Solving Interval Multi-objective Optimization Problems Using Evolutionary Algorithms with Preference Polyhedron

Jing Sun
School of
Information and Electrical Engineering
China University of Mining and Technology
Xuzhou, China
jing8880@sina.com

Dunwei Gong
School of
Information and Electrical Engineering
China University of Mining and Technology
Xuzhou, China
dwgong@vip.163.com

Xiaoyan Sun
School of
Information and Electrical Engineering
China University of Mining and Technology
Xuzhou, China
xysun78@126.com

ABSTRACT
Multi-objective optimization (MOO) problems with interval parameters are popular and important in real-world applications. Previous evolutionary optimization methods aim to find a set of well-converged and evenly-distributed Pareto-optimal solutions. We present a novel evolutionary algorithm (EA) that interacts with a decision maker (DM) during the optimization process to obtain the DM's most preferred solution. First, the theory of a preference polyhedron for an optimization problem with interval parameters is built up. Then, an interactive evolutionary algorithm (IEA) for MOO problems with interval parameters based on the above preference polyhedron is developed. The algorithm periodically provides a part of non-dominated solutions to the DM, and a preference polyhedron, based on which optimal solutions are ranked, is constructed with the worst solution chosen by the DM as the vertex. Finally, our method is tested on two bi-objective optimization problems with interval parameters using two different value function types to emulate the DM's responses. The experimental results show its simplicity and superiority to the *posteriori* method.

Categories and Subject Descriptors
I.2 [Artificial Intelligence]: Learning; H.1.2 [Models and Principles]: User/Machine systems- *human information processing*

General Terms
Algorithms

Keywords
evolutionary algorithm, interaction, multi-objective optimization, interval, preference polyhedron

1. INTRODUCTION
When handling optimization problems in real-world applications, it is usually necessary to consider several conflicting objectives simultaneously. Furthermore, due to many objective and/or subjective factors, these objectives and/or constraints frequently contain uncertain parameters, e.g., fuzzy numbers, random variables, and intervals. These problems are called uncertain MOO problems.

Existing methods of solving uncertain optimization problems can mainly be classified into three categories, i.e., random programming [3, 8], fuzzy programming [13, 21] and interval programming [4, 10, 14], according to the types of uncertain parameters [4]. Uncertain parameters of optimization problems solved with interval programming are intervals, and the upper and the lower limits or the midpoints and the radius of these intervals should be known beforehand. It is usually easy to acquire the values of these parameters. In addition, random variables or fuzzy numbers can be transformed into intervals by the confidence level [7] or the α -cut [21], thus, optimization problems with random parameters or fuzzy parameters can be transformed into the ones with interval parameters. Therefore, studying optimization problems with interval parameters is of theoretical significance and practical application values. We focus on the MOO problems with interval parameters in this study.

EAs are a kind of globally stochastic optimization methods inspired by nature evolution and heredity mechanisms. Since EAs can simultaneously search for several Pareto-optimal solutions in one run, they become efficient methods of solving MOO problems, such as VEGA [17] and NSGA-II [5].

EAs for MOO problems with interval parameters [10, 14] aim to find a set of well-converged and evenly-distributed Pareto-optimal solutions. However, in practice, it is necessary to arrive at the DM's most preferred solution. Thus, compared with single-objective optimization problems, in MOO, there are at least two equally important tasks: an optimization task for finding Pareto-optimal solutions and a decision-making task for choosing one most preferred solution [2]. The relationship between the two tasks causes three different methods: the first one is making decision before optimization, called *a priori* methods; the second one is making decision after optimization, called *a posteriori* methods; and the last one is making decision during optimization, called interactive methods, which are very promising and specifically suitable for interactively solving hard MOO problems.

Recent work, which will be elaborated in section 2, shows that eliciting the DM's preference information during the optimization process and constructing the DM's preference model are hot topics. Related methods can be grouped into the following three categories, i.e., machine learning based [1, 12], fit based [6, 19] and preference convex cone/polyhedron cone based [9, 18] methods. For the first two methods, pairwise comparisons of all alternatives are conducted. For the last one, it is necessary to choose the best and/or the worst solutions/solution from all alternatives, which alleviates the DM's burden in evaluations and avoids selecting a suitable explicit value function.

There exist many interactive multi-objective evolutionary algorithms (MOEAs) for deterministic MOO problems. But hitherto, to the best of our knowledge, there have been no interactive methods for MOO problems with interval parameters. Therefore, we think that seeking the DM's most preferred solution for a MOO problem with interval parameters is challenging and meaningful.

In this study, we propose a preference-based EA for MOO problems with interval parameters by employing the framework of NSGA-II, which incorporates an optimization-cum-decision-making procedure. In order to alleviate the DM's burden in evaluations, a preference polyhedron is created to approximate the DM's value function in term of the idea of convex cone [11], and used to guide the search toward the DM's preferred region. Our method's outstanding feature is that we need not know the exact form of the value function, but only assume that it is quasi-concave and non-decreasing.

The remaining of this paper is structured as follows. In section 2, the related work is reviewed. Section 3 presents the theories of preference polyhedron in the context of uncertainties. In section 4, the framework of our algorithm is expounded. The applications of our method in typical bi-objective optimization problems with interval parameters and the analysis are given in section 5. Section 6 outlines the main conclusions of our work and suggests possible opportunities to be further researched.

2. RELATED WORK

Consider the following MOO problem:

$$\begin{aligned} \max\ & f(\mathbf{x},\mathbf{c}) = (f_1(\mathbf{x},\mathbf{c}), f_2(\mathbf{x},\mathbf{c}), \cdots, f_m(\mathbf{x},\mathbf{c})) \\ \text{s.t.}\ & \mathbf{x} \in S \subseteq R^n \\ & \mathbf{c} = (c_1, c_2, \cdots, c_l)^T, c_k = [\underline{c}_k, \overline{c}_k], k = 1,2,\cdots,l \end{aligned} \tag{1}$$

where

$\mathbf{x}$ is an n -dimensional decision variable,

S is a decision space of $\mathbf{x}$,

$f_i(\mathbf{x},\mathbf{c})$ is the i -th objective function with interval parameters, $i = 1,2,\cdots,m$,

$\mathbf{c}$ is an interval vector parameter, where c_k is the k -th component of $\mathbf{c}$ with $\underline{c}_k$ and $\overline{c}_k$ being its lower and upper limits, respectively.

Each objective value in problem (1) is an interval due to its interval parameters, denoted as $f_i(\mathbf{x},\mathbf{c}) \triangleq [\underline{f}_i(\mathbf{x},\mathbf{c}), \overline{f}_i(\mathbf{x},\mathbf{c})]$. If the scale of different objective values in problem (1) is different, they need to be normalized [22]. Therefore, we assume that the scale of all objective values in problem (1) is identical.

Krettek et al. proposed a novel interactive MOEA, which combines an EA with an instance based supervised online learning scheme for the DM's preferences [12]. Battiti and Passerini adopted the methodology of reactive search optimization for interactive MOEA. The aim of the learning is to construct the approximation function of the DM's preferences [1]. The above two methods progressively acquire the DM's preference information during the optimization process. The DM's value function is approximately constructed by machine learning techniques to learn the DM's preferences, and guides subsequent evolutions of the population.

Deb et al. suggested an interactive MOEA based on preference. A strictly increasing value function is modeled by using the preference information progressively from the DM after every few generations of a MOEA. A preference based dominance principle and termination criterion are used to direct the search toward the more preferred solution [6]. Sinha et al. proposed a generalized polynomial value function to fit the DM's preference information. The number of product terms in this polynomial function is variable. This makes the algorithm more efficiently eliminates cases where the value function cannot be fitted to the DM's preferences [19]. These two methods fit the DM's preferences by an optimization procedure based on the preference information periodically provided by the DM. Explicit value functions can be obtained by these methods, while the type of functions should be chosen *a priori*.

Fowler et al. focused on multi-objective knapsack problems. They presented an interactive MOEA for quasi-concave preference functions based on the theory of [11]. In this algorithm, the solutions are periodically sent to the DM for his/her evaluation, and the resulting preference information is used to form preference cones consisting of inferior solutions. The cones implicitly rank solutions that the DM has not considered and direct the search to the DM's preferred region [9].

Sinha et al. proposed a preference based methodology, where the information provided by the DM in the intermediate runs of a MOEA is used to construct a polyhedral cone. This polyhedral cone is used to eliminate a part of search space and conduct a more focused search. The dominance principle is modified to look for better solutions lying in the preferred region [18].

The advantages of the above two methods is as follows. It is not necessary to know the explicit type of the DM's value function. The worst/the best solution chosen from alternatives and other alternatives are used to form a convex cone/polyhedral cone reflecting the DM's preferences. The dominance relation is modified on the basis of this implicit value function, and the search is focused on the preferred region.

The above methods effectively solve practical MOO problems to find the DM's most preferred solution, and numerical experiments confirm their superior capabilities in solving many-objective optimization problems as well. Nevertheless, they solely apply to deterministic MOO problems.

In this study, we progressively acquire the DM's preference information during the optimization process to construct the DM's preference model. The objective values are intervals for MOO problems with interval parameters considered here, and no approaches, which assist the DM making decision by convex cone in case of the values of alternatives being intervals, exist so far.

Therefore, we should build up the fundamental theory of the DM's preference model, i.e. preference polyhedron, in the objective space where the objective values are intervals. The construction and application of the preference polyhedron are also our core techniques.

3. PREFERENCE POLYHEDRAL FOR INTERVAL MOO PROBLEMS

For the basic knowledge of interval arithmetic, please refer to [15].

Denote the set of all closed intervals on R as $I(R)$ and a subspace on R^n as $D = D_1 \times D_2 \times \cdots \times D_n \subseteq R^n$, then the set of all n-dimensional interval vectors on D can be denoted as $I(D) = \left\{ A \middle| A \in I(R^n), a_i \subseteq D_i, i = 1, 2, \ldots, n \right\}$.

The definitions of the convex set and the quasi-concave function based on intervals are given in the next subsection based on the definitions in [16].

3.1 Convex Set and Quasi-concave Function for Interval

Definition 1 For $\forall A, B \in I(D)$, if

$$\lambda A + (1-\lambda) B \in I(D), \lambda \in [0,1] \tag{2}$$

is held, $I(D)$ is called a convex interval set on $I(R^n)$.

Definition 2 If $A_1, A_2, \ldots, A_m$ are n-dimensional interval vectors in $I(D)$, then

$$E = \left\{ A \middle| A = \sum_{i=1}^{m} \mu_i A_i, \sum \mu_i = 1, \mu_i \geq 0 \right\} \tag{3}$$

is called a convex polyhedron generated by $A_1, A_2, \ldots, A_m$.

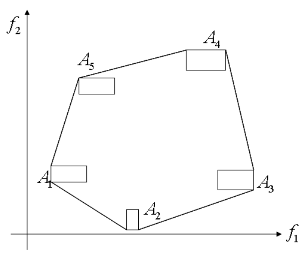

Figure 1. Convex polyhedron generated by two-dimensional interval vectors.

Figure 1 illustrates a convex polyhedron generated by two-dimensional interval vectors $A_1, A_2, \ldots, A_5$.

Definition 3 Let $F(\Theta)$ be an interval function defined on a convex interval set $I(D)$. For $\forall A, B \in D$ and $\lambda \in [0,1]$, if $F(\lambda A + (1-\lambda)B) \geq_{IN} \min\{F(A), F(B)\}$ is held, $F(\Theta)$ is called a quasi-concave interval function.

A quasi-concave interval function can be obtained by replacing real arguments of a quasi-concave real function with intervals.

3.2 Preference Polyhedron

THEOREM 1 Assume a quasi-concave interval function $F(\Theta)$ is defined in an n-dimensional metric space. Consider the distinct alternative solution points $g_i \in I(R^n), i = 1, 2, \ldots, m$, where m is the number of solution points, and the convex polyhedron E is generated by these points. Assume that $F(g_k) = \min_i F(g_i)$, if $y \subset E$, it follows that $F(y) \geq_{IN} F(g_k)$.

THEOREM 2 Assume the conditions are the same as the above theorem, and $F(g_i) >_{IN} F(g_k)$, $i \neq k$, if $z \subset Z$ and $z \neq g_k$, where $Z = \left\{ z \middle| z = \sum_{i=1, i \neq k}^{m} \mu_i (g_k - g_i), \mu_i \geq 0 \right\}$, it follows that $F(g_k) \geq_{IN} F(z)$.

The proofs of these theorems are omitted for brevity.

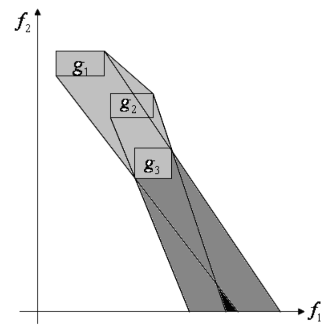

Figure 2. Application of theorems.

Figure 2 illustrates the application of the theorems with a simple bi-criterion example. For three solutions, say g_1, g_2 and g_3, if g_3 is the worst solution in Figure 2, it follows that any solution lying in the light grey region is at least as preferred as g_3 according to theorem 1, and any solution lying in the dark grey region is not preferred to g_3 from theorem 2. Further, any solution lying in the black region is not preferred to g_3. Then, we can draw the following conclusion: any solution lying in either the dark grey region or the black region is not preferred to g_3. So, when a decision is made among many solutions, all solutions lying in the dark grey and the black regions may be eliminated and need never be evaluated. Consequently, the number of evaluations can be greatly reduced.

It can be observed from Figure 2 that all solutions can be divided into three categories, i.e., the preferred, the non-preferred and the uncertain preference solutions. This means that a solution can be ranked based on the convex polyhedron in Figure 2, called the preference polyhedron in this study. Accordingly, the regions in which three kinds of solutions lie are called the preferred, the non-preferred and the uncertain preference regions, respectively.

4. IEA WITH PREFERENCE POLYHEDRON

We propose an IEA for MOO problems with interval parameters based on the preference polyhedron in this section. Having evolved τ generations by an EA for MOO problems with interval parameters, we provide the DM with $\eta \geq 2$ sparse optimal solutions from the non-dominated solutions every τ generations, and request the DM to choose the worst one from them and the best one from them and the recent best solution. With these optimal solutions sent to the DM, we create a preference polyhedron in the objective space, expounded in subsection 4.1.

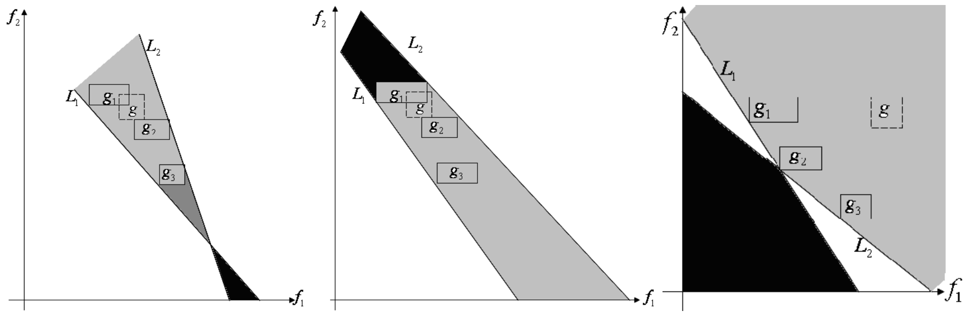

Figure 3(a) **Figure 3(b)** **Figure 3(c)**

Figure 3. Locations of g_k in objective space.

Till the next τ generations, we use the constructed preference polyhedron to sort individuals, described in subsection 4.2. When the termination criterion is met, the DM is asked to select the most preferred one from non-dominated and the recent best solutions. The detailed steps are as follows:

Step 1 Initialize a population $P(0)$ of size N; let $t=0$ and an EA for MOO problems with interval parameters is executed for τ generations; the value of t is incremented by one after each generation;

Step 2 If $(t\text{-}1) \bmod \tau = 0$, and the number of non-dominated solutions is not less than 2, choose η sparse optimal solutions; otherwise, go to step 4;

Step 3 Select the worst and the best solutions, and construct a preference polyhedron with the worst solution as the vertex;

Step 4 Employ the tournament selection of size 2, and perform crossover and mutation operations to create an offspring $Q(t)$ of size N;

Step 5 Combine $P(t)$ and $Q(t)$, and denote the combination as $R(t)$;

Step 6 Rank optimal solutions based on the preference polyhedron, and select the first N superior individuals to form $P(t+1)$;

Step 7 Judge whether the algorithm's termination criterion is met. If yes, choose the most preferred solution; otherwise, let $t=t+1$, and go to step 2.

Remark: IP-MOEA is employed to evolve the initial population for τ generations in this study.

4.1 Constructing Preference Polyhedron

Given the complexity of constructing the preference polyhedron in a high-dimensional objective space where the values are intervals, we only discuss the case of a two-dimensional space. The construction of the preference polyhedron and the ranking method based on the above polyhedron, however, are suitable for any multi-objective optimization problem as well.

We simplify the DM's preferred and non-preferred regions as follows:

(1) For the preferred region, we extend it to the light grey region in Figure 3, suggesting that we regard a part of uncertain reference individuals as the preferred ones;

(2) For the non-preferred region, we reduce it to the black region, shown as Figure 3(b) and Figure 3(c), or the dark grey and the black regions, shown as Figure 3(a), indicating that a part of non-preferred individuals are considered as the uncertain preference ones.

The detailed steps of constructing a preference polyhedron are as follows:

First, select the worst value, denoted as $g_k = \left(\left[\underline{g}_k^1, \overline{g}_k^1\right], \left[\underline{g}_k^2, \overline{g}_k^2\right]\right)$, from η objectives, i.e. $g_1, g_2, \cdots, g_\eta$. The location of g_k in the objective space may be one of the following cases: (1) g_k has the minimal value in the first objective among η objectives, as g_1 of Figure 3 (b); (2) g_k has the minimal value in the second objective, as g_3 of Figure 3 (a); (3) g_k has no minimal value in any objective, as g_2 of Figure 3 (c).

Then, calculate the slopes of lines L_1 and L_2 in Figure 3 according to the above three cases. For the first case, calculate $k_{1i} = \dfrac{\underline{g}_i^2 - \underline{g}_k^2}{\underline{g}_i^1 - \underline{g}_k^1}$ and $K_{2i} = \dfrac{\overline{g}_i^2 - \overline{g}_k^2}{\overline{g}_i^1 - \overline{g}_k^1}$, and then the slops of lines L_1 and L_2 are $K_1 = \min\limits_{\substack{i\in\{1,\cdots,\eta\} \\ i\neq k}} K_{1i}$ and $K_2 = \max\limits_{\substack{i\in\{1,\cdots,\eta\} \\ i\neq k}} K_{2i}$, respectively. The slopes for the other two cases can be calculated in a similar way;

Finally, construct an open convex polygon, i.e. preference polyhedron, by taking g_k and lines L_1 as well as L_2 as the vertex and edges, respectively.

4.2 Ranking Optimal Solutions Based on Preference Polyhedron

The detailed method is as follows: first, the dominance relation based on intervals [14] is used to sort $R(t)$; then, the individuals with the same rank are classified into three categories, i.e. the preferred, the uncertain preference and the non-preferred individuals; finally, the individuals with both the same rank and category are further ranked based on the crowding metric [14].

We sort the individuals with the same rank using the constructed preference polyhedron in subsection 4.1. The individuals in the objective space have three possible locations relative to the polyhedron: (1) inside the polyhedron, which are placed first when sorting, denoted as *S*; (2) below the polyhedron, which are placed last, denoted as *NS*; (3) outside the polyhedron, i.e.,

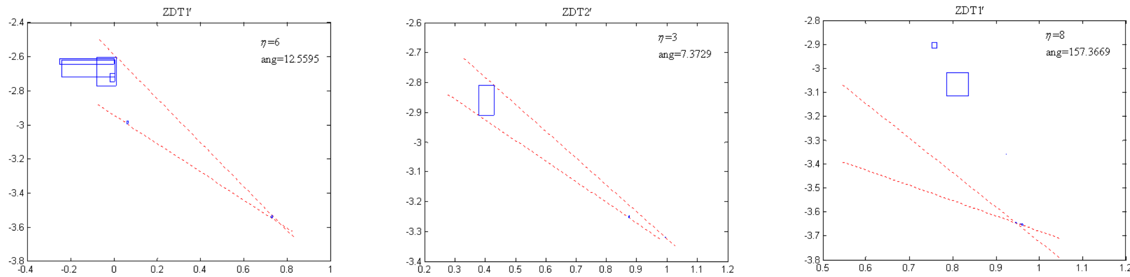

Figure 4(a) **Figure 4(b)** **Figure 4(c)**

Figure 4. Three types of preference.

outside the region formed by lines L_1 and L_2, which are placed in the middle, denoted as *U*. It can be observed from Figure 3 that no matter what the situation of g_k, the region formed by these two lines includes the interior and the bottom of the polyhedron.

Consider the second case, shown as Figure 3(a). If the lower left and the upper right vertices of individual g lies above line L_1 and below L_2, respectively, or the lower left and the upper right vertices of g lies above L_2 and below L_1, respectively, then g is in the interior or the bottom of the polyhedron. To be more specific, if g lies above the worst solution, i.e., $\underline{g}^2 > \underline{g}_k^2$, then g is in the interior of the polyhedron; while lies below, i.e., $\underline{g}^2 \le \underline{g}_k^2$, then g is in the bottom of the polyhedron. Otherwise, g lies outside the polyhedron.

The other two cases can be analogously discussed, and the detailed analysis is omitted here for brevity.

The above ranking method is suitable to select individuals in step 4. The detailed process is as follows. Randomly select two individuals from the population, first, compare them using the non-dominated sorting, and select the one with a lower rank. If their ranks are same, compare them by the preference polyhedron, and select the one according to the preferred, the uncertain preference and the non-preferred orders. If they have the same preference, select the one with a larger crowding metric; otherwise, randomly choose one.

5. APPLICATIONS IN INTERVAL MOO PROBLEMS

The proposed algorithm's performances are confirmed by optimizing two benchmark bi-objective optimization problems and comparing with an *a posteriori* method. The implementation environment is as follows: Pentium(R) Dual-Core CPU, 2G RAM, and Matlab7.0.1. Each algorithm is run for 20 times independently, and the averages of these results are calculated. We choose two bi-objective optimization problems with interval parameters, i.e. $\text{ZDT1}'$ and $\text{ZDT2}'$, from [14] as benchmark problems.

5.1 Value Function

For $\text{ZDT1}'$ and $\text{ZDT2}'$, the following quasi-concave increasing value function

$$V_1(f_1, f_2) = (f_1 + 0.4)^2 + (f_2 + 5.5)^2 \qquad (4)$$

and linear value function

$$V_2(f_1, f_2) = 1.25 f_1 + 1.50 f_2 \qquad (5)$$

are used to emulate the DM to make decision, respectively.

5.2 Parameter Settings

Our algorithm is run for 200 generations with the population size of 40. Simulated binary crossover (SBX) operator and polynomial mutation [5] are employed, and the crossover and mutation probabilities are set to 0.9 and $1/30$, respectively. In addition, the distribution indices for crossover and mutation operators with $\eta_c = 20$ and $\eta_m = 20$ are adopted, respectively. The number of decision variables, in the range of $[0,1]$, is 30 for the two test problems.

There are another two parameters in our algorithm: the number of generations between two consecutive decision-making, τ, and the number of individuals provided to the DM for evaluation, η. The human fatigue problem is inherent to IEAs, and the algorithm leads to a rapid convergence for a small population [20]. In order to actually simulate the DM's interaction, the minimal value of τ and the maximal one of η are 8 and 10, respectively, in our experiments.

5.3 Performance Measures

The following three measures are employed to investigate the performance of our algorithm in our experiments:

(1) The best value of the preference function (V metric, for short). This index measures the degree of the DM's satisfaction with the optimal solution. The larger the value of V metric, the higher the DM's satisfaction with the optimal solution is.

(2) The number of uncertain preference individuals (U metric, for short). This index reflects the degree of the DM's cognition on the evaluated individuals. The smaller the value of U metric, the clearer the DM's preference is.

(3) The angle between lines L_1 and L_2 (A metric, for short). This index reflects the degree of the DM's cognition on the evaluated individuals as well. The larger the value of A metric, the clearer the DM's preference is, and the easier the DM's preferred solution is to be found.

5.4 Results and Analysis

Our experiments are divided into four groups. The first and the second ones investigate two properties of our algorithm, including the construction of the preference polyhedron and the convergence in the objective space. The third and the forth ones examine the influences of different values of τ and η on the performance of our algorithm, respectively. We also compare the

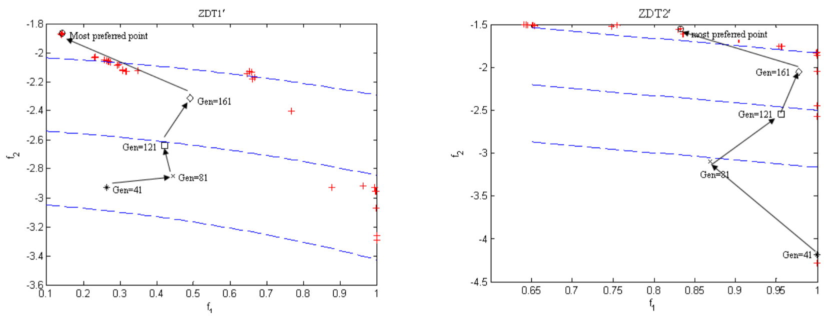

Figure 5. Course of searching for most preferred point.

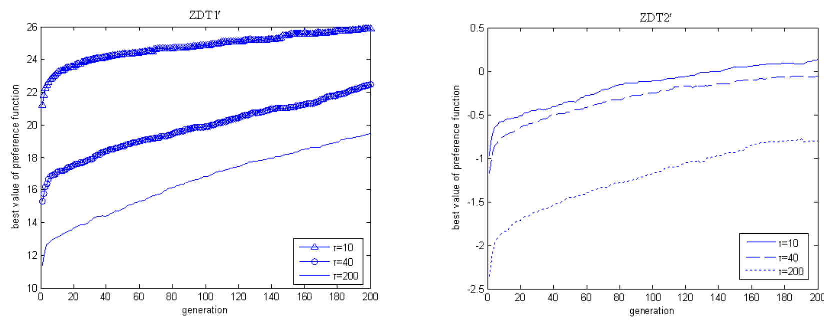

Figure 6. Curves of V metrics w.r.t. number of generations when $\eta = 8$.

proposed method with the *posteriori* one, i.e., the value of τ is 200, and the decision-making is executed at the end of the algorithm, in the third group.

5.4.1 Constructing Preference Polyhedron

The value of τ is set to be 10 in the first group of experiments. Figure 4 depicts three types of preference polyhedrons constructed at the 51st generation, indicating that all three cases discussed in subsection 4.1 may appear during the evolution.

Different types of preference polyhedrons represent different directions and ranges of the search. For instance, the first type, shown as Figure 4 (a), shows the upper left direction is the DM's preference direction, and the upper left part of the region formed by two dotted lines is the DM's preferred region.

5.4.2 Convergence of Our Algorithm

The values of τ and η are set to be 20 and 8, respectively, in the second group of experiments. In Figure 5, the asterisk, the fork, the box and the diamond represent the best point of 41st, 81st, 121st and 161st generation, respectively, the circle and the plush sigh represent the DM's most preferred point and the Pareto front, respectively, and dotted lines are the contours of the value function. Figure 5 illustrates the course of searching for the most preferred point.

It can be easily observed from Figure 5 that:

(1) The value of the best point increases along with the evolution of a population, indicating that the best point is more and more suitable to the DM's preferences.

(2) The best point converges to the most preferred point, which suggests that our algorithm is convergent and the DM can surely arrive at his/her most preferred solution.

(3) The search direction constantly changes along with the evolution of a population, which implies that specifying the search direction *a priori* is unpractical, and interaction with the DM during the optimization process is necessary to obtain his/her most preferred solution.

5.4.3 Influence of τ on Our Algorithm's Performance

The value of η is set to be 8 in the third group of experiments. Figure 6 shows the curves of V metrics w.r.t. the number of generations for different values of τ. It can be observed from Figure 6 that:

(1) For the same value of τ, the value of V metric increases along with the evolution of a population, indicating that the obtained solution is more and more suitable to the DM's preferences.

(2) For the same generation, the value of V metric increases along with the decrease of the value of τ, or equivalently the increase of the interaction frequency, suggesting that the more frequent the interaction, the better the most preferred solution is. The interactive method thus obviously outperforms the *posteriori* method.

Figure 7 illustrates the curves of U metrics w.r.t. the number of generations for different values of τ. It can be observed from Figure 7 that for the same generation, the value of U metric has no relationship with the interaction frequency in early stage of the evolution, and its value obviously decreases along with the increase of the interaction frequency in later stage. This implies that the interaction frequency seldom affects the evolution in early stage, whereas the increase of the interaction frequency can make the DM's preferences clearer in later stage.

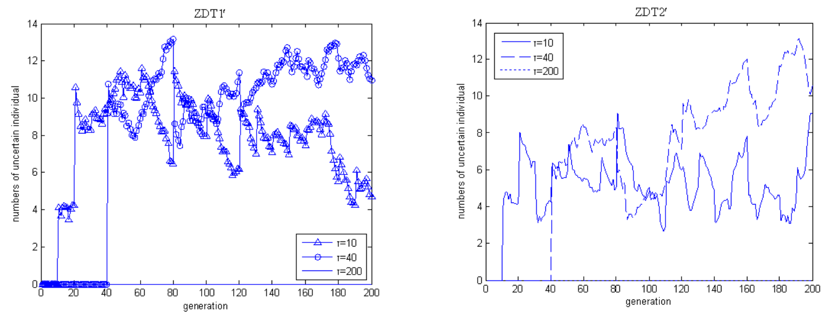

Figure 7. Curves of U metrics w.r.t. number of generations when $\eta = 8$.

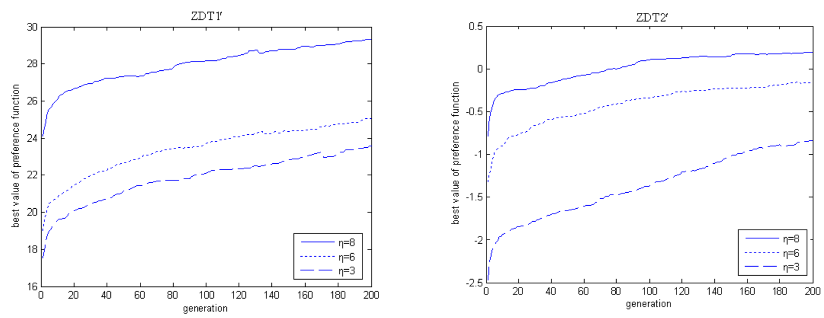

Figure 8. Curves of V metrics w.r.t number of generations when τ=10 .

5.4.4 Influence of η on Our Algorithm's Performance

The value of τ is set to be 10 in the last group of experiments. Figure 8 shows the curves of V metrics w.r.t the number of generations for different values of η .

As it can be observed from Figure 8 that:

(1) For the same value of η , the value of V metric increases along with the number of generations, indicating that the obtained optimal solution is more and more suitable for the DM's preferences.

(2) For the same generation, the value of V metric increases along with the increase of the value of η , suggesting that the larger the value of η , the easier the DM finds the most preferred solution.

Table 1. U and A metrics w.r.t. different value of τ

	ZDT1′		ZDT2′	
	U metric	A metric	U metric	A metric
$\eta = 3$	21.88	17.12	24.71	16.63
$\eta = 6$	12.71	19.32	9.78	25.89
$\eta = 8$	10.90	22.26	5.80	27.12

Figure 9 shows the curves of U metrics w.r.t. the number of generations for different values of η . We can observe from Figure 9 that for the same generation, the value of U metric decreases along with the increase of the value of η , which implies that the larger the value of η , the clearer the DM's preference is.

Table 1 lists the averages of U and A metrics w.r.t. the number of generations for different values of η . It can be observed from Table 1 that the larger the value of η , the bigger the average of A metrics is, whereas the smaller the average of U metrics is, which further indicates that the larger the value of η , the clearer the DM's preference is.

Based on the above experimental results and analysis, we draw the following conclusions: (1) our method outperforms the *posteriori* method; (2) if both the interaction frequency of the DM and the value of η increase, our algorithm's performance will be significantly improved, while the DM's burden in evaluations is also increased.

6. CONCLUSIONS

MOO problems with interval parameters are popular and important, few effective methods of solving them, however, exist as a result of their complexity.

We focus on these problems and present an interactive evolutionary method of solving MOO problems with interval parameters. The preference polyhedron is employed to sort optimal solutions, and guide the search to the DM's preferred region to finally obtain his/her most preferred solution. The key techniques of our algorithm are constructing the preference polyhedron and sorting individuals using the above polyhedron.

As analyzed earlier, the interaction frequency seldom affects the evolution in the early stage of the evolution, and the increase of the interaction frequency can evidently improve our algorithm's performance in the later stage. The DM's burden in evaluations, however, is increased. If the interaction frequency is appropriately adjusted along with different stages of the evolution, not only can our algorithm alleviate the DM's burden in evaluations, but also

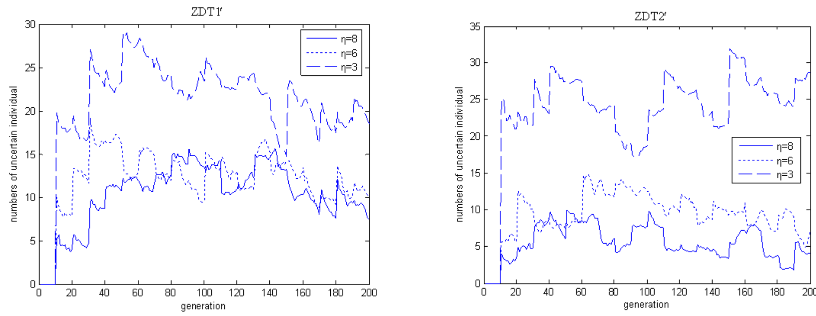

Figure 9 Curves of U metrics w.r.t. number of generations when τ=10 .

has good performance. Therefore, determining a proper interaction frequency according to the stage of the evolution is our future research topic.

7. ACKNOWLEDGMENTS

This work was jointly supported by National Natural Science Foundation of China, grant No. 60775044, Program for New Century Excellent Talents in Universities, grant No. NCET-07-0802, and Natural Science Foundation of HHIT, grant No. 2010150037.

8. REFERENCES

[1] R.Battiti and A.Passerini. Brain-computer evolutionary multiobjective optimization: A genetic algorithm adapting to the decision maker. *IEEE Trans. Evol. Comput.*, 14(5):671－687, October 2010.

[2] J. Branke, K. Deb, K. Miettinen, and R. Slowinski. *Multiobjective optimization - Interactive and Evolutionary Approaches*. Springer, Heidelberg, 2008.

[3] J. Catro. A stochastic programming approach to cash management in banking. *Eur. J. Oper. Res.*, 192(3):963－974, February 2009.

[4] Z. Q. Chen, L. K. Dai, and Y. X. Sun. Feasibility analysis for optimization of uncertain systems with interval parameters. *Acta Automatica Sinca*, 30(3):455-459, March 2004.

[5] K. Deb, A. Pratap, S. Agarwal, and T. Mrystivan. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.*, 6(2):182－197, April 2002.

[6] K. Deb, A. Sinha, P. Korhonen, and J. Wallenius. An interactive evolutionary multi-objective optimization method based on progressively approximated value functions. *IEEE Trans. Evol. Comput.*, 14(5):723－739, October 2010.

[7] H. Eskandari, C. D. Geiger, and R. Bird. Handling uncertainty in evolutionary multiobjective optimazation: SPGA. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 4130－4137. IEEE, September 2007.

[8] S. E. Fleten and T. K. Kristoffersen. Short-term hydropower production planning by stochastic programming. *Comput. Oper. Res.*, 35(8):2656－2671, August 2008.

[9] J. W. Fowler, E. S. Gel, M. M. Koksalan, P. Korhonen, J. L. Marquis, and J. Wallenius. Interactive evolutionary multi-objective optimization for quasi-concave preference functions. *Eur. J. Oper. Res.*, 26(2):417–425, 2010.

[10] D. W. Gong, N. N. Qin, and X. Y. Sun. Evolutionary optimization algorithm for multi-objective optimization problems with interval parameters. In *Proceedings of Fifth IEEE International Conference on Bio-Inspired Computing: Theories and Applications*, pages 411–420, 2010.

[11] P. Korhonen, J. Wallenius, and S. Zionts. Solving the discrete multiple criteria problem using convex cones. *Management Science*, 30(11):1336–1345, 1984.

[12] J. Krettek, J. Braun, F. Hoffmann, T. Bertram, T. Ewald, H.-G. Schubert, and H. Lausch. Interactive evolutionary multiobjective optimization for hydraulic valve controller parameters. In *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 816–821, 2009.

[13] M.Kumar, D.Arndt, S.Kreuzfeld, K.Thurow, N.Stoll, and R.Stoll. Fuzzy techniques for subjective workload-score modeling under uncertainties. *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, 38(6):1449–1464, December 2008.

[14] P. Limbourg and D. E. Aponte. An optimization algorithm for imprecise multi-objective problem functions. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 459–466, 2005.

[15] R.E. Moore, R.B. Kearfott, and M.J. Cloud. *Introduction to interval analysis*. SIAM, Philadelphia, 2009.

[16] R.T. Rockafellar. *Convex analysis*. Princeton University Press, New Jersey, 1970.

[17] J. D. Schaffer. Multiple objective optimization with vector evaluated genetic algorithms. In *Proceedings of the First International Conferenceon Genetic Algorithms*, pages 93–100, 1985.

[18] A. Sinha, K. Deb, P. Korhonen, and J. Wallenius. An interactive evolutionary multiobjective optimization method based on polyhedral cones. In *Proceedings of Learning and Intelligent Optimization Conference*, pages 318–332, 2010.

[19] A. Sinha, K. Deb, P. Korhonen, and J. Wallenius. Progressively interactive evolutionary multiobjective optimization method using generalized polynomial value functions. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1–8, 2010.

[20] H.Takagi. Interactive evolutionary computation: fusion of the capabilities of EC optimization and human evaluation. In *Proceedings of the IEEE*, pages 1275–1296. IEEE, September 2001.

[21] H. F. Wang and M. L. Wang. A fuzzy multiobjective linear programming. *Fuzzy sets Syst.*, 86(1):61–72, 1997.

[22] Z.Yong, G.Dun-wei, and Z.Yong. Evolutionary optimization methods for hybrid index optimization problems and application. *Control and Decision*, 22(3):352–356, March 2007.