Incorporation of Fuzzy Preferences into Evolutionary Multiobjective Optimisation

Yaochu Jin, Bernhard Sendhoff

2002

Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning, Vol.1, pp. 26-30, Singapore, November 2002

INCORPORATION OF FUZZY PREFERENCES INTO EVOLUTIONARY MULTIOBJECTIVE OPTIMIZATION

Yaochu Jin and Bernhard Sendhoff

Future Technology Research
Honda R&D Europe (D) GmbH
Carl-Legien-Strasse 30
63073 Offenbach/Main, Germany
Email: {yaochu.jin, bernhard.sendhoff}@de.hrdeu.com

ABSTRACT

A method for incorporating fuzzy preferences into evolutionary multiobjective optimization is proposed. After introducing three commonly used models for describing fuzzy preferences, a method to convert fuzzy preferences into real-valued weight intervals is suggested. It is argued that to convert fuzzy preferences into interval-based weights is more consistent with the motivation of using fuzzy preferences than to convert them into single-valued crisp weights. The weight intervals are combined with the evolutionary dynamic weighted aggregation to obtain the preferred Pareto-optimal solutions. Simulation examples are given to show how the desired Pareto-optimal solutions can be obtained.

1. INTRODUCTION

An important issue in multiple objective optimization is the handling of human preferences. Finding all Pareto-optimal solutions is not the final goal: a decision has to be made from the available alternatives. Usually, a decision is made based on the decision-maker's preferences. Such preferences can usually be represented with the help of fuzzy logic. In general, preferences can be incorporated either before, during or after the optimization process takes place.

In [2], a method for converting linguistic fuzzy preference relations into crisp weights for optimization has been introduced. One weakness in converting fuzzy preferences into single-valued weights is that a lot of information is lost in the process. In this paper, we develop a method that converts fuzzy preference relations into interval-based weights. The interval-based weights are combined with the dynamic weighted aggregation proposed in [4, 5] to obtain the desired Pareto-optimal solutions. In this way, we are able to obtain a number of preferred Pareto solutions instead of only one.

2. FUZZY PREFERENCE MODELS AND GROUP DECISION-MAKING

2.1. Three Fuzzy Preference Models

Since human judgments including preferences are often imprecise, fuzzy logic can play an important role in decision making and multi-criterion optimization. Given n alternatives $o_1, o_2, ..., o_n$, preferences can be provided in the following three ways [1]:

- A preference ordering of the alternatives (ordered from best to worst):

$$O = \{o_1, o_2, ..., o_n\}, o_i \in [1, n]. \qquad (1)$$

  For example, the ordering $O = \{2, 1, 3, 4\}$ means that alternative o_2 is the most important and o_4 is the least important.

- Utility values. In this case, a scaled real number is assigned to each alternative to indicate its relative importance:

$$U = \{u_i; i = 1, 2, ..., n\}. \qquad (2)$$

  Utility values can be given in a difference scale or a ratio scale. If the ratio scale is used, we have $u_i \in [0, 1]$. If a difference scale is used, then u_i's are normalized so that $max\, U - min\, U \leq 1$.

- Fuzzy preference relations. In this case, the preferences are expressed by a binary relation matrix P of size $n \times n$, whose elements $p_{ij} \in [0, 1]$ are the preferences x_i over x_j that satisfy the following conditions:

$$p_{ij} + p_{ji} = 1 \qquad (3)$$

$$p_{ii} = 0.5 \qquad (4)$$

Note that both utility functions and fuzzy preference relations are numeric. Naturally, there are also utility functions and fuzzy preferences in linguistic forms:

- *Fuzzy utility functions*. Each alternative has a linguistic utility such as *Very important, Important, Don't care, Not important* and so on.
- *Linguistic fuzzy preference relations*. For example, instead of assigning a numeric fuzzy preference $p_{ij} = 0.8$, one can also express ones judgment as "objective x_i is *more important than* x_j".

Among the three preference models, fuzzy preference relations are the most widely used models. Of course, prefernce ordering and utility functions can be converted into fuzzy preferences. Consider a utility function $u(x)$ based on a difference scale, then utility values can be transformed into fuzzy preference relations as follows:

$$p_{ij} = \frac{1}{2}(1 + u_i - u_j). \tag{5}$$

Similarly, a preference ordering can also be transformed into a fuzzy preference relation. Supposing an ordering $O = \{o_1, o_2, ..., o_n\}$ is arranged from best to worst, an example function to transform the ordering into a fuzzy preference relation is

$$p_{ij} = \frac{1}{2}\left(1 + \frac{o_j - o_i}{n-1}\right). \tag{6}$$

2.2. Fuzzy Group Decision Making

In practice, there are usually several experts who make their judgments and provide preference decisions. Suppose there are m experts and their preferences of alternative x_i over x_j are p_{ij}^l, where $i, j = 1, 2, ..., n$ and $l = 1, 2, ..., m$. Therefore, there are m preference matrices $P^l, l = 1, 2, ...m$ of size $n \times n$. In decision making, an alternative can be chosen either based on a *direct or indirect approach* [6, 3].

The direct approach derives a collective preference P^c based on P^l and then uses P^c to get a solution. There are two different types of methods depending on whether numeric or linguistic fuzzy preference relations are used. If the fuzzy preferences are numeric, a collective preference can be obtained by

$$p_{ij}^c = \frac{1}{m}\sum_{l=1}^{m} p_{ij}^l. \tag{7}$$

It is transitive if all the individual preferences are transitive.

Indirect pproaches [3] to group decision making under linguistic fuzzy preference relations are generally based on two different preference degrees : a *linguistic non-dominance degree* defined in [7] and a *dominance degree* using the concept of *fuzzy majority* [6]. In [3], an indirect approach is proposed using the *Linguistic Ordered Weighted Aggregation* (LOWA) operator [8].

3. EVOLUTIONARY DYNAMIC WEIGHTED AGGREGATION

In spite of its weaknesses, the conventional weighted aggregation approach to multiobjective optimization is very attractive due to its simplicity and efficiency. Furthermore, it has been found in [4, 5] that the weaknesses of the conventional aggregation approach can be overcome by systematically changing the weights during optimization without the loss of its simplicity and efficiency. The following two methods have been proposed:

- Random weighted aggregation (RWA). It is natural to take advantage of the population for obtaining multiple Pareto-optimal solutions in one single run in evolutionary optimization. Imagine that the i-th individual in the population has its own weight combination $(w_1^i(t), w_2^i(t))$ in generation t, then the evolutionary algorithm will be able to find different Pareto-optimal solutions. To realize this, it is found that the weight combinations need to be distributed uniformly and randomly among the individuals and a re-distribution is necessary in each generation [4]:

$$w_1^i(t) = \mathrm{rdm}(P)/P, \tag{8}$$
$$w_2^i(t) = 1.0 - w_1^i(t), \tag{9}$$

  where $i = 1, 2, ..., P$ denotes the i-th individual in the population, P is the population size, and t is the index for generation number. The function $rdm(P)$ generates a uniformly distributed random number between 0 and P. In this way, we can get a uniformly distributed random weight combination (w_1^i, w_2^i) among the individuals, where $0 \le w_1^i, w_2^i \le 1$ and $w_1^i + w_2^i = 1$. Notice that the weight combinations are regenerated in every generation.

- Dynamic weighted aggregation (DWA). In DWA, all individuals have the same weight combination, which is changed gradually generation by generation. Once the individuals reach any point on the Pareto front, the slow change of the weights will force the individuals to keep moving along the Pareto front gradually if the Pareto front is convex. If the Pareto front is concave, the individuals will still traverse along the Pareto front, however, in a different fashion [4]. The change of the weights can be realized as follows:

$$w_1(t) = |\sin(2\pi t/F)|, \tag{10}$$
$$w_2(t) = 1.0 - w_1(t), \tag{11}$$

  where t is the number of generation. Here the sine function is used simply because it is a plain periodical

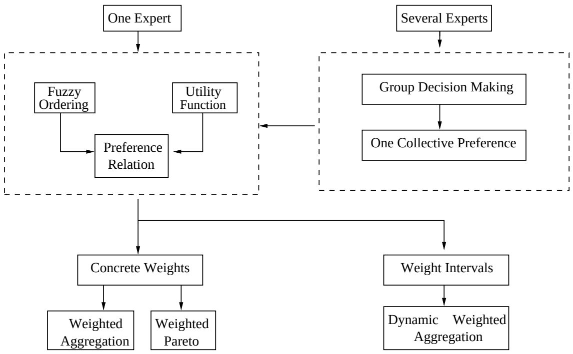


Figure 1: The procedure for application of fuzzy preferences in MOO.

function between 0 and 1. In this case, the weights $w_1(t)$ and $w_2(t)$ will change from 0 to 1 periodically from generation to generation. The change frequency can be adjusted by F. The frequency should not be too high so that the algorithm is able to converge to a solution on the Pareto front. On the other hand, it seems reasonable to let the weight change from 0 to 1 at least twice during the whole optimization.

The weights are changed between $[0, 1]$ in RWA and DWA to achieve all the Pareto-optimal solutions. In practice, it is not unusual that only part of the solutions are desired, which is specified by user preferences. In this case, the weights can be changed between $[w^{min}, w^{max}]$, where $0 \leq w^{min} < w^{max} \leq 1$.

4. FUZZY PREFERENCES INCORPORATION IN MOO

A general procedure for applying fuzzy preferences to multiobjective optimization is illustrated in Fig. 1. It is seen that before they can be incorporated into multiuobjective optimization, the fuzzy preferences must be converted into real-valued weights or weight intervals. In the following, a method for converting fuzzy preferences into weight intervals is suggested.

4.1. Converting Fuzzy Preferences into Crisp Weights

Consider an MOO problem with six objectives $\{o_1, o_2, ..., o_6\}$ [2] . Suppose among these six objectives, o_1 and o_2, o_3 and o_4 are *equally important*. Thus we have four classes of objectives: $c_1 = \{o_1, o_2\}, c_2 = \{o_3, o_4\}, c_3 = \{o_5\}$ and $c_4 = \{o_6\}$. Besides, we have the following preference relations:

c_1 is much more important than c_2;
c_1 is more important than c_3;
c_4 is more important than c_1;
c_3 is much more important than c_2

From these preferences, it is easy to get the following preference matrix:

$$P = \begin{bmatrix} EI & MMI & MI & LI \\ MLI & EI & MLI & MLI \\ LI & MMI & EI & LI \\ MI & MMI & MI & EI \end{bmatrix} \tag{12}$$

From the above fuzzy preference matrix, we can get the following real-valued preference relation matrix R:

$$R = \begin{bmatrix} \epsilon & \beta & \delta & \gamma \\ \alpha & \epsilon & \alpha & \alpha \\ \gamma & \beta & \epsilon & \gamma \\ \delta & \beta & \delta & \epsilon \end{bmatrix} \tag{13}$$

Based on this relation matrix, the weight for each objective can be obtained by:

$$w(o_i) = \frac{S(o_i, R)}{\sum_{i=1}^{k} S(o_i, R)} \ , \tag{14}$$

where

$$S(o_i, R) = \sum_{j=1, j\neq i}^{k} p_{ij} \ . \tag{15}$$

For the above example, we have

$$w_1 = w_2 = \frac{2-\alpha}{8+2\alpha}, \tag{16}$$

$$w_3 = w_4 = \frac{3\alpha}{8+2\alpha}, \tag{17}$$

$$w_5 = \frac{1-\alpha+2\gamma}{8+2\alpha}, \tag{18}$$

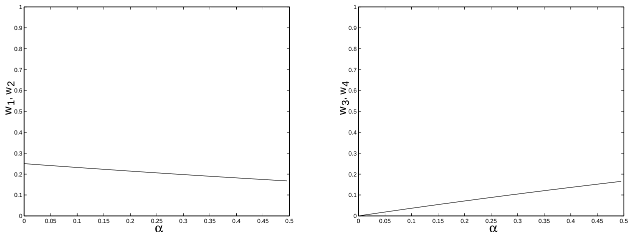
$$w_6 = \frac{3-\alpha-2\gamma}{8+2\alpha}. \tag{19}$$

Since α and γ can vary between 0 and 0.5, one needs to specify a value for α and γ heuristically (recall that $\gamma < \alpha$) to convert the fuzzy preferences into a single-valued weight combination, which can then be applied to the conventional weighted aggregation to achieve one solution.

4.2. Converting Fuzzy Preferences into Weight Intervals

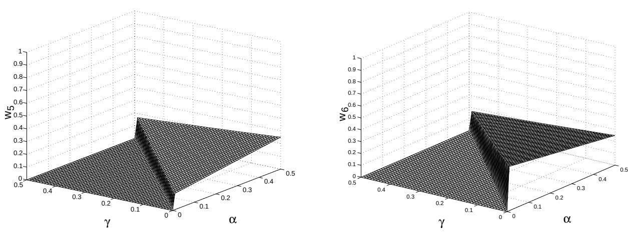
In order to convert fuzzy preferences into one weight combination, it is necessary to specify a value for α and γ. On the one hand, there are no explicit rules on how to specify these parameters, on the other hand, a lot of information will be lost in this process. One more natural way to deal with these problems is to convert the fuzzy preferences into a weight combination with each weight being described by an interval instead of a single value.

To show how the value of the parameters affects that of the weights, an experiment is carried out for the above example. Fig. 2 and Fig. 3 show the change of the weights with the change of the parameters.



(a) (b)

Figure 2: Change of the weights with the change of the parameters. (a) w_1, w_2; (b) w_3, w_4.



(a) (b)

Figure 3: Change of the weights with the change of the parameters. (a) w_5; (b) w_6.

It can be seen from the figures that the weights vary a lot when the parameters (α and γ) change in the allowed range. Thus, each weight obtained from the fuzzy preferences is an interval on $[0, 1]$. Very interestingly, a weight combination in interval values can be nicely incorporated into multiobjective optimization with the help of the RWA and DWA introduced in Section 3. Suppose the maximal and minimal value of a weight is w^{max} and w^{min} when the parameters change, then we can modify Equation (16) as follows:

$$w_1^i(t) = w_1^{min} + (w_1^{max} - w_1^{min}) * \text{rdm}(P)/P. \tag{20}$$

Similarly, Equation (18) can be modified as follows to find out the preferred Pareto solutions:

$$w_1^i(t) = w_1^{min} + (w_1^{max} - w_1^{min}) * |\sin(2\pi t/F)|. \tag{21}$$

In this way, the evolutionary algorithm can achieve a set of Pareto solutions reflected by the fuzzy preferences. However, since DWA cannot control the movement of the individuals if the Pareto front is concave, fuzzy preferences incorporation into MOO using DWA is applicable to convex Pareto fronts only, whereas RWA works for both convex and concave fronts.

To illustrate how this method works, some examples on two-objective optimization are presented in the following. In the simulations, we consider two different fuzzy preferences:

1. *Objective 1 is more important than objective 2*;
2. *Objective 1 is less important than objective 2*.

For the first preference, we can get the following preference matrix:

$$P = \begin{bmatrix} 0.5 & \delta \\ \gamma & 0.5 \end{bmatrix}, \tag{22}$$

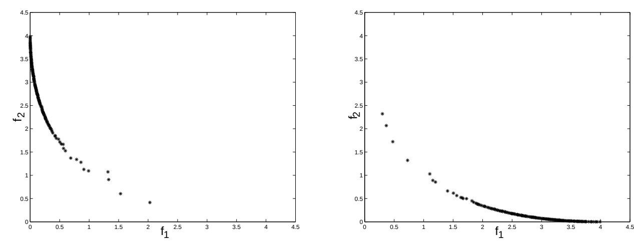
where $0.5 < \delta < 1$ and $0 < \gamma < 0.5$. Therefore, the weights for the two objectives are:

$$w_1^i(t) = 0.5 + 0.5 * \text{rdm}(P)/P, \tag{23}$$

$$w_2^i(t) = 1.0 - w_1^i(t). \tag{24}$$

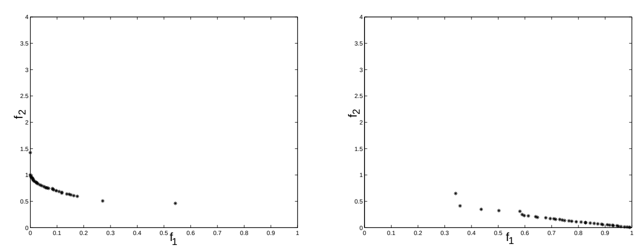
The weights for the second preference can be obtained similarly.

Simulation results are carried out on the first three test functions in [5], where F_1 and F_2 have a convex Pareto front, whereas F_3 has a concave Pareto front. The simulation results on F_1 and F_2 are given in Figures 4 and 5 using the RWA method, and in Figures 6 and 7 using the DWA method.



(a) (b)

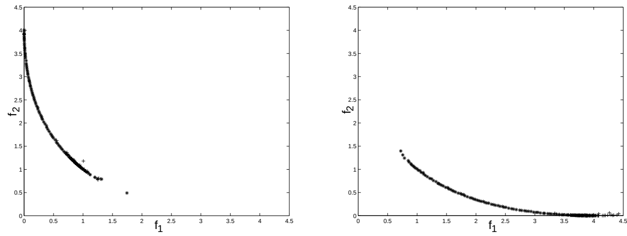
Figure 4: RWA results on F_1. (a) f_1 is more important than f_2; (b) f_1 is less important than f_2



(a) (b)

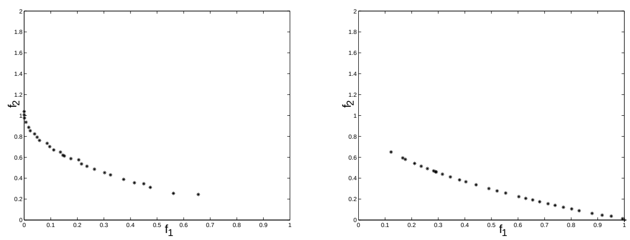
Figure 5: RWA results on F_2. (a) f_1 is more important than f_2; (b) f_1 is less important than f_2

It can be seen that the performance of RWA and DWA are similar on the two test functions with a convex Pareto



(a) (b)

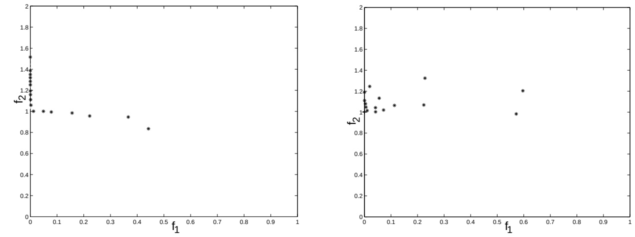
Figure 6: DWA results on F_1. (a) f_1 is more important than f_2; (b) f_1 is less important than f_2



(a) (b)

Figure 7: DWA results on F_2. (a) f_1 is more important than f_2; (b) f_1 is less important than f_2

front. However, the performance on the test function F_3, which has a concave Pareto front, is very different. The performance of the DWA is quite bad, see Fig. 8(b), whereas the performance of RWA is acceptable, refer to Fig. 8(a).



(a) (b)

Figure 8: Results on F_3 for the preference "f_1 is more important than f_2". (a) RWA; (b) DWA.

5. CONCLUSIONS

A method to obtain the Pareto solutions that are specified by human preferences is suggested. The main idea is to convert the fuzzy preferences into interval-based weights. With the help of the dynamic weighted aggregation method, it is shown to be successful to find the preferred solutions on two test functions with a convex Pareto front. Compared to the method in [2], our method is able to find a number of solutions instead of only one, given a set of fuzzy preferences over different objectives. We believe this is consistent with the motivation of fuzzy logic.

6. REFERENCES

[1] F. Chiclana, F. Herrera, and E. Herrera-Viedma. Integrating three representation models in fuzzy multipurpose decision making based on fuzzy preference relations. Technical Report DECSAI-96106, Department of Computer Science and Artificial Intelligence, University of Granada, May 1996.

[2] D. Cvetkovic and I. Parmee. Use of preferences for GA-based multi-objective optimization. In *Proceedings of 1999 Genetic and Evolutionary Computation Conference*, pages 1504–1510, San Francisco, California, 1999. Morgan Kaufmann.

[3] F. Herrera and J.L. Verdegay. On group decision making under linguistic preferences and fuzzy linguistic quantifiers. In B. Bouchon, R. Yager, and L. Zadeh, editors, *Fuzzy Logic and Soft Computing*, pages 173–180. World Scientific, 1995.

[4] Y. Jin, T. Okabe, and B. Sendhoff. Adapting weighted aggregation for multi-objective evolution strategies. In *Proceedings of First International Conference on Evolutionary Multi-Criterion Optimization*, Leture Notes in Computer Science, pages 96–110, Zurich, March 2001. Springer.

[5] Y. Jin, M. Olhofer, and B. Sendhoff. Evolutionaynamic weighted aggregation: Why does it work and how? In L. Spector, E. Goodman, A. Wu, W.B. Langdon, H.-M. Voigt, M. Gen, S. Sen, M. DOrigo, S. Pezeshk, M. Garzon, and E. Burke, editors, *Proceedings of Genetic and Evolutionary Computation Conference*, pages 1042–1049, San Francisco, CA, 2001.

[6] J. Kacprzyk. Group decision making with a fuzzy linguistic majority. *Fuzzy Sets and Systems*, 18:105–118, 1986.

[7] S.A. Orlovsky. Decision making with a fuzzy preference relation. *Fuzzy Sets and Systems*, 1:155–167, 1986.

[8] R.R. Yager. On ordered weighted averaging aggregation operators in multi-criteria decision making. *IEEE Transactions on Systems, Man, and Cybernetics*, 18:183–190, 1988.