 907-918.
- Collette, Y., Siarry, P. (2003) *Multiobjective Optimization Principles and Case Studies*. Springer, p. 293.
- Deb, K., Pratab, A., Agarwal, S., Meyarivan, T. (2002) A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II. Springer, p. 849–858.
- Downing, C.E., Ringuest, J.L. (1996) An Experimental Evaluation of the Efficacy of Four Multi-Objective Linear Programming Algorithms. *European Journal of Operational Research*, 104, p. 549-558.
- Dragan, A.S., Josef, B., Mark, S.M. (2011) A DSS Generator for Multiobjective Optimisation of Spreadsheet-Based Models. *Environmental Modelling & Software*, 26, p. 551-561.
- Eskelinen, P., Miettinen, K., Klamroth, K., Hakanen, J. (2010) Pareto Navigator for Interactive Nonlinear Multiobjective Optimization. *Operation Research Spectrum*, 32, p. 211–227.
- Fonseca, C. M., Fleming, P. J. (1995) Multiobjective Genetic Algorithm Made Easy: Selection, Sharing, and Mating Restriction. *Proceedings of the 1st International Conference on Genetic Algorithms in Engineering. Systems: Innovations and Applications, IET*, p.45-52.
- Geoffrion, A.M., Dyer, J.S., Feinberg, A. (1972) An Interactive Approach for Multicriterion Optimization, with an Application to the Operation of an Academic Department. *Management Science*, 19, p. 357-368.
- Hartikainen M., Miettinen K., Wiecek M., M. (2012) PAINT: Pareto Front Interpolation for Nonlinear Multiobjective Optimization. *Computational Optimization and Applications*, 52, p 845–867.
- Luque, M., Ruiz, F., & Miettinen, K. (2011). Global formulation for interactive multiobjective optimization. *OR Spectrum*, 33, p. 27–48.

- Miettinen, K.M. (1999) Nonlinear Multiobjective Optimization. Kluwer's International Series in Operational Research & Management Science, p. 298.
- Miettinen, K., Mäkelä, M. (1997) Interactive Method NIMBUS for Nondifferentiable Multiobjective Optimization Problems. Proceedings of The XI-th International Conference on Multiple Criteria Decision Making: Multicriteria Analysis, Springer-Verlag, Berlin-Heidelberg, p. 310 – 319.
- Miettinen, K., Mäkelä, M. (2006) Synchronous Approach in Interactive Multiobjective Optimization. European Journal of Operational Research, 170, p. 909 - 922.
- Ravindra, V., Tappeta J., Renaud E. (2006) Interactive Multiobjective Optimization Procedure. AIAA Journal, p. 1-15.
- Tappeta, R.V., Renaud, J.E., Messac, A. (2000) Interactive Physical Programming: Tradeoff Analysis and Decision Making in Multicriteria Optimization. Computational Control, AIAA Journal, 36, p. 219-226.
- Tarkkanen, S., Miettinen, K., Hakanen, J., Isomäki, H. (2013) Incremental User-Interface Development for Interactive Multiobjective Optimization. Expert Systems with Applications, 40, p. 3220–3232.
- Van Veldhuizen, D.A. (1999) Multiobjective Evolutionary Algorithms: Classification, Analyses and New Innovations. PhD thesis, Graduate School of Engineering, Air Force Institute of Technology, Wright Patterson AFB, Ohio, January 1999.
- Vinnet, R., Fonteix, C., Marc, I. (1996) Multicriteria Optimization Using a Genetic Algorithm for Determining a Pareto Set. International Journal of Systems Science, 27(2), p. 255-260.
- Xiao, N., Bennet, D.A., Armstrong, M.P. (2007) Interactive Evolutionary Approaches to Multiobjective Spatial Decision Making: A Synthetic Review. Computers, Environment and Urban Systems, 31, p. 232-252.
- Zitler, E., Thiele, L. (1998) Multiobjective Evolutionary Algorithms: A Comparative Case Study and the Strength Pareto Approach. Technical Report TIK, 43, Computer Engineering and Communication Network Lab, Swiss Federal Institute of Technology (ETH).
- Zujevs, A., Eiduks J. (2011) New Decision Maker Model for Multiobjective Optimization Interactive Methods, 17th International Conference on Information and Software Technologies, Kaunas, Lithuania.
- WEB (a) BPMN <http://www.bpmn.org>
- WEB (b) BIZAGI <http://www.bizagi.com>
- WEB (c) IDSS prototype source codes for Matlab:
<https://github.com/shoc2005/dssprototype>
- WEB (d) MATLAB <http://se.mathworks.com>

Received November 11, 2014, revised March 9, 2015, accepted March 9, 2015