Informatics in Education, 2011, Vol. 10, No. 2, 213–224

A Decision Support System for Solving Multiple Criteria Optimization Problems

Ernestas FILATOVAS, Olga KURASOVA
Vilnius University, Institute of Mathematics and Informatics
Akademijos 4, LT-08663 Vilnius, Lihuania
e-mail: ernest.filatov@gmail.com, olga.kurasova@mii.vu.lt

Received: July 2011

Abstract. In this paper, multiple criteria optimization has been investigated. A new decision support system (DSS) has been developed for interactive solving of multiple criteria optimization problems (MOPs). The weighted-sum (WS) approach is implemented to solve the MOPs. The MOPs are solved by selecting different weight coefficient values for the criteria and solving single-criterion optimization problems (SOPs). However, the WS approach does not guarantee either finding evenly distributed solutions of a Pareto front, or finding Pareto optimal solutions in non-convex regions. In order to solve the problems, some methods have been proposed. In this paper, an adaptive weighted-sum (AWS) approach has been used. We suggest presenting the solutions, obtained not only by the weighted-sum method, but also by the adaptive weighted-sum approach, to a decision maker (DM). Thus the decision making will be facilitated.

Keywords: multiple criteria optimization, decision support system, interactive optimization, decision making, Pareto front.

Introduction

When solving the applied tasks in various areas (production, planning, chemistry, education, etc.), we face multiple criteria optimization problems. Often the criteria are contradictory. It is impossible to improve the value of a criterion without deterioration of the values of the other criteria. A lot of methods have been proposed for solving multiple criteria optimization problems (Veldhuizen, 1999; Miettinen, 1999; Ehrgott, 2005; Figueira *et al.*, 2005). The methods can be grouped into non-interactive (generating) and interactive ones. The methods of the first group find the whole set or a subset of non-dominated solutions. In the interactive methods, a decision maker is involved not only in decision making, but also in the solving process (Alves and Climaco, 2006). It is necessary to develop DSSs, which helps a DM to make a decision.

It is often faced with various multiple criteria optimization problems in education: score calculation in informatics contests (Skupiene, 2011), evaluation of academic staff (Salmuni *et al.*, 2006), school scheduling (Fügenschuh and Martin, 2006; Pupeikienė *et al.*, 2009). Solving these problems a suitable DSS is useful for a DM to solve problems more efficiently.

In the paper, we introduce a new decision support system for solving multiple criteria optimization problems in an interactive way. The system is implemented in Matlab. The opportunities of optimization and parallel computing toolboxes are used for solving the optimization problem and implementing a parallel algorithm. The wide used method, the weighted-sum approach, is implemented in the system. In order to improve the weighted-sum method, an adaptive weighted-sum approach is implemented, too. The method helps a DM to find better solutions of the MOPs.

Multiple criteria optimization

The general form of a multiple criteria optimization (minimization) problem (Zionts, 1989) is:

$$\min_{X=(x_1,\dots,x_n)\in D} F(X) = \left[f_1(X), f_2(X), \dots, f_m(X)\right]^T, \tag{1}$$

where D is a bounded domain in the n-dimensional Euclidean space R^n, $X = (x_1, x_2, \dots, x_n)$ is a vector of variables, the functions $f_j(X)$: $R^n \to R^1$ are criteria and m is the number of criteria.

One of the possible ways of solving the system of problems (1) is to form a single-criterion problem. Commonly the weighted-sum approach is used (Zadeh, 1963) for this purpose. This method was proposed almost 50 years ago, but it still remains the basic and most popular method for solving MOPs. The reasons are that it is effective in most cases of solving MOPs and is simple to implement. The idea of this method is presented below. All the criteria multiplied by the positive weight coefficients $\omega_j,\ j = \overline{1,m}$, are summed up and the following single-criteria minimization problem is solved:

$$\min_{X=(x_1,\dots,x_n)\in D} \sum_{j=1}^{m} \omega_j f_j(X), \tag{2}$$

where $\sum_{j=1}^{m} \omega_j = 1$. The solving process of problem (2) is reiterated by selecting different combinations of the weight coefficient values $\omega_j,\ j = \overline{1,m}$. The values can be selected randomly, by some rule or by a DM. Many solutions are obtained and they are Pareto points. The most acceptable ones are selected by a DM.

The disadvantage of the WS approach is that it does not guarantee finding evenly distributed solutions of a Pareto front. Solutions often appear only in some parts of the Pareto front, while no solutions are obtained in the other parts. The approach does not find Pareto optimal solutions in non-convex regions. In order to avoid these demerits, various methods have been proposed. The most popular ones are described here.

The normal boundary intersection (NBI) method uses a scalarization algorithm which allows a uniform spread of solutions on the Pareto front (Das and Dennis, 1998). Also, the method is independent of the relative scales of different criteria. The NBI method discovers solutions in the non-convex region while they cannot be obtained by the WS approach. Disadvantages of the NBI method are that sometimes non Pareto-optimal solutions are obtained and this method works properly only with two-criteria problems. If more than two-criteria problems are solved, some Pareto front regions are not found.

The normal constraint (NC) method is an improvement over the NBI method (Messac *et al.*, 2003; Messac and Mattson, 2004). Disadvantages of the NC method are that dominated solutions are found, and a Pareto filter is needed to remove the dominated solutions.

The adaptive weighted-sum method is similar to the WS approach (Kim and de Weck, 2005). The AWS method searches the points of the Pareto front in the regions where the Pareto front points have not been discovered by the WS method, by introducing some additional constraints to these regions. The advantages of this method are that it produces well-distributed solutions, finds Pareto optimal solutions in non-convex regions. In the two-criteria case, the Pareto filter is not necessary, because the dominated solutions are rejected automatically.

The AWS approach consists of some procedures:

1. At first, problem (2) is solved by the WS method.
2. Then, intervals between Pareto points are identified, in which the Pareto points have not been found and the lengths of the intervals are not so small.
3. The Pareto front points are searched in the identified intervals: problem (2) is solved by the WS approach with the additional constraints introduced, which restrict a feasible region in each identified interval.

2–3 steps are repeated while the Pareto points are evenly distributed on the Pareto front in feasible regions and non-feasible regions are found.

The above procedures are applied, if a two-criteria optimization problem is solved ($m = 2$). In the paper (Kim and de Weck, 2006), the method was adapted for solving a multiple criteria problem, if $m > 2$. As examples, the three-criteria cases were used there. In the case where $m = 3$, the above-mentioned intervals must be changed to patches and the length of the interval to the area of patches. If $m > 3$, the implementation of the

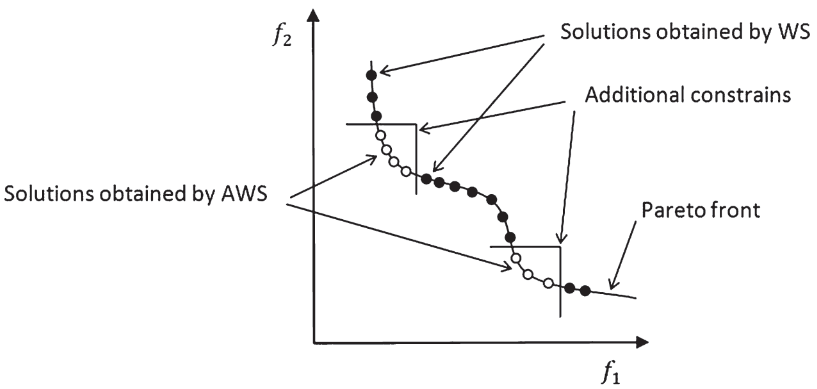


Fig. 1. The solutions obtained by the WS and AWS methods (in case of two-criteria).

method is very complicated. In the three or more criteria cases, the Pareto filter is required.

In this paper, a multiple criteria problem of selecting the optimal nutritive value is investigated (Petkus and Filatovas, 2008, 2009). The DSS is adjusted for solving this problem. The objective of the problem is to minimize farmers' expenditure on nutrition by the optimal selection of feed ingredients in cattle-breeding. It is taken into consideration that cattle diets consist of different ingredients (e.g., maize, corn, peas, fish oil, etc.) on the one hand, and each ingredient differs by different nutritive characteristics (e.g., protein, calcium, natrium, etc.), on the other hand. The cost price as well as violations of the requirements of the nutritive characteristics values must be minimized. So, the multiple criteria problem must be solved.

The feed cost price f_1, as one of the criteria, is calculated by the formula:

$$f_1(x_1,\dots,x_n)=\sum_{i=1}^{n} x_i k_i. \tag{3}$$

Here x_i is a constituent part of the ith ingredient in feed; k_i is the price of the ith ingredient for a weight unit; n is the number of ingredients.

Other criteria are violations of the requirements. They are calculated by the following formula:

$$f_2(x_1,\dots,x_n)=\begin{cases} 0, & \text{if } R^j_{\min}\leqslant R^j\leqslant R^j_{\max},\\ R^j-R^j_{\max}, & \text{if } R^j-R^j_{\max}>0,\ j=\overline{2,m},\\ R^j_{\min}-R^j, & \text{if } R^j_{\min}-R^j>0, \end{cases} \tag{4}$$

where $\sum_{i=1}^{n} x_i=1;\ x_i^{\min}\leqslant x_i\leqslant x_i^{\max},\ i=\overline{1,n}; R^j=\sum_{i=1}^{n} x_i A_{ij}(x_1,\dots,x_n)$. $R^j_{\min}$ and $R^j_{\max}$ are the recommended permissible minimal and maximal amount of the jth nutritive characteristics in feed; A_{ij} is a nonlinear function that expresses the value of the jth nutritive characteristics of the ith ingredient; is the number of nutritive characteristics in feed; $x_i^{\min}$ and $x_i^{\max}$ are the minimal and maximal value of the constituent part of the ith ingredient. In further experimental investigations, the values $n=50,\ m=15$ are used.

Criterion (3) is contradictory to the group of criteria (4). With an increase in violations of the permissible amount of nutritive characteristics, the price of feed is falling. Furthermore, it is necessary to obtain the solution that each violation does not exceed the requirements, denoted by DM. These requirements can differ from the values $R^j_{\min}$ and $R^j_{\max}$.

When the AWS approach is applied, the problem of 15 criteria is aggregated to two-criteria as follows: f_1 is the cost price, $f_2'=\sum_{j=2}^{15} f_j$, i.e., the sum of violations of the requirements. The following minimization problem is solved:

$$\min_{X=(x_1,\dots,x_n)\in D}\bigl(\omega_1 f_1(X)+\omega_2 f_2'(X)\bigr), \tag{5}$$

where $\omega_1+\omega_2=1$.

Decision Support System

The DM's involvement is necessary in solving multiple criteria optimization problems interactively (Huang *et al.*, 2005; Miettinen and Mäkelä, 2006; Klamroth and Miettinen, 2008). Thus, a decision support system must be designed with a graphical user's interface that facilitates DM's work and permits to review the results and to plan the process of solving. When solving MOPs, the graphical representation plays an important role for decision making (Vassilev *et al.*, 2006; Blasco *et al.*, 2008; Ginevičius and Podvezko, 2008; Dowhań *et al.*, 2009).

A DSS for solving MOPs is developed in Matlab. The opportunity for solving problems in a parallel way is implemented in the DSS. A DM can choose the number of processors for solving the problem according to the complexity of the problem and the available computational resources.

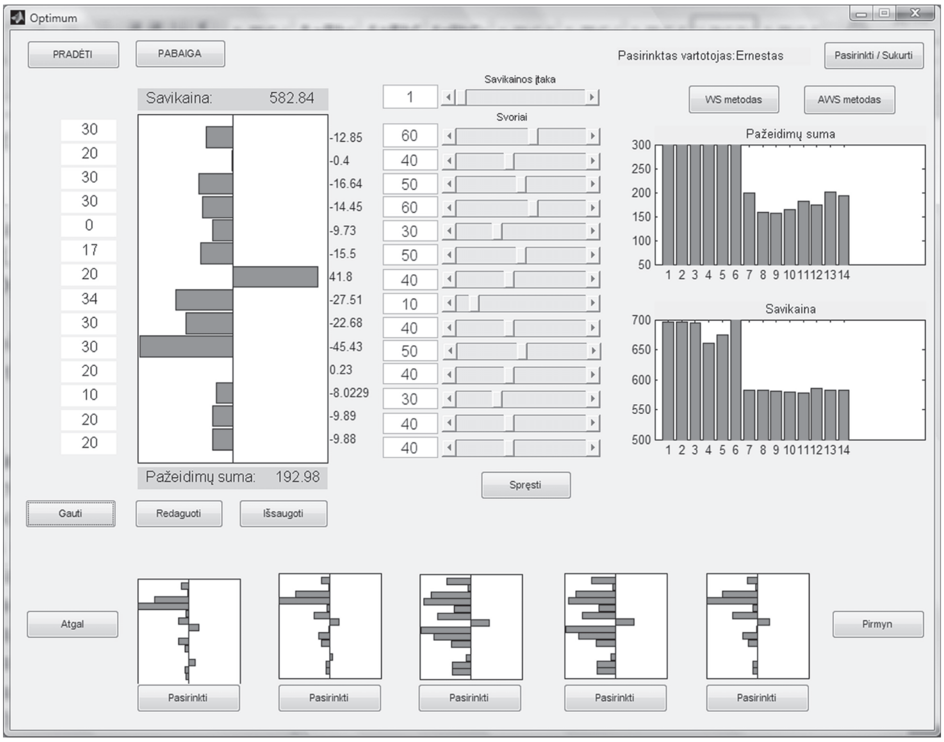
In this investigation, in order to demonstrate the DSS features, the system has been adapted to solve a concrete problem – the problem of selecting the optimal nutritive value (the cost price is one of the criteria (3), the other 14 criteria are violations of permissible minimal and maximal norms (4)). The aim of the problem is to achieve the solution with minimal violations of the recommended permissible minimal and maximal amounts of the nutritive characteristics in feed at a lower price. The DSS has been slightly changed according to the specificity of the problem.

The graphic interface of DSS is presented in Fig. 2. The top right corner of the window displays the last obtained or edited solution. Fourteen horizontal bars present deviations from the norm of values of the corresponding nutritive characteristics, i.e., violations of the requirements, when the values $\omega_j,\ j = \overline{2,15}$, of the weight coefficients, presented on the left, are selected. The numerical values of violations are presented on the right. On the top of fourteen horizontal bars, the value of the cost price f_1 (*Savikaina*) is located. At the bottom, the sum f_2' of violations of the requirements (*Pažeidimų suma*) is presented.

A block in the middle of the DSS is designed for changing the weight coefficients $\omega_j,\ j = \overline{2,15}$. A DM can change the values ω_j, typing the desirable values into the textboxes, or using the scroll bars. On the right of the window, the column chart of the sums of violations and the values of cost prices, obtained in each experiment, are presented in order to observe the solving process. The bottom of the window presents the solutions (5 blocks) that have been obtained and memorized up to the moment. The blocks display only five memorized solutions, nevertheless, it is possible to review and use any other memorized solution for further editing with the help of buttons *Pirmyn* and *Atgal*. By using the buttons *Pasirinkti* under each block, a DM can select one of the memorized solutions for editing. A possibility has been provided to delete any of the memorized solutions that is improper and therefore needless of editing, as far as the DM is concerned.

Moreover, some different users (DMs) can solve the same problem independently. Their solutions obtained are saved separately and later a comparative analysis can be made and the best result achieved can be selected.

The general schema of solving the multiple criteria optimization problem using the DSS is described as follows:



Pasirinktas vartotojas:Ernestas

Savikaina: 582.84

30	-12.85	60
20	-0.4	40
30	-16.64	50
30	-14.45	60
0	-9.73	30
17	-15.5	50
20	41.8	40
34	-27.51	10
30	-22.68	40
30	-45.43	50
20	0.23	40
10	-8.0229	30
20	-9.89	40
20	-9.88	40

Savikainos įtaka: 1

Pažeidimų suma: 192.98

Fig. 2. Graphic interface of the decision support system.

1. The optimization problem of 15 criteria (cost price and 14 violations of the requirements) is solved by the WS method. A DM starts solving the problem with the initial data (combinations of coefficient values $\omega_j,\ j=\overline{1,15}$, selected by the DM). The solving process is continued by reiterating different combinations of coefficient values ω_j interactively and the optimization problem is aggregated to a single-criterion optimization problem according to the WS approach. The problem is solved several times and the DM chooses some acceptable solutions.
2. If the DM fails to obtain acceptable solutions by the WS method, then the two-criteria problem (5) can be solved by the AWS method. All the solutions obtained by the AWS method are saved.
3. The DM can select acceptable solutions, obtained by the AWS method. The cost prices and all 14 violations of the requirements of the solutions are computed. The graphical view of violations is shown for the DM. If the solution is not acceptable, the DM can try to improve the solution by selecting other coefficient values ω_j.
4. The DM stops the solving process when the preferable solutions are found.

The Matlab function *fmincon* is used for solving a single-criterion optimization problem. This function includes some optimization algorithms: Trust-Region-Reflective Optimization (Coleman and Li 1994, 1996), Active-Set Optimization (Han 1977; Powell 1978; Gill 1981), Interior-Point Optimization (Byrd *et al.*, 1999; Waltz, 2006), SQP Op-

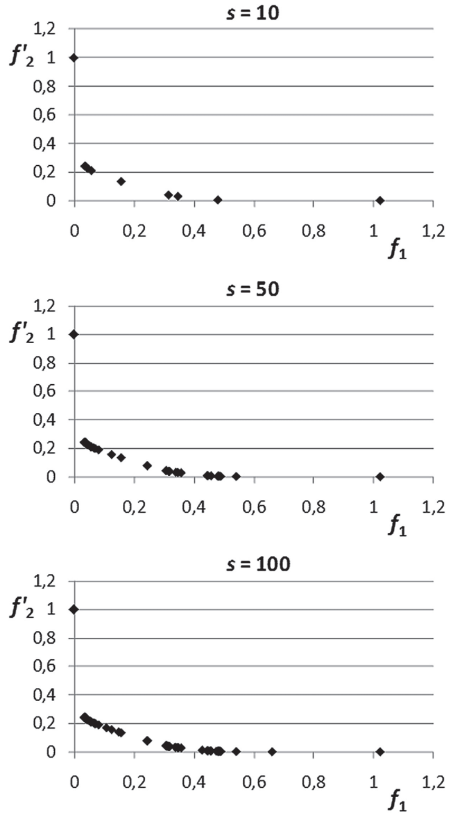


Fig. 3. Solutions obtained by the WS method with various values s.

timization (Nocedal and Wright, 2006). The function *fmincon* selects the most suitable method or a combination of methods for solving a problem. When solving the problem, the function *fmincon* finds a minimizer X of the function starting at X_0. Starting from different points X_0, different results can be obtained.

Experimental Investigations

The goal of experimental investigations is to show the necessity to apply the AWS method to distribute the solutions on the Pareto front evenly in feasible regions. As mentioned before, implementation of the AWS method is very complicated in the case of more than 3 criteria, so two-criteria optimization problem (5) is solved in this investigation. The first criterion f_1 is the cost price; the second criterion f'_2 is the sum of violations of the requirements. The values of criteria are normalized in the interval [0, 1].

At first, we investigate the distribution of solutions on the Pareto front when solving problem (5) by the WS method and the values of X_0 are fixed. The values of the weight coefficient ω_1 are selected from 0 to 1 at different steps s, and the values of the weight co-

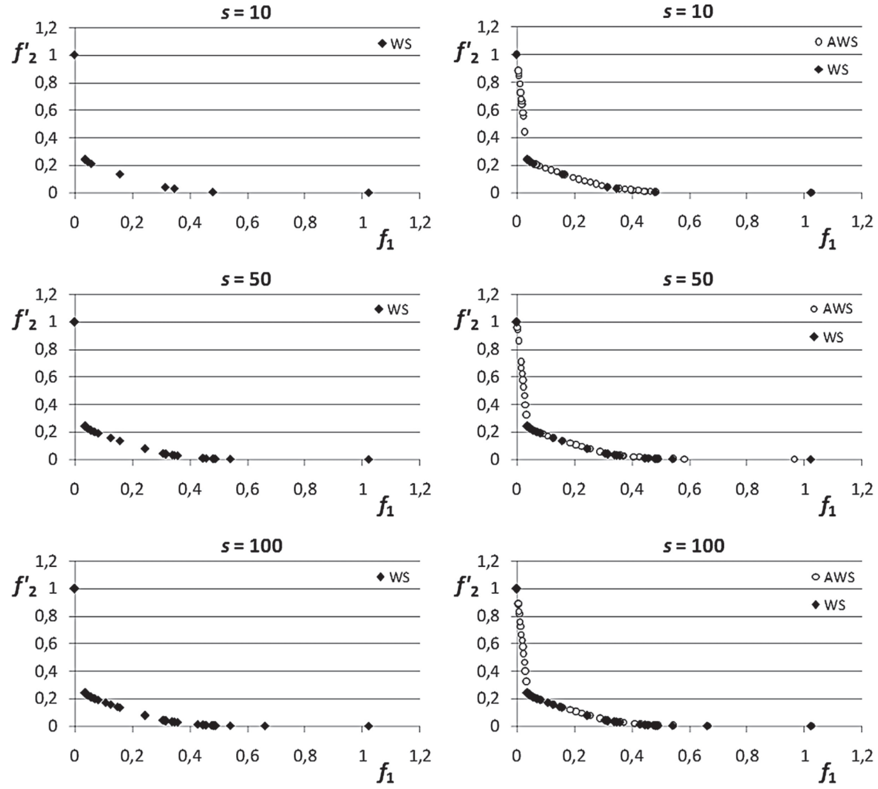


Fig. 4. Solutions obtained by the WS and AWS methods with various values s.

efficient ω_2 are calculated by formula $\omega_2 = 1 - \omega_1$. Three different steps $s = 10, 50, 100$ are selected and the obtained solutions are presented in Fig. 3. We see that the solution points of the Pareto front are not distributed evenly. With an increase of the step value s the number of solutions increases too, but some solutions are coincident and they are crowded together in some groups. No solutions are obtained in some parts of the Pareto front.

In the second investigation, we compare distribution of solutions on the Pareto front obtained by the WS and AWS methods. As mentioned before, the AWS method uses the results obtained by the WS method. The AWS method finds the solutions in those parts of the Pareto front where the WS method was unable to find them. The solutions obtained in the first investigation (Fig. 3) were used in the AWS method. The obtained results are presented in Fig. 4. We see that the solutions by the AWS method fill in the parts of the Pareto front which were not filled in by the WS method. However, some parts are not filled in because no feasible points were found here. It depends on the specificity of problem (5). Moreover, the experimental results show that a small step value s is enough for the AWS method to obtain noncoincident solution points and even distribution of the Pareto front. So, in further experiments, the step value $s = 10$ is used.

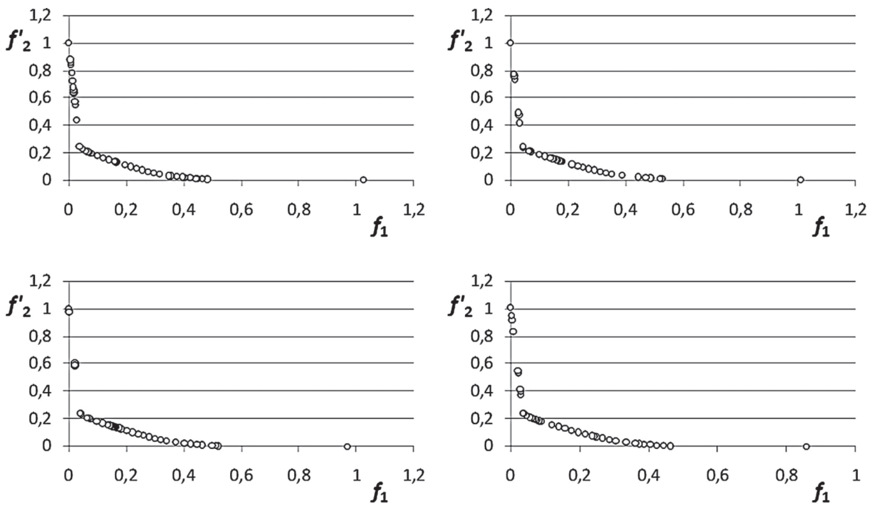


Fig. 5. Solutions obtained by the AWS method with various values X_0.

In the next investigation, we analyze the dependences of distributions of the obtained solutions on different starting points X_0. Many sets of the starting points X_0 are used in the experiments, but only four most distinguished results are presented in Fig. 5. We see that distributions of the points on the Pareto front differ slightly, but not essentially. According to the experimental results, most of the solutions are obtained in one part of the Pareto front: $f_1 \in (0.05,\ 0.55)$ and $f_2' \in (0.004,\ 0.23)$. The values of the criterion f_1 vary in a much wider interval than the values of the criterion f_2'. Thus, with a slight decrease of the sum of violations (f_2'), the cost price (f_1) increases significantly. The solutions are slightly dependent on X_0, but selection of proper X_0 is not the essential task in solving problem (5).

Conclusions

In the paper, a new decision support system is introduced and investigated. Not only the weighted-sum approach, but also the adaptive weighted-sum method is implemented in this system. The DSS is developed with a view to facilitate decision making. It is able to solve large-scale problems, because distributed computing is implemented. The experimental investigations show that even a large set of weight coefficients does not guarantee finding evenly distributed solutions of the Pareto front when the problem is solved by the WS method, because many solution points are coincident or close to each other. The AWS method finds the solution points distributed on the Pareto front evenly. A DM can use the obtained solutions in order to find the preferable solution. Moreover, the solutions obtained by the AWS method slightly depend on the starting point for optimization, so its selection is not a primary task in solving MOPs.

In this investigation, the DSS has been adapted to solve a problem of selecting the optimal nutritive value. However, the DSS can be easily modified for solving other similar multiple criteria optimization problems, for example, school scheduling problem. There are no schedules that satisfy all restrictions and personal preferences. So, the DSS can help to compare some alternative schedules interactively – violations of the restrictions should be visualized. In this case, it would be easier to a DM to choose the preferable solution.

References

Alves, M.J., Clımaco, J. (2007). A review of interactive methods for multiobjective integerand mixed-integer programming. *European Journal of Operational Research*, 180(1), 99–115.

Byrd, R.H., Mary E. Hribar, Nocedal, J. (1999). An interior point algorithm for large-scale nonlinear programming. *SIAM Journal on Optimization*, 9(4), 877–900.

Coleman, T.F., Li, Y. (1994). On the convergence of reflective Newton methods for large-scale nonlinear minimization subject to bounds. *Mathematical Programming*, 67(2), 189–224.

Coleman, T.F., Li, Y. (1996). An interior, trust region approach for nonlinear minimization subject to bounds. *SIAM Journal on Optimization*, 6(2), 418–445.

Das, I., Dennis, J. E. (1998). Normal-boundary intersection: a new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM Journal on Optimization*, 8(3), 631–657.

Van Veldhuizen, D. A. (1999). Multiobjective evolutionary algorithms: classifications, analyses, and new innovations. *PhD Thesis*, Air Force Institute of Technology, Ohio.

Dowhań, L., Wymysłowski, A., Dudek, R. (2009). Multi-objective decision support system in numerical reliability optimization of modern electronic packaging. *Microsystem Technologies*, 15(12), 1777–1783.

Ehrgott, M. (2005). Multicriteria optimization. *Lecture Notes in Economics and Mathematical Systems*, 491, Springer-Verlag, Berlin.

Figueira, J., Greco, S., Ehrgott, M. (2005). *Multiple Criteria Decision Analysis: State of the Art Surveys*. Springer-Verlag, Berlin.

Fügenschuh, A., Martin, A. (2006). A multicriteria approach for optimizing bus schedules and school starting times. *Annals Of Operations Research*, 147(1), 199–216.

Gill, P.E., Murray, W., Wright, M.H. (1981). *Practical Optimization*. Academic Press, London.

Han, S.P. (1977). A globally convergent method for nonlinear programming. *Journal of Optimization Theory and Applications*, 22(3), 297–309.

Kim, I.Y., de Weck, O.L. (2005). Adaptive weighted-sum method for bi-objective optimization: Pareto front generation. *Structural and Multidisciplinary Optimization*, 29(2), 149–158.

Kim, I.Y., de Weck, O.L. (2005). Adaptive weighted sum method for multiobjective optimization: a new method for Pareto front generation. *Structural and Multidisciplinary Optimization,* 31(2), 105–116.

Messac, A., Ismail-Yahaya, A., Mattson, C.A. (2003). The normalized normal constraint method for generating the Pareto frontier. *Structural and Multidisciplinary Optimization*, 25(2), 86–98.

Messac, A., Mattson, C.A. (2004). Normal constraint method with guarantee of even representation of complete Pareto frontier. *AIAA Journal*, 42(10), 2101–2111.

Miettinen, K. (1999). Nonlinear multiobjective optimization. *International Series in Operations Research and Management Science*, 12, Kluwer Academic Publishers, Dordrecht.

Nocedal, J. Wright, S. J. (2006). *Numerical Optimization*. Springer Series in Operations Research, Springer-Verlag, Berlin.

Petkus, T., Filatovas, E. (2008). Decision making to solve multiple criteria optimization problems in computer networks. *Information Technology and Control*, 37(1), 63–68.

Petkus, T., Filatovas, E., Kurasova, O. (2009). Investigation of human factors while solving multiple criteria optimization problems in computer network. *Technological and Economic Development of Economy*, 15(3), 464–479.

Powell, M.J.D. (1978). A fast algorithm for nonlinearly constrained optimization calculations. In: Watson, G.A. (Ed.), *Numerical Analysis, Lecture Notes in Mathematics*, 630, 144–157.

Pupeikienė, L., Strukov, D., Bivainis, V., Mockus, J. (2009). Optimization algorithms in school scheduling programs: study, analysis and results. *Informatics in Education*, 8(1), 69–84.

Salmuni, W., Mustaffa, W., Shokory, S.M., Kamis, H. (2006). The analytical hierarchy process: multi criteria decision making for promoting academic staff in higher education. *The Journal of Global Business Management*, 2(1).
`http://www.jgbm.org/page/16%20Wan%20Salmuni%20Wan%20Mustaffa.pdf`.

Skūpienė, J. (2011). Score calculation in informatics contests using multiple criteria decision methods. *Informatics In Education*, 10(1), 89–103.

Vassilev, V., Genova, K., Vassileva, M. (2006). A multicriteria decision support system multidecision-1. *Information Theories & Applications*, 13, 203–209.

Waltz, R.A., Morales, J.L., Nocedal, J., Orban, D. (2006). An interior algorithm for nonlinear optimization that combines line search and trust region steps. *Mathematical Programming*, 107(3), 391–408.

Zadeh, L. (1963). Optimality and non-scalar-valued performance criteria. *IEEE Transactions on Automatic Control*, 8(1), 59–60.

Zionts, S. (1989). Multiple criteria mathematical programming: an updated overview and several approaches. In: Karpak, B., Zionts, S. (Eds.), *Multiple Criteria Decision Making and Risk Analysis Using Microcomputers*, 56, 7–60, Springer-Verlag, New York.

E. Filatovas is a PhD student at Institute of Mathematics and Informatics of Vilnius University. He received bachelor's degree in mathematics (2004) and master's degree in informatics (2006) from Vilnius Pedagogical University. His research interests include multi-criteria optimization, multi-criteria decision support, parallel computing, learning technologies.

O. Kurasova is a senior researcher at Institute of Mathematics and Informatics of Vilnius University as well as an associate professor at Information Technologies Department of Vilnius Pedagogical University. She received bachelor's degree in mathematics (1999) and master's degree in informatics (2001) from Vilnius Pedagogical University, PhD in informatics from Vytautas Magnus University and Institute of Mathematics and Informatics (2005). Her research interests include data mining methods, artificial neural networks, multidimensional data visualization, dimensionality reduction, multi-criteria optimization.

Sprendimų paramos sistema daugiakriteriniams optimizavimo uždaviniams

Ernestas FILATOVAS, Olga KURASOVA

Šiame straipsnyje nagrinėjama nauja sprendimų paramos sistema, skirta daugiakriteriniams optimizavimo uždaviniams spręsti interaktyviai. Sistemoje realizuotas svorinės sumos metodas, kuriuo daugiakriteriniai optimizavimo uždaviniai spendžiami parenkant skirtingas svorių kombinacijas ir taip suvedant uždavinį į vienakriterinį. Tačiau svorinės sumos metodas neužtikrina tolygaus sprendinių pasiskirstymo ant Pareto paviršiaus ir neranda sprendinių tose vietose, kuriose tikslo funkcija yra neiškyli. Šiems trukumams pašalinti yra siūlomi įvairūs metodai. Vienas iš jų, adaptyvus svorinės sumos metodas, yra realizuotas sukurtoje sprendimų paramos sistemoje ir analizuojamas šiame darbe. Siekiant palengvinti sprendimų priėmimą sprendžiant daugiakriterinius optimizavimo uždavinius, siūloma sprendimų priėmėjui pateikti rezultatus, gautus uždavinį sprendžiant ne tik svorinės sumos metodu, bet ir adaptyviu svorinės sumos metodu. Sprendimų priėmėjas iš gautų sprendinių gali pasirinkti jam priimtiniausius. Siūloma sprendimų priėmimo sistema gali būti adaptuota įvairių daugiakriterinių optimizavimo uždavinių sprendimui, pavyzdžiui, mokyklos tvarkaraščių sudarymui. Sistema, kurioje būtų vizualizuoti apribojimų pažeidimai, padėtų palyginti alternatyvius tvarkaraščius, kas palengvintų sprendimų priėmimą.